



Supervisione e controllo XML-based
da Windows a Windows CE

Manuale Programmazione Linguaggio VBA

Versione 11.6- Ed. ott.2019

Sommario

1. LINGUAGGIO VBA	1
1.1. INTRODUZIONE	1
1.1.1. Premesse	1
1.2. CONCETTI GENERALI	1
1.2.1. Basic Script nel Progetto	1
1.3. BASIC SCRIPT NEL PROGETTO	1
1.3.1. WinWrap Basic Language	1
1.4. SUPPORTO VB.NET E UNICODE	2
1.4.1. Librerie Basic Script	3
1.4.2. Sub, Function, Eventi, Metodi e Proprietà	4
1.4.3. Variabili nei Basic Script	7
1.4.4. Programmazione Rapida	8
1.5. CODICE NEI BASIC SCRIPT E NEI SIMBOLI	9
1.5.1. Basic Script come Risorsa	9
1.5.2. Espressioni Basic Script nelle Proprietà degli Oggetti	11
1.5.3. Basic Script VBA™ nel Codice degli Oggetti e Allarmi	13
1.6. BASIC SCRIPT CONDIVISI	14
1.6.1. Editor Basic Script	15
1.7. EDITOR BASIC SCRIPT	15
1.7.1. Menù Modifica (Basic Script)	17
1.7.2. Menù Debug	18
1.7.3. Barra Utensili Basic Script	19
1.8. FINESTRA ESPLORATORE SCRIPT	20
1.9. DEBUG DEI BASIC SCRIPT	22
1.9.1. Proprietà Basic Script	23
1.10. PROPRIETÀ BASIC SCRIPT	23
1.10.1. Proprietà Generali di un Basic Script	23
1.10.2. Proprietà Modo di un Basic Script	23
1.10.3. Proprietà Esecuzione di un Basic Script	24
1.10.4. Proprietà Script Debug Output di un Basic Script	25
1.10.5. ActiveX/OCX	25
1.11. ADO NEI BASIC SCRIPT	25
1.12. ACTIVEX/OCX NEI BASIC SCRIPT	27
1.12.1. Come utilizzare ActiveX con Licenza	28
1.12.2. Esempio di utilizzo ActiveX/OCX nei Basic Script	29
1.13. API INTERFACCE BASIC	31
1.14. UTILIZZO DELLE INTERFACCE BASIC SCRIPT	31
1.14.1. AlarmCmdTarget	35
Functions	35
GetAlarmThreshold, AlarmCmdTarget Function	35
GetXMLSettings, AlarmCmdTarget Function	36
Properties	36
AlarmOnQualityGood, AlarmCmdTarget Property	36
DeviceName, AlarmCmdTarget Property	37
Enabled, AlarmCmdTarget Property	37
EnableDispatchingVariableName, AlarmCmdTarget Property	38
EnableVariableName, AlarmCmdTarget Property	38
Isteresis, AlarmCmdTarget Property	39
Name, AlarmCmdTarget Property	39
ThresholdExclusive, AlarmCmdTarget Property	39
VariableName, AlarmCmdTarget Property	40
1.14.2. AlarmThresholdCmdTarget	40
Events	40
AlarmLoading, AlarmThresholdCmdTarget Event	40
AlarmUnloading, AlarmThresholdCmdTarget Event	41
OnAckAlarm, AlarmThresholdCmdTarget Event	41
OnCommentAlarm, AlarmThresholdCmdTarget Event	41
OnHelpAlarm, AlarmThresholdCmdTarget Event	41
OnResetAlarm, AlarmThresholdCmdTarget Event	42
OnSetAlarm, AlarmThresholdCmdTarget Event	42
Functions	42

GetTransactionID, AlarmThresholdCmdTarget Function	42
GetUniqueID, AlarmThresholdCmdTarget Function.....	43
GetTotNumReset, AlarmThresholdCmdTarget Function.....	43
GetTotNumAck, AlarmThresholdCmdTarget Function.....	44
GetTotNumOn, AlarmThresholdCmdTarget Function.....	44
ResetAlarm, AlarmThresholdCmdTarget Function	45
GetAlarmObject, AlarmThresholdCmdTarget Function	45
GetXMLSettings, AlarmThresholdCmdTarget Function	46
AckAlarm, AlarmThresholdCmdTarget Function	46
Properties.....	47
AlarmArea, AlarmThresholdCmdTarget Property	47
Attachment, AlarmThresholdCmdTarget Property	47
BackColor, AlarmThresholdCmdTarget Property.....	47
Beep, AlarmThresholdCmdTarget Property.....	48
BlinkBackColor, AlarmThresholdCmdTarget Property.....	48
BlinkOnNewAlarm, AlarmThresholdCmdTarget Property.....	49
BlinkTextColor, AlarmThresholdCmdTarget Property	49
CommentOnAck, AlarmThresholdCmdTarget Property	49
CommandList..., AlarmThresholdCmdTarget Property	50
Condition, AlarmThresholdCmdTarget Property	51
DateTimeACK, AlarmThresholdCmdTarget Property	51
DateTimeACKMs, AlarmThresholdCmdTarget Property.....	52
DateTimeFromTimeStamp, AlarmThresholdCmdTarget Property	52
DateTimeOFF, AlarmThresholdCmdTarget Property.....	53
DateTimeOFFMs, AlarmThresholdCmdTarget Property	53
DateTimeON, AlarmThresholdCmdTarget Property	53
DateTimeOnMs, AlarmThresholdCmdTarget Property	54
DateTimeRESET, AlarmThresholdCmdTarget Property	54
DateTimeRESETMs, AlarmThresholdCmdTarget Property	55
DurationFormat, AlarmThresholdCmdTarget Property.....	55
Help, AlarmThresholdCmdTarget Property	56
LastComment, AlarmThresholdCmdTarget Property	56
LastTotalTimeOn, AlarmThresholdCmdTarget Property	57
Log, AlarmThresholdCmdTarget Property.....	57
Name, AlarmThresholdCmdTarget Property	58
PlaysoundContinuously, AlarmThresholdCmdTarget Property	58
Print, AlarmThresholdCmdTarget Property	59
ReadAccessLevel, AlarmThresholdCmdTarget Property	59
Recipient, AlarmThresholdCmdTarget Property.....	59
RepeatSpeechEverySec, AlarmThresholdCmdTarget Property.....	60
SecDelay, AlarmThresholdCmdTarget Property.....	60
SendFaxEnabledACK, AlarmThresholdCmdTarget Property.....	61
SendFaxEnabledOFF, AlarmThresholdCmdTarget Property	61
SendFaxEnabledON, AlarmThresholdCmdTarget Property	62
SendFaxEnabledRESET, AlarmThresholdCmdTarget Property	62
SendMailEnabledACK, AlarmThresholdCmdTarget Property	62
SendMailEnabledOFF, AlarmThresholdCmdTarget Property	63
SendMailEnabledON, AlarmThresholdCmdTarget Property.....	63
SendMailEnabledRESET, AlarmThresholdCmdTarget Property	64
SendSMSEnabledACK, AlarmThresholdCmdTarget Property	64
SendSMSEnabledOFF, AlarmThresholdCmdTarget Property	65
SendSMSEnabledON, AlarmThresholdCmdTarget Property	65
SendSMSEnabledRESET, AlarmThresholdCmdTarget Property	65
SendVoiceEnabledACK, AlarmThresholdCmdTarget Property	66
SendVoiceEnabledOFF, AlarmThresholdCmdTarget Property.....	66
SendVoiceEnabledON, AlarmThresholdCmdTarget Property.....	67
SendVoiceEnabledRESET, AlarmThresholdCmdTarget Property	67
Severity, AlarmThresholdCmdTarget Property	68
SpeechEnabled, AlarmThresholdCmdTarget Property	68
SpeechEnableVariable, AlarmThresholdCmdTarget Property.....	69
Status, AlarmThresholdCmdTarget Property.....	69
StatusVariable, AlarmThresholdCmdTarget Property	70
SupportAcknowledge, AlarmThresholdCmdTarget Property	70
SupportReset, AlarmThresholdCmdTarget Property.....	70
TextColor, AlarmThresholdCmdTarget Property	71
Text, AlarmThresholdCmdTarget Property	71
SupportResetWithConditionOn, AlarmThresholdCmdTarget Property	72
Threshold, AlarmThresholdCmdTarget Property.....	72
ThresholdLow, AlarmThresholdCmdTarget Property	72
TotalTimeOn, AlarmThresholdCmdTarget Property	73
VariableSeverity, AlarmThresholdCmdTarget Property.....	73
VariableThreshold, AlarmThresholdCmdTarget Property	74

VariableThresholdLow, AlarmThresholdCmdTarget Property	74
WriteAccessLevel, AlarmThresholdCmdTarget Property	75
1.14.3. AlarmWndCmdTarget	75
Events.....	75
OnAckAll, AlarmWndCmdTarget Event	75
OnAckSel, AlarmWndCmdTarget Event	76
OnCommentSel, AlarmWndCmdTarget Event	76
OnGetHistory, AlarmWndCmdTarget Event	76
OnHelp, AlarmWndCmdTarget Event	76
OnInsertOrUpdateAlarm, AlarmWndCmdTarget Event	77
OnOPCAEEEvent, AlarmWndCmdTarget Event	77
OnResetAll, AlarmWndCmdTarget Event	77
OnResetSelSel, AlarmWndCmdTarget Event	78
OnToggleSound, AlarmWndCmdTarget Event	78
Functions.....	78
AckSelectedAlarms, AlarmWndCmdTarget Function	78
EditLayout, AlarmWndCmdTarget Function.....	79
EditCopy, AlarmWndCmdTarget Function	79
GetNumTotalAlarms, AlarmWndCmdTarget Function	80
GetSelectedAlarm, AlarmWndCmdTarget Function	80
GetSelHistory, AlarmWndCmdTarget Function	80
LoadExtSettings, AlarmWndCmdTarget Function.....	81
RecalcLayout, AlarmWndCmdTarget Function	81
ResetSelectedAlarms, AlarmWndCmdTarget Function	82
Refresh, AlarmWndCmdTarget Function.....	82
SelectAll, AlarmWndCmdTarget Function	83
SaveExtSettings, AlarmWndCmdTarget Function	83
Properties	84
AckAllBtnText, AlarmWndCmdTarget Property	84
AckSelBtnText, AlarmWndCmdTarget Property	84
AlarmFilter, AlarmWndCmdTarget Property	84
AlarmFilterMask, AlarmWndCmdTarget Property	85
AlarmFilterSeverity, AlarmWndCmdTarget Property	85
AlarmFilterSeverityCondition, AlarmWndCmdTarget Property	86
AreaFilter, AlarmWndCmdTarget Property	86
AutoLayout, AlarmWndCmdTarget Property	87
Autoscroll, AlarmWndCmdTarget Property.....	87
BlinkTime, AlarmWndCmdTarget Property.....	88
ButtonPos, AlarmWndCmdTarget Property	88
ButtonSize, AlarmWndCmdTarget Property	89
Clickable, AlarmWndCmdTarget Property	89
ExtSettingsFile, AlarmWndCmdTarget Property.....	89
FormatDateTime, AlarmWndCmdTarget Property	90
FormatDuration, AlarmWndCmdTarget Property	90
GetHistoryBtnTex, AlarmWndCmdTarget Property.....	91
GraphicButtons, AlarmWndCmdTarget Property	91
HasSpin, AlarmWndCmdTarget Property	92
HelpBtnText, AlarmWndCmdTarget Property	92
HisLogBackColor, AlarmWndCmdTarget Property	93
HisLogTextColor, AlarmWndCmdTarget Property.....	93
HorizontalSpin, AlarmWndCmdTarget Property	94
IncludeMilliseconds, AlarmWndCmdTarget Property	94
MaxOPCAEEEvents, AlarmWndCmdTarget Property	95
NetworkBackupServerName, AlarmWndCmdTarget Property	95
NetworkServer, AlarmWndCmdTarget Property	96
OPCAEServer, AlarmWndCmdTarget Property.....	96
RefreshTimePoll, AlarmWndCmdTarget Property	97
ResetAllSelBtnText, AlarmWndCmdTarget Property	97
ResetSelBtnText, AlarmWndCmdTarget Property	97
ScrollTime, AlarmWndCmdTarget Property.....	98
ShowAckAllBtn, AlarmWndCmdTarget Property	98
ShowAckSelBtn, AlarmWndCmdTarget Property	99
ShowDateTime, AlarmWndCmdTarget Property	99
ShowGetHistoryBtn, AlarmWndCmdTarget Property	100
ShowHelpBtn, AlarmWndCmdTarget Property.....	100
ShowHigherSeverity, AlarmWndCmdTarget Property	101
ShowResetAllBtn, AlarmWndCmdTarget Property	101
ShowResetSelBtn, AlarmWndCmdTarget Property	102
ShowSoundOnBtn, AlarmWndCmdTarget Property	102
SoundOnBtnText, AlarmWndCmdTarget Property.....	103
SpinSize, AlarmWndCmdTarget Property	103
SubItemAck, AlarmWndCmdTarget Property	104

SubItemAckPos, AlarmWndCmdTarget Property	104
SubItemAckWidth, AlarmWndCmdTarget Property	105
SubItemCondition, AlarmWndCmdTarget Property	105
SubItemConditionPos, AlarmWndCmdTarget Property	106
SubItemConditionWidth, AlarmWndCmdTarget Property	106
SubItemDuration, AlarmWndCmdTarget Property	107
SubItemDurationPos, AlarmWndCmdTarget Property	107
SubItemDurationWidth, AlarmWndCmdTarget Property	108
SubItemImage, AlarmWndCmdTarget Property	108
SubItemImagePos, AlarmWndCmdTarget Property	108
SubItemImageWidth, AlarmWndCmdTarget Property	109
SubItemOff, AlarmWndCmdTarget Property	109
SubItemOffPos, AlarmWndCmdTarget Property	110
SubItemOffWidth, AlarmWndCmdTarget Property	110
SubItemOn, AlarmWndCmdTarget Property	111
SubItemOnPos, AlarmWndCmdTarget Property	111
SubItemOnWidth, AlarmWndCmdTarget Property	112
SubItemReset, AlarmWndCmdTarget Property	112
SubItemResetPos, AlarmWndCmdTarget Property	113
SubItemResetWidth, AlarmWndCmdTarget Property	113
SubItemSeverity, AlarmWndCmdTarget Property	114
SubItemSeverityPos, AlarmWndCmdTarget Property	114
SubItemSeverityWidth, AlarmWndCmdTarget Property	115
SubItemStatus, AlarmWndCmdTarget Property	115
SubItemStatusPos, AlarmWndCmdTarget Property	115
SubItemStatusWidth, AlarmWndCmdTarget Property	116
SubItemText, AlarmWndCmdTarget Property	116
SubItemTextPos, AlarmWndCmdTarget Property	117
SubItemTextWidth, AlarmWndCmdTarget Property	117
SubItemTotalNumAck, AlarmWndCmdTarget Property	118
SubItemTotalNumAckPos, AlarmWndCmdTarget Property	118
SubItemTotalNumAckWidth, AlarmWndCmdTarget Property	119
SubItemTotalNumOn, AlarmWndCmdTarget Property	119
SubItemTotalNumOnPos, AlarmWndCmdTarget Property	120
SubItemTotalNumOnWidth, AlarmWndCmdTarget Property	120
SubItemTotalNumReset, AlarmWndCmdTarget Property	121
SubItemTotalNumResetWidth, AlarmWndCmdTarget Property	121
SubItemTotalTimeOn, AlarmWndCmdTarget Property	122
SubItemTotalTimeOnPos, AlarmWndCmdTarget Property	122
SubItemTotalTimeOnWidth, AlarmWndCmdTarget Property	123
1.14.4. ButtonCmdTarget..... 123	
Functions	123
GetCommandsInterfaceOnPressed, ButtonCmdTarget Function	123
GetCommandsInterfaceOnRelease, ButtonCmdTarget Function	124
GetCommandsInterfaceWhileDown, ButtonCmdTarget Function	124
GetShortcutText, ButtonCmdTarget Function	125
Properties.....	125
AsciiKeyShortcut, ButtonCmdTarget Property	125
Border, ButtonCmdTarget Property	126
ButtonStyle, ButtonCmdTarget Property	126
Clickable, ButtonCmdTarget Property	127
CommandStateVariable, ButtonCmdTarget Property	128
CommandType, ButtonCmdTarget Property	128
DisableCommandsOnCheckedState, ButtonCmdTarget Property	129
EnableShortcut, ButtonCmdTarget Property	129
ExecuteCommandsOnMouseMove, ButtonCmdTarget Property	130
ImageBtnChecked, ButtonCmdTarget Property	130
ImageBtnDisabled, ButtonCmdTarget Property	131
ImageBtnPressed, ButtonCmdTarget Property	131
ImageBtnReleased, ButtonCmdTarget Property	132
ImpulsiveTime, ButtonCmdTarget Property	132
OverlapImageText, ButtonCmdTarget Property	133
RadioBtnNumOptions, ButtonCmdTarget Property	133
RadioCheckBtnSize, ButtonCmdTarget Property	134
Round3DStyle, ButtonCmdTarget Property	135
ShowShortcut, ButtonCmdTarget Property	135
TriStateCentralZero, ButtonCmdTarget Property	136
VirtualKeyShortcut, ButtonCmdTarget Property	136
1.14.5. ChartWndCmdTarget..... 137	
Events	137
OnErrorRecordset, ChartWndCmdTarget Event.....	137
OnRecordsetMoveNext, ChartWndCmdTarget Event	137

OnRecordsetQueryEnd, ChartWndCmdTarget Event	137
OnRecordsetQueryStart, ChartWndCmdTarget Event.....	137
Functions.....	138
GetChartInterface, ChartWndCmdTarget Function.....	138
LoadExtSettings, ChartWndCmdTarget Function	138
RecalcLayout, ChartWndCmdTarget Function.....	139
SaveExtSettings, ChartWndCmdTarget Function	139
Properties	140
AddStackVariable, ChartWndCmdTarget Property	140
ArrayType, ChartWndCmdTarget Property.....	140
BackupLink, ChartWndCmdTarget Property	141
Border, ChartWndCmdTarget Property	141
Clickable, ChartWndCmdTarget Property	142
DataDefaultQuery, ChartWndCmdTarget Property.....	142
DataFilterBy, ChartWndCmdTarget Property.....	143
DataSortBy, ChartWndCmdTarget Property.....	144
ExtSettingsFile, ChartWndCmdTarget Property	145
ElevationVariable, ChartWndCmdTarget Property.....	145
LinkedDataLogger, ChartWndCmdTarget Property.....	146
NetworkBackupServerName, ChartWndCmdTarget Property.....	146
NetworkServerName, ChartWndCmdTarget Property	147
NumSamples, ChartWndCmdTarget Property	147
RotationVariable, ChartWndCmdTarget Property.....	148
Title1, ChartWndCmdTarget Property	148
Title2, ChartWndCmdTarget Property	149
Title3, ChartWndCmdTarget Property	149
Title4, ChartWndCmdTarget Property	150
Title5, ChartWndCmdTarget Property	150
Title6, ChartWndCmdTarget Property	151
Title7, ChartWndCmdTarget Property	151
Title8, ChartWndCmdTarget Property	152
Variable1, ChartWndCmdTarget Property.....	152
Variable2, ChartWndCmdTarget Property.....	153
Variable3, ChartWndCmdTarget Property.....	153
Variable4, ChartWndCmdTarget Property.....	154
Variable5, ChartWndCmdTarget Property.....	154
Variable6, ChartWndCmdTarget Property.....	155
Variable7, ChartWndCmdTarget Property.....	155
Variable8, ChartWndCmdTarget Property.....	156
<i>1.14.6. ClientRulesInterface</i>	<i>156</i>
Properties	156
ClientTimeout, ClientRulesInterface Property.....	156
DefaultClientUser, ClientRulesInterface Property.....	157
Name, ClientRulesInterface Property	157
PingTime, ClientRulesInterface Property	158
Priority, ClientRulesInterface Property	158
Protocol, ClientRulesInterface Property.....	159
RasStation, ClientRulesInterface Property	159
UseRASStation, ClientRulesInterface Property	160
<i>1.14.7. CommandAlarmCmdTarget.....</i>	<i>160</i>
Properties	160
Action, CommandAlarmCmdTarget Property.....	160
AreaFilter, CommandAlarmCmdTarget Property	162
PrintSettingsLandscape, CommandAlarmCmdTarget Property	163
PrintSettingsPageHeight, CommandAlarmCmdTarget Property.....	164
PrintSettingsPageWidth, CommandAlarmCmdTarget Property.....	165
PrintSettingsPortSettings, CommandAlarmCmdTarget Property	166
PrintSettingsPrinterName, CommandAlarmCmdTarget Property	167
PrintSettingsPrinterPort, CommandAlarmCmdTarget Property	169
PrintSettingsShowPrintDialog, CommandAlarmCmdTarget Property.....	170
Recipient, CommandAlarmCmdTarget Property.....	171
StatisticRptFile, CommandAlarmCmdTarget Property.....	173
StatisticRptReferenceDate, CommandAlarmCmdTarget Property.....	174
StatisticRptReferenceDuration, CommandAlarmCmdTarget Property	175
StatisticRptReferencePeriod, CommandAlarmCmdTarget Property.....	176
StatisticRptShowToolBar, CommandAlarmCmdTarget Property	177
StatisticRptShowTree, CommandAlarmCmdTarget Property	178
TextualRptBottomMargin, CommandAlarmCmdTarget Property.....	180
TextualRptLeftMargin, CommandAlarmCmdTarget Property.....	181
TextualRptMaxPages, CommandAlarmCmdTarget Property	182
TextualRptOutputFile, CommandAlarmCmdTarget Property.....	183
TextualRptRightMargin, CommandAlarmCmdTarget Property.....	184

TextualRptSQLQuery, CommandAlarmCmdTarget Property	185
TextualRptTemplateFile, CommandAlarmCmdTarget Property	186
TextualRptTopMargin, CommandAlarmCmdTarget Property	187
Functions	189
ConvertPeriodNumToString, CommandAlarmCmdTarget Function	189
GetCommandBaseInterface, CommandAlarmCmdTarget Function	190
1.14.8. <i>CommandBaseCmdTarget</i>	191
Properties	191
Type, CommandBaseCmdTarget Property	191
XmlSettings, CommandBaseCmdTarget Property	192
1.14.9. <i>CommandEventCmdTarget</i>	193
Property	193
Event, CommandEventCmdTarget Property	193
Functions	194
GetCommandBaseInterface, CommandEventCmdTarget Function	194
1.14.10. <i>CommandHelpCmdTarget</i>	195
Properties	195
Action, CommandHelpCmdTarget Property	195
Topic, CommandHelpCmdTarget Property	196
Functions	197
GetCommandBaseInterface, CommandHelpCmdTarget Function	197
1.14.11. <i>CommandLanguageCmdTarget</i>	198
Properties	198
Language, CommandLanguageCmdTarget Property	198
Functions	199
GetCommandBaseInterface, CommandLanguageCmdTarget Function	199
1.14.12. <i>CommandsListCmdTarget</i>	200
Functions	200
AddToTail, CommandsListCmdTarget Function	200
AddToHead, CommandsListCmdTarget Function	201
DiscardChanges, CommandsListCmdTarget Function	202
GetCommandInterfaceAtPos, CommandsListCmdTarget Function	203
GetCommandTypeAtPos, CommandsListCmdTarget Function	204
GetTotNumCommands, CommandsListCmdTarget Function	204
InsertAfter, CommandsListCmdTarget Function	205
InsertBefore, CommandsListCmdTarget Function	206
MoveToHead, CommandsListCmdTarget Function	207
MoveToTail, CommandsListCmdTarget Function	208
RemoveAll, CommandsListCmdTarget Function	209
RemoveAtPos, CommandsListCmdTarget Function	209
RemoveFromHead, CommandsListCmdTarget Function	210
RemoveFromTail, CommandsListCmdTarget Function	211
SaveChanges, CommandsListCmdTarget Function	212
SetAtPos, CommandsListCmdTarget Function	213
SwapCommands, CommandsListCmdTarget Function	214
1.14.13. <i>CommandMenuCmdTarget</i>	214
Functions	214
GetCommandBaseInterface, CommandMenuCmdTarget Function	214
Properties	215
Menu, CommandMenuCmdTarget Property	215
MeasureBtnText, TrendCmdTarget Property	216
XPos, CommandMenuCmdTarget Property	217
YPos, CommandMenuCmdTarget Property	218
1.14.14. <i>CommandReportCmdTarget</i>	219
Properties	219
Action, CommandReportCmdTarget Property	219
DLR, CommandReportCmdTarget Property	221
EmbeddedReportName, CommandReportCmdTarget Property	222
Height, CommandReportCmdTarget Property	223
Landscape, CommandReportCmdTarget Property	224
PageHeight, CommandReportCmdTarget Property	225
PageWidth, CommandReportCmdTarget Property	226
PrinterPort, CommandReportCmdTarget Property	227
PrinterName, CommandReportCmdTarget Property	229
PortSettings, CommandReportCmdTarget Property	230
RecipeCSVSeparator, CommandReportCmdTarget Property	232
Recipient, CommandReportCmdTarget Property	233
ReportExportFormat, CommandReportCmdTarget Property	234
ReportReferencePeriod, CommandReportCmdTarget Property	235
ReportShowFilterByDate, CommandReportCmdTarget Property	236
ReportShowToolbar, CommandReportCmdTarget Property	237
ReportShowTree, CommandReportCmdTarget Property	239

ShowPrintDialog, CommandReportCmdTarget Property	240
TextualRptBottomMargin, CommandReportCmdTarget Property.....	241
TextualRptLeftMargin, CommandReportCmdTarget Property	242
TextualRptMaxPages, CommandReportCmdTarget Property	243
TextualRptOutputFile, CommandReportCmdTarget Property	244
TextualRptRightMargin, CommandReportCmdTarget Property.....	245
TextualRptSQLQuery, CommandReportCmdTarget Property	246
TextualRptTemplateFile, CommandReportCmdTarget Property	248
TextualRptTopMargin, CommandReportCmdTarget Property.....	249
Width, CommandReportCmdTarget Property	250
XPos, CommandReportCmdTarget Property.....	251
YPos, CommandReportCmdTarget Property	252
Functions.....	253
GetCommandBaseInterface, CommandReportCmdTarget Function	253
1.14.15. CommandScriptCmdTarget	254
Properties	254
Action, CommandScriptCmdTarget Property.....	254
MoreInstanceAllowed, CommandScriptCmdTarget Property.....	255
Parameters, CommandScriptCmdTarget Property.....	257
Script, CommandScriptCmdTarget Property	258
SynchroTimeout, CommandScriptCmdTarget Property	259
Functions.....	260
GetCommandBaseInterface, CommandScriptCmdTarget Function	260
1.14.16. CommandSynopticCmdTarget.....	261
Properties	261
Action, CommandSynopticCmdTarget Property	261
Height, CommandSynopticCmdTarget Property	262
KeepProportionsOnPrint, CommandSynopticCmdTarget Property	263
Monitor, CommandSynopticCmdTarget Property	264
ParameterFile, CommandSynopticCmdTarget Property	265
PrintBottomMargin, CommandSynopticCmdTarget Property	267
PrintLeftMargin, CommandSynopticCmdTarget Property	268
PrintPageHeight, CommandSynopticCmdTarget Property	269
PrintPageWidth, CommandSynopticCmdTarget Property	270
PrintRightMargin, CommandSynopticCmdTarget Property	271
PrintTopMargin, CommandSynopticCmdTarget Property	272
ResizableBorder, CommandSynopticCmdTarget Property	273
ShowBorder, CommandSynopticCmdTarget Property.....	274
ShowCaption, CommandSynopticCmdTarget Property	275
ShowMaximizedBtn, CommandSynopticCmdTarget Property	276
ShowMinimizedBtn, CommandSynopticCmdTarget Property	278
ShowSystemMenu, CommandSynopticCmdTarget Property	279
SynopticName, CommandSynopticCmdTarget Property	280
Width, CommandSynopticCmdTarget Property	281
XPos, CommandSynopticCmdTarget Property.....	282
YPos, CommandSynopticCmdTarget Property	283
Functions.....	284
GetCommandBaseInterface, CommandSynopticCmdTarget Function	284
1.14.17. CommandSystemCmdTarget	285
Properties	285
Action, CommandSystemCmdTarget Property	285
CommandLine, CommandSystemCmdTarget Property	286
Timeout, CommandSystemCmdTarget Property	287
WorkingPath, CommandSystemCmdTarget Property	289
Functions.....	290
GetCommandBaseInterface, CommandSystemCmdTarget Function.....	290
1.14.18. CommandUsersCmdTarget	291
Properties	291
Action, CommandUsersCmdTarget Property	291
Level, CommandUsersCmdTarget Property.....	292
Functions.....	293
GetCommandBaseInterface, CommandUsersCmdTarget Function	293
1.14.19. CommandVariableCmdTarget.....	294
Functions.....	294
GetCommandBaseInterface, CommandVariableCmdTarget Function	294
Properties	295
Action, CommandVariableCmdTarget Property.....	295
MaxChar, CommandVariableCmdTarget Property	296
MaxValue, CommandVariableCmdTarget Property	297
MinValue, CommandVariableCmdTarget Property.....	298
MoveToVariable, CommandVariableCmdTarget Property	299
PasswordStyle, CommandVariableCmdTarget Property	300

StrobeTime, CommandVariableCmdTarget Property	301
Value, CommandVariableCmdTarget Property	303
Variable, CommandVariableCmdTarget Property.....	304
1.14.20. DBVariableCmdTarget	305
1.15. UTILIZZO DELLA DBVARIABLECMDTARGET	305
Functions	305
CreateNewVar, DBVariableCmdTarget Function	305
GetDynamicVariable, DBVariableCmdTarget Function.....	306
GetNumDynTag, DBVariableCmdTarget Function	307
GetNumStructDefinitions, DBVariableCmdTarget Function	307
GetNumVariables, DBVariableCmdTarget Function	307
GetRealTimeDBADODConn, DBVariableCmdTarget Function.....	308
GetStructureDefinitionsList, DBVariableCmdTarget Function	309
GetTraceDBADODConn, DBVariableCmdTarget Function	309
GetTraceDBDSNConnectionString, DBVariableCmdTarget Function	310
GetVariableAddressInfo, DBVariableCmdTarget Function	310
GetVariableDescription, DBVariableCmdTarget Function	311
GetVariableObject, DBVariableCmdTarget Function.....	312
GetVariableSize, DBVariableCmdTarget Function	312
GetVariableValue, DBVariableCmdTarget Function	313
GetXMLSettings, DBVariableCmdTarget Function	313
IsFirstDBInstance, DBVariableCmdTarget Function	313
IsValidVariable, DBVariableCmdTarget Function	314
PurgeDynTag, DBVariableCmdTarget Function	314
QualityOf, DBVariableCmdTarget Function	315
SetVariableValue, DBVariableCmdTarget Function	315
VariableInUse, DBVariableCmdTarget Function.....	316
Properties.....	317
EnableInUseVarMng, DBVariableCmdTarget Property	317
EnableNTSecurityOPCServerTag, DBVariableCmdTarget Property	317
OPCServerEnableAEAck, DBVariableCmdTarget Property	318
OPCServerMinImpersonationLevel, DBVariableCmdTarget Property.....	318
OPCServerShutdownClientsTimeout, DBVariableCmdTarget Property	319
PurgeDynTagTimer, DBVariableCmdTarget Property	319
TraceDBChangerColName, DBVariableCmdTarget Property	319
TraceDBDefVarCharPrecision, DBVariableCmdTarget Property	320
TraceDBDsn, DBVariableCmdTarget Property	320
TraceDBLocalTimeColName, DBVariableCmdTarget Property	321
TraceDBMaxCacheBeforeFlush, DBVariableCmdTarget Property	322
TraceDBMaxError, DBVariableCmdTarget Property.....	322
TraceDBMaxNumberTrans, DBVariableCmdTarget Property.....	322
TraceDBMSecColName, DBVariableCmdTarget Property	323
TraceDBQualityColName, DBVariableCmdTarget Property	323
TraceDBRecycleDBConnection, DBVariableCmdTarget Property	324
TraceDBTimeColName, DBVariableCmdTarget Property	324
TraceDBUser, DBVariableCmdTarget Property	325
TraceDBUserColName, DBVariableCmdTarget Property	325
TraceDBValueAfterColName, DBVariableCmdTarget Property	326
TraceDBValueBeforeColName, DBVariableCmdTarget Property	326
TraceDBValueColName, DBVariableCmdTarget Property	327
TraceDBTimeStampColName, DBVariableCmdTarget Property	327
TraceDBVarNameColName, DBVariableCmdTarget Property	328
TraceDBVarDescNameColName, DBVariableCmdTarget Property	328
TraceDBVarGroupNameColName, DBVariableCmdTarget Property	329
TraceDBVarNameColName, DBVariableCmdTarget Property	330
TraceUseIMDB, DBVariableCmdTarget Property.....	330
UseSharedDynTag, DBVariableCmdTarget Property	331
1.15.1. DBVarObjCmdTarget.....	331
Functions	331
Description, DBVarObjCmdTarget Function	331
GetAccessLevelReadMask, DBVarObjCmdTarget Function	332
GetAccessLevelWriteMask, DBVarObjCmdTarget Function.....	332
GetAddress, DBVarObjCmdTarget Function.....	333
GetAlarmListName, DBVarObjCmdTarget Function.....	333
GetAlarmListNum, DBVarObjCmdTarget Function	334
GetAlarmObject, DBVarObjCmdTarget Function.....	334
GetAreaType, DBVarObjCmdTarget Function.....	335
GetBitNumber, DBVarObjCmdTarget Function	335
GetDataLoggerListNum, DBVarObjCmdTarget Function	336
GetDataLoggerName, DBVarObjCmdTarget Function.....	336
GetEventListName, DBVarObjCmdTarget Function	337

GetEventListNum, DBVarObjCmdTarget Function	337
GetEventObject, DBVarObjCmdTarget Function	338
GetInitialTimeInUse, DBVarObjCmdTarget Function	338
GetInUseCount, DBVarObjCmdTarget Function	339
GetInUseObjectAt, DBVarObjCmdTarget Function	339
GetInUseObjectNameAt, DBVarObjCmdTarget Function	340
GetLastTimeInUse, DBVarObjCmdTarget Function.....	341
GetMemberObjectFromIndex, DBVarObjCmdTarget Function.....	341
GetMemberObjectFromName, DBVarObjCmdTarget Function.....	342
GetName, DBVarObjCmdTarget Function	342
GetNumObjectsInHeap, DBVarObjCmdTarget Function	343
GetStructName, DBVarObjCmdTarget Function.....	343
GetStructParentObject, DBVarObjCmdTarget Function.....	344
GetTimeNotInUse, DBVarObjCmdTarget Function	344
GetTimeStamp, DBVarObjCmdTarget Function	345
GetTimeStampMS, DBVarObjCmdTarget Function	345
GetType, DBVarObjCmdTarget Function	346
GetXMLSettings, DBVarObjCmdTarget Function	346
IsOPCServerEnabled, DBVarObjCmdTarget Function	347
IsOPCServerOnRequest, DBVarObjCmdTarget Function	347
IsShared, DBVarObjCmdTarget Function.....	348
IsValid, DBVarObjCmdTarget Function.....	348
ResetStatisticData, DBVarObjCmdTarget Function	349
SetOPCServerEnabled, DBVarObjCmdTarget Function	349
SetStructName, DBVarObjCmdTarget Function	350
SetTimeStamp, DBVarObjCmdTarget Function	351
SetTimeStampFromDate, DBVarObjCmdTarget Function.....	351
SetType, DBVarObjCmdTarget Function.....	352
Properties	352
AviFileProp, DBVarObjCmdTarget Property.....	352
BGColorProp, DBVarObjCmdTarget Property	353
BlinkProp, DBVarObjCmdTarget Property	353
BmpFileProp, DBVarObjCmdTarget Property.....	354
CloseBitString, DBVarObjCmdTarget Property	354
DynamicSettings, DBVarObjCmdTarget Property.....	355
EnableFactor, DBVarObjCmdTarget Property	355
EnableNetworkServer, DBVarObjCmdTarget Property	356
EnableScalingFactor, DBVarObjCmdTarget Property	356
EngineeringUnit, DBVarObjCmdTarget Property	357
FactorGain, DBVarObjCmdTarget Property	357
FactorOffset, DBVarObjCmdTarget Property	357
FGColorProp, DBVarObjCmdTarget Property.....	358
Group, DBVarObjCmdTarget Property.....	358
HtmlFileProp DBVarObjCmdTarget Property	359
InheritQuality, DBVarObjCmdTarget Property.....	359
InUse, DBVarObjCmdTarget Property	360
InverseFactor, DBVarObjCmdTarget Property	360
InverseScaling, DBVarObjCmdTarget Property.....	361
LastChangeComment, DBVarObjCmdTarget Property	361
MapRealTimeODBCUpdateQuality, DBVarObjCmdTarget Property.....	362
MapRealTimeToDB, DBVarObjCmdTarget Property	362
MapRealTimeToDBMode, DBVarObjCmdTarget Property.....	363
MapRealTimeToDBRefreshTime, DBVarObjCmdTarget Property.....	363
NetworkClientEnable, DBVarObjCmdTarget Property	364
NetworkClientMode, DBVarObjCmdTarget Property	364
NetworkClientServerName, DBVarObjCmdTarget Property	365
NetworkClientUpdateQuality, DBVarObjCmdTarget Property.....	365
NetworkServerEnable, DBVarObjCmdTarget Property	366
NetworkServerIsWritable, DBVarObjCmdTarget Property	366
OPCGroupName, DBVarObjCmdTarget Property.....	367
OPCServerAccessRights,DBVarObjTarget Property	367
OpenBitString, DBVarObjCmdTarget Property.....	368
Quality, DBVarObjCmdTarget Property	368
ScaleMax, DBVarObjCmdTarget Property	369
ScaleMin, DBVarObjCmdTarget Property.....	369
ScaleRawMax, DBVarObjCmdTarget Property	370
ScaleRawMin, DBVarObjCmdTarget Property	370
SharedRetentive, DBVarObjCmdTarget Property	371
SndFileProp DBVarObjCmdTarget Property.....	371
StatisticData DBVarObjCmdTarget Property	372
StatisticDataAverage, DBVarObjCmdTarget Property	372
StatisticDataMaxValue, DBVarObjCmdTarget Property.....	373

StatisticDataMinValue, DBVarObjCmdTarget Property	373
StatisticDataNumSamples, DBVarObjCmdTarget Property	374
StatisticTotalTimeOn, DBVarObjCmdTarget Property.....	374
TraceAddMsgLog, DBVarObjCmdTarget Property	375
TraceAddDescCol, DBVarObjCmdTarget Property.....	375
TraceAddGroupCol, DBVarObjCmdTarget Property	376
TraceComment, DBVarObjCmdTarget Property.....	376
TraceEnable, DBVarObjCmdTarget Property	377
TraceEnableFromToTime, DBVarObjCmdTarget Property	377
TraceFromTime, DBVarObjCmdTarget Property	378
TraceMaxAgeDays, DBVarObjCmdTarget Property	378
TraceMaxAgeHours, DBVarObjCmdTarget Property	379
TraceMaxAgeMins, DBVarObjCmdTarget Property	379
TraceTableName, DBVarObjCmdTarget Property	380
TraceToTime, DBVarObjCmdTarget Property	380
Value, DBVarObjCmdTarget Property	380
1.15.2. DisplayEditCmdTarget	381
Functions	381
GetComboListInterface, DisplayEditCmdTarget Function.....	381
IsCombo, DisplayEditCmdTarget Function.....	381
IsSpin, DisplayEditCmdTarget Function	382
LoadExtSettings, DisplayEditCmdTarget Function	382
RecalcLayout, DisplayEditCmdTarget Function.....	383
SaveExtSettings, DisplayEditCmdTarget Function.....	383
Properties.....	384
EditingPassword, DisplayEditCmdTarget Property	384
ExtSettingsFile, DisplayEditCmdTarget Property	384
FormatData, DisplayEditCmdTarget Property	385
FormatVariable, DisplayEditCmdTarget Property.....	385
HasSpin, DisplayEditCmdTarget Property.....	386
InvertSelection, DisplayEditCmdTarget Property.....	386
IsEditable, DisplayEditCmdTarget Property	387
IsSpinHoriz, DisplayEditCmdTarget Property	387
PromptPad, DisplayEditCmdTarget Property.....	387
ScaleUnit, DisplayEditCmdTarget Property	388
SpinStep, DisplayEditCmdTarget Property	388
TimeToWaitToIncrease, DisplayEditCmdTarget Property	388
ValMax, DisplayEditCmdTarget Property	389
ValMin, DisplayEditCmdTarget Property.....	389
ValueToDisplay, DisplayEditCmdTarget Property.....	390
Variable, DisplayEditCmdTarget Property.....	391
VariableMax, DisplayEditCmdTarget Property	391
VariableMin, DisplayEditCmdTarget Property.....	391
1.15.3. DLRCmdTarget.....	392
Functions	392
GetADODConn, DLRCmdTarget Function	392
GetColumn, DLRCmdTarget Function	393
GetColumnNameList, DLRCmdTarget Function	393
GetDNSConnectionString, DLRCmdTarget Function.....	394
GetNextTickLocalTime, DLRCmdTarget Function.....	394
GetNextTickTime, DLRCmdTarget Function	395
Properties.....	395
ActivateVariable, DLRCmdTarget Property	395
CRWReportFile, DLRCmdTarget Property	396
DeleteVariable, DLRCmdTarget Property.....	396
DSN, DLRCmdTarget Property.....	397
DurationDays, DLRCmdTarget Property	397
DurationHours, DLRCmdTarget Property.....	398
DurationMinutes, DLRCmdTarget Property	398
Enabled, DLRCmdTarget Property	399
EnableTimeFrom, DLRCmdTarget Property	399
EnableTimeFromTo, DLRCmdTarget Property	400
EnableTimeTo, DLRCmdTarget Property	400
EnableTimeVariable, DLRCmdTarget Property	401
ExecuteVariable, DLRCmdTarget Property	401
Filter, DLRCmdTarget Property.....	402
FilterVariable, DLRCmdTarget Property	402
InsertVariable, DLRCmdTarget Property	402
IsRecipe, DLRCmdTarget Property	403
LocalTimeColName, DLRCmdTarget Property	403
MaxCacheBeforeFlush, DLRCmdTarget Property	404
MaxError, DLRCmdTarget Property	404

MaxNumberTrans, DLRCmdTarget Property	405
MoveFirstVariable, DLRCmdTarget Property	405
MoveLastVariable, DLRCmdTarget Property	406
MoveNextVariable, DLRCmdTarget Property	406
MovePrevVariable, DLRCmdTarget Property	407
MSecColName, DLRCmdTarget Property	407
Name, DLRCmdTarget Property	408
PrintVariable, DLRCmdTarget Property	408
Query, DLRCmdTarget Property	409
QueryVariable, DLRCmdTarget Property	409
ReadVariable, DLRCmdTarget Property	410
ReasonColName, DLRCmdTarget Property	410
RecipeIndexName, DLRCmdTarget Property	411
RecordOnChange, DLRCmdTarget Property	411
RecordOnChangeDeadBand, DLRCmdTarget Property	412
RecordOnChangeDeadBandPercent, DLRCmdTarget Property	413
RecordOnChangeEnableDeadBand, DLRCmdTarget Property	413
RecordOnlyWhenQualityGood, DLRCmdTarget Property	413
RecordOnTime, DLRCmdTarget Property	414
RecordOnVariable, DLRCmdTarget Property	414
RecVariable, DLRCmdTarget Property	415
RecycleDBConnection, DLRCmdTarget Property	415
ResetVariable, DLRCmdTarget Property	416
Sort, DLRCmdTarget Property	416
SortVariable, DLRCmdTarget Property	417
StatusVariable, DLRCmdTarget Property	417
TableName, DLRCmdTarget Property	418
TimeColName, DLRCmdTarget Property	418
TimeRecHour, DLRCmdTarget Property	419
TimeRecMin, DLRCmdTarget Property	419
TimeRecMSec, DLRCmdTarget Property	420
TimeRecSec, DLRCmdTarget Property	420
UserColName, DLRCmdTarget Property	420
UserName, DLRCmdTarget Property	421
VarCharsMax, DLRCmdTarget Property	421
1.15.4. DLRColumnCmdTarget.....	422
Properties	422
AddNumUpdatesCol, DLRColumnCmdTarget Property	422
AddQualityColumn, DLRColumnCmdTarget Property	422
Name, DLRColumnCmdTarget Property	423
NumUpdatesColumnName, DLRColumnCmdTarget Property	423
QualityColumnName, DLRColumnCmdTarget Property	424
RecipeIndex, DLRColumnCmdTarget Property	425
RecipeTempVariable, DLRColumnCmdTarget Property	425
RecordType, DLRColumnCmdTarget Property	426
StatisticAverageValue, DLRColumnCmdTarget Property	426
StatisticMaxValue, DLRColumnCmdTarget Property	427
StatisticMinValue, DLRColumnCmdTarget Property	427
StatisticNumUpdates, DLRColumnCmdTarget Property	428
Variable, DLRColumnCmdTarget Property	428
1.15.5. DLRWndCmdTarget.....	429
Events.....	429
OnFilter, DLRWndCmdTarget Event.....	429
OnPrint, DLRWndCmdTarget Event	429
OnRefresh, DLRWndCmdTarget Event	429
Functions.....	430
EditCopy, DLRWndCmdTarget Function.....	430
EditLayout, DLRWndCmdTarget Function	430
LoadExtSettings, DLRWndCmdTarget Function	431
RecalcLayout, DLRWndCmdTarget Function.....	431
Refresh, DLRWndCmdTarget Function	431
SaveExtSettings, DLRWndCmdTarget Function	432
Properties	432
AutoLayout, DLRWndCmdTarget Property	432
ButtonPos, DLRWndCmdTarget Property	433
ButtonSize, DLRWndCmdTarget Property	434
Clickable, DLRWndCmdTarget Property	434
DLR, DLRWndCmdTarget Property	435
ExtSettingsFile, DLRWndCmdTarget Property	435
FilterBtnText, DLRWndCmdTarget Property	436
FilterFromDate, DLRWndCmdTarget Property	436
FilterToDate, DLRWndCmdTarget Property	437

FilterUser, DLRWndCmdTarget Property	437
GraphicButtons, DLRWndCmdTarget Property	438
IncludeMilliseconds, DLRWndCmdTarget Property	438
MaxCount, DLRWndCmdTarget Property	439
NetworkBackupServerName, DLRWndCmdTarget Property	439
NetworkServerName, DLRWndCmdTarget Property	440
PrintBtnText, DLRWndCmdTarget Property	440
Project, DLRWndCmdTarget Property	441
Query, DLRWndCmdTarget Property	441
RefreshBtnText, DLRWndCmdTarget Property	442
ShowFilterBtn, DLRWndCmdTarget Property	442
ShowPrintBtn, DLRWndCmdTarget Property	443
ShowRefreshBtn, DLRWndCmdTarget Property	444
SortBy, DLRWndCmdTarget Property	444
SubItemReason, DLRWndCmdTarget Property	445
SubItemReasonPos, DLRWndCmdTarget Property	445
SubItemReasonWidth, DLRWndCmdTarget Property	446
SubItemTime, DLRWndCmdTarget Property	446
SubItemTimePos, DLRWndCmdTarget Property	447
SubItemTimeWidth, DLRWndCmdTarget Property	447
SubItemUser, DLRWndCmdTarget Property	448
SubItemUserPos, DLRWndCmdTarget Property	448
SubItemUserWidth, DLRWndCmdTarget Property	449
<i>1.15.6. DrawCmdTarget.....</i>	<i>449</i>
Events	449
Click, Generic Event	449
DbClick, Generic Event	449
KeyDown, Generic Event	450
KeyPress, Generic Event	450
KeyUp, Generic Event	450
KillFocus, Generic Event	450
MouseDown, Generic Event	451
MouseMove, Generic Event	451
MouseUp, Generic Event	451
OnChange, Generic Event	452
OnChangeExecutionCanceled, Generic Event	452
OnChangeExecutionToPromoter, Generic Event	453
OnExecutionPending, Generic Event	453
OnFireExecution, Generic Event	453
OnFireSynapse, Generic Event	453
OnPostPaint, Generic Event	454
OnPrePaint, Generic Event	454
OnTextChanged, Generic Event	454
OnTextChanging, Generic Event	454
OnTimer, Generic Event	455
OnToolTip, Generic Event	455
SetFocus, Generic Event	455
SymbolLoading, Generic Event	455
SymbolUnloading, Generic Event	456
Functions	456
AddPolyPoint, DrawCmdTarget Function	456
CloseThisSynoptic, DrawCmdTarget Function	456
ConvertAngleToPoint, DrawCmdTarget Function	457
ConvertPointToAngle, DrawCmdTarget Function	458
CursorPosToObjectPos, DrawCmdTarget Function	458
DeletePolyPoint, DrawCmdTarget Function	459
EnableVariableEvent, DrawCmdTarget Function	459
GetActiveXObject, DrawCmdTarget Function	460
GetAlias, DrawCmdTarget Function	461
GetAliasListName, DrawCmdTarget Function	461
GetAliasListValue, DrawCmdTarget Function	461
GetCommandsInterfaceOnRelease, DrawCmdTarget Function	462
GetConnectorObjectConnected, DrawCmdTarget Function	462
GetContainerObject, DrawCmdTarget Function	463
GetCursorPos, DrawCmdTarget Function	463
GetCursorPosInObject, DrawCmdTarget Function	464
GetGaugeObject, DrawCmdTarget Function	464
GetNumConnectionsOnSynapse, DrawCmdTarget Function	465
GetNumPolyPoint, DrawCmdTarget Function	465
GetNumSynapsis, DrawCmdTarget Function	466
GetObjectConnectedOnSynapse, DrawCmdTarget Function	466
GetObjectInterface, DrawCmdTarget Function	467

GetOnScreenPosition, DrawCmdTarget Function	467
GetPolyPointOnScreenX, DrawCmdTarget Function	468
GetPolyPointOnScreenY, DrawCmdTarget Function.....	468
GetSubGaugeObject, DrawCmdTarget Function	469
GetSubObject, DrawCmdTarget Function	469
GetSubTrendObject, DrawCmdTarget Function	470
GetSynapseName, DrawCmdTarget Function.....	471
GetSynapsePoint, DrawCmdTarget Function.....	471
GetSynopticObject, DrawCmdTarget Function	472
GetTrendObject, DrawCmdTarget Function	472
GetUniqueObjectID, DrawCmdTarget Function	473
GetXMLSettings, DrawCmdTarget Function	473
HasSynapsis, DrawCmdTarget Function.....	474
InflateObject, DrawCmdTarget Function	474
IsCursorOnObject, DrawCmdTarget Function.....	474
IsGlobalObjectName, DrawCmdTarget Function	475
IsSynapseConnected, DrawCmdTarget Function	475
LoadExtSettings, DrawCmdTarget Function.....	476
MoveObject, DrawCmdTarget Function	476
OffsetObject, DrawCmdTarget Function	477
PolyPointX, DrawCmdTarget Function	477
PolyPointY, DrawCmdTarget Function	478
PrintThisSynoptic, DrawCmdTarget Function	478
Prop, DrawCmdTarget Function	479
RemoveAlias, DrawCmdTarget Function.....	479
RemoveAllAliases, DrawCmdTarget Function	480
ResetColors, DrawCmdTarget Function	480
SaveExtSettings, DrawCmdTarget Function.....	480
ScaleObject, DrawCmdTarget Function	481
SetAlias, DrawCmdTarget Function	481
ShowPropList, DrawCmdTarget Function.....	482
SynapsePassExecution, DrawCmdTarget Function.....	482
ZOrderMoveBack, DrawCmdTarget Function.....	483
ZOrderMoveForward, DrawCmdTarget Function	483
ZOrderMoveToBack, DrawCmdTarget Function	484
ZOrderMoveToFront, DrawCmdTarget Function.....	484
Properties	484
AdaptFontSize, DrawCmdTarget Property	484
AlignFont, DrawCmdTarget Property	485
AlignFontOffsetX, DrawCmdTarget Property	486
AlignFontOffsetY, DrawCmdTarget Property	486
AntialiasingFont, DrawCmdTarget Property	487
AutoRepeatClick, DrawCmdTarget Property.....	487
BackBrushPattern, DrawCmdTarget Property.....	488
BackBrushVisible, DrawCmdTarget Property.....	488
BackColor, DrawCmdTarget Property.....	489
BackColorBrightness, DrawCmdTarget Property	489
BackColorHue, DrawCmdTarget Property	490
BackColorSaturation, DrawCmdTarget Property	490
BitmapAlignment, DrawCmdTarget Property	491
BitmapID, DrawCmdTarget Property	492
BitmapKeepAspectRatio, DrawCmdTarget Property	492
BitmapOffsetX, DrawCmdTarget Property	493
BitmapOffsetY, DrawCmdTarget Property.....	493
BitmapStretched, DrawCmdTarget Property	494
BitmapTransparent, DrawCmdTarget Property.....	494
BitmapTransparentColor, DrawCmdTarget Property	495
BorderType, DrawCmdTarget Property	495
CenterRotation, DrawCmdTarget Property	496
DefStructName, DrawCmdTarget Property	496
DefStructNameAbsolute, DrawCmdTarget Property	497
DrawingState, DrawCmdTarget Property.....	497
DrawingStateShadow, DrawCmdTarget Property.....	498
EdgeColor, DrawCmdTarget Property	498
EmbeddedSynoptic, DrawCmdTarget Property.....	499
EnableExecution, DrawCmdTarget Property.....	499
EnableVariable, DrawCmdTarget Property	500
ExtSettingsFile, DrawCmdTarget Property.....	500
FillBrushPattern, DrawCmdTarget Property	501
FillColor, DrawCmdTarget Property	501
FillingMode, DrawCmdTarget Property.....	502
FillingPercent, DrawCmdTarget Property	502

Font3D, DrawCmdTarget Property	503
FontBold, DrawCmdTarget Property	503
FontCharSet, DrawCmdTarget Property	504
FontEscapement, DrawCmdTarget Property	505
FontHeight, DrawCmdTarget Property	505
FontItalic, DrawCmdTarget Property	505
FontName, DrawCmdTarget Property	506
GradientColor, DrawCmdTarget Property	506
GradientFill, DrawCmdTarget Property	507
Height, DrawCmdTarget Property	507
Hilite, DrawCmdTarget Property	508
LineArrowHeight, DrawCmdTarget Property	508
LineArrowType, DrawCmdTarget Property	509
LineEndingX, DrawCmdTarget Property	509
LineEndingY, DrawCmdTarget Property	510
LineStartingX, DrawCmdTarget Property	510
LineStartingY, DrawCmdTarget Property	511
LinkedTextFormat, DrawCmdTarget Property	511
LinkedTextFormatVariable, DrawCmdTarget Property	512
Look3D, DrawCmdTarget Property	512
Look3DPressed, DrawCmdTarget Property	513
MetaFile, DrawCmdTarget Property	513
MouseCapture, DrawCmdTarget Property	513
ObjectName, DrawCmdTarget Property	514
ObjectPublicName, DrawCmdTarget Property	514
PenColorBrightness, DrawCmdTarget Property	515
PenColorHue, DrawCmdTarget Property	515
PenColorSaturation, DrawCmdTarget Property	516
PenStyle, DrawCmdTarget Property	516
PenVisible, DrawCmdTarget Property	517
PenWidth, DrawCmdTarget Property	518
Rotation, DrawCmdTarget Property	518
Shadow, DrawCmdTarget Property	519
ShadowColor, DrawCmdTarget Property	519
ShadowXOffset, DrawCmdTarget Property	519
ShadowYOffset, DrawCmdTarget Property	520
ShowFocusRectangle, DrawCmdTarget Property	520
ShowHiliteRectangle, DrawCmdTarget Property	521
SynapseBackColor, DrawCmdTarget Property	521
SynapseValue, DrawCmdTarget Property	522
SynapseValueFromID, DrawCmdTarget Property	522
SynapsisVisible, DrawCmdTarget Property	523
StatusVariable, DrawCmdTarget Property	523
Title, DrawCmdTarget Property	524
ToolTip, DrawCmdTarget Property	524
Transparency, DrawCmdTarget Property	525
VariableBackColor, DrawCmdTarget Property	525
VariableBitmapList, DrawCmdTarget Property	525
VariableComposedMovement, DrawCmdTarget Property	526
VariableEdgeColor, DrawCmdTarget Property	526
VariableEndingX, DrawCmdTarget Property	527
VariableEndingY, DrawCmdTarget Property	527
VariableFillColor, DrawCmdTarget Property	527
VariableFilling, DrawCmdTarget Property	528
VariableLinkedText, DrawCmdTarget Property	528
VariableMoveX, DrawCmdTarget Property	529
VariableMoveY, DrawCmdTarget Property	529
VariableRotation, DrawCmdTarget Property	530
VariableScaling, DrawCmdTarget Property	530
VariableStartingX, DrawCmdTarget Property	530
VariableStartingY, DrawCmdTarget Property	531
VariableVisible, DrawCmdTarget Property	531
VariableXRotationCenter, DrawCmdTarget Property	532
VariableYRotationCenter, DrawCmdTarget Property	532
Visible, DrawCmdTarget Property	533
Width, DrawCmdTarget Property	533
Xpos, DrawCmdTarget Property	533
XRotationCenter, DrawCmdTarget Property	534
Ypos, DrawCmdTarget Property	534
YRotationCenter, DrawCmdTarget Property	535
1.15.7. EventCmdTarget	535
Functions	535

GetXMLSettings, EventCmdTarget Function.....	535
Properties	536
CommandList, EventCmdTarget Property	536
Condition, EventCmdTarget Property	536
Enable, EventCmdTarget Property	537
EnableVariable, EventCmdTarget Property	537
Name, EventCmdTarget Property	538
Value, EventCmdTarget Property	538
Variable, EventCmdTarget Property	539
<i>1.15.8. GaugeCmdTarget</i>	<i>539</i>
Functions.....	539
LoadExtSettings, GaugeCmdTarget Function	539
SaveExtSettings, GaugeCmdTarget Function	540
Properties	540
BarBackColor, GaugeCmdTarget Property	540
BarBias, GaugeCmdTarget Property	541
BarBorder, GaugeCmdTarget Property	541
BarFillColor, GaugeCmdTarget Property	542
BarVisible, GaugeCmdTarget Property	542
CenterPos, GaugeCmdTarget Property	543
ColorWarningZone, GaugeCmdTarget Property	543
EnableWarningZone, GaugeCmdTarget Property	544
EndAngle, GaugeCmdTarget Property	544
EndWarningZone, GaugeCmdTarget Property	545
ExtSettingsFile, GaugeCmdTarget Property	545
FontHeightScale, GaugeCmdTarget Property	546
FontNameScale, GaugeCmdTarget Property	546
FormatVariable, GaugeCmdTarget Property	547
GaugeMaxVariable, GaugeCmdTarget Property	547
GaugeMinVariable, GaugeCmdTarget Property	548
GaugeType, GaugeCmdTarget Property	548
GaugeVariable, GaugeCmdTarget Property	549
GaugeWarningZoneEndVariable, GaugeCmdTarget Property	549
GaugeWarningZoneStartVariable, GaugeCmdTarget Property	550
GeneralGap, GaugeCmdTarget Property	550
InvertScale, GaugeCmdTarget Property	551
KnobBorder, GaugeCmdTarget Property	551
KnobColor, GaugeCmdTarget Property	552
LabelEvery, GaugeCmdTarget Property	552
MaxValue, GaugeCmdTarget Property	553
MinValue, GaugeCmdTarget Property	553
NeedleBorder, GaugeCmdTarget Property	554
NeedleBorderColor, GaugeCmdTarget Property	555
NeedleColor, GaugeCmdTarget Property	555
NeedleLength, GaugeCmdTarget Property	556
NeedleShadow, GaugeCmdTarget Property	556
NeedleShadowColor, GaugeCmdTarget Property	557
NeedleVisible, GaugeCmdTarget Property	557
NeedleWidth, GaugeCmdTarget Property	558
ScaleColor, GaugeCmdTarget Property	559
ScaleFormat, GaugeCmdTarget Property	559
ScaleMajorDiv, GaugeCmdTarget Property	560
ScaleMinorDiv, GaugeCmdTarget Property	560
ScaleRightBottom, GaugeCmdTarget Property	561
ScaleUnit, GaugeCmdTarget Property	561
ScaleVisible, GaugeCmdTarget Property	562
SliderBorder, GaugeCmdTarget Property	562
SliderColor, GaugeCmdTarget Property	563
SliderVisible, GaugeCmdTarget Property	563
StartAngle, GaugeCmdTarget Property	564
StartWarningZone, GaugeCmdTarget Property	565
Title, GaugeCmdTarget Property	565
TitleVisible, GaugeCmdTarget Property	565
<i>1.15.9. Generic Events.....</i>	<i>566</i>
Click, Generic Event	566
DbClick, Generic Event	566
KeyDown, Generic Event	566
KeyPress, Generic Event	567
KeyUp, Generic Event	567
KillFocus, Generic Event	567
MouseDown, Generic Event	567
MouseMove, Generic Event	568

MouseUp, Generic Event.....	568
MouseWheel, Generic Events	569
OnChange, Generic Event.....	569
OnChangeExecutionCanceled, Generic Event.....	570
OnChangeExecutionToPromoter, Generic Event.....	570
OnExecutionPending, Generic Event.....	570
OnFireExecution, Generic Event	570
OnFireSynapse, Generic Event	571
OnGesture, Generic Event	571
OnPostPaint, Generic Event	571
OnPrePaint, Generic Event.....	572
OnPreSymbolLoading, Generic Event	572
OnSize, Generic Event	572
OnTextChanged, Generic Event.....	573
OnTextChanging, Generic Event	573
OnTimer, Generic Event	573
OnToolTip, Generic Event	573
SetFocus, Generic Event.....	574
SymbolLoading, Generic Event.....	574
SymbolUnloading, Generic Event.....	574
1.15.10. GridWndCmdTarget.....	574
Events	574
OnQueryEnd, GridWndCmdTarget Event.....	574
OnQueryNext, GridWndCmdTarget Event	575
OnQueryStart, GridWndCmdTarget Event	575
OnSelChanged, GridWndCmdTarget Event.....	575
OnSelChanging, GridWndCmdTarget Event.....	575
OnSQLError, GridWndCmdTarget Event.....	575
OnUpdatingDSN, GridWndCmdTarget Event.....	576
Functions	576
AddColumn, GridWndCmdTarget Function	576
CellEditable, GridWndCmdTarget Function	576
DeleteColumn, GridWndCmdTarget Function.....	577
DeleteRow, GridWndCmdTarget Function	577
EditCopy, GridWndCmdTarget Function	578
EnsureVisible, GridWndCmdTarget Function.....	578
FocusCellEditable, GridWndCmdTarget Function	579
GetColCount, GridWndCmdTarget Function.....	579
GetRowCount, GridWndCmdTarget Function	580
GetSelectedRange, GridWndCmdTarget Function.....	580
InsertRow, GridWndCmdTarget Function	581
IsCellSelected, GridWndCmdTarget Function.....	581
IsCellValid, GridWndCmdTarget Function.....	582
IsCellVisible, GridWndCmdTarget Function.....	583
LoadExtSettings, GridWndCmdTarget Function.....	583
LoadFromTextFile, GridWndCmdTarget Function.....	584
RecalcLayout, GridWndCmdTarget Function	584
Refresh, GridWndCmdTarget Function.....	585
SaveExtSettings, GridWndCmdTarget Function.....	585
SaveToTextFile, GridWndCmdTarget Function	585
SelectAll, GridWndCmdTarget Function	586
SetSelectedRange, GridWndCmdTarget Function	587
UpdateDatabase, GridWndCmdTarget Function	587
UpdateVariables, GridWndCmdTarget Function.....	588
Properties.....	588
AutoLayout, GridWndCmdTarget Property.....	588
ButtonPos, GridWndCmdTarget Property	589
ButtonSize, GridWndCmdTarget Property	590
CellBkColor, GridWndCmdTarget Property	590
CellFgColor, GridWndCmdTarget Property.....	591
CellModified, GridWndCmdTarget Property.....	591
CellText, GridWndCmdTarget Property	592
Clickable, GridWndCmdTarget Property	592
ColumnsWidth, GridWndCmdTarget Property	593
CopyBtnText, GridWndCmdTarget Property	593
DeleteBtnText, GridWndCmdTarget Property.....	594
DSN, GridWndCmdTarget Property	594
ExtSettingsFile, GridWndCmdTarget Property.....	595
FocusCellBkColor, GridWndCmdTarget Property	595
FocusCellCol, GridWndCmdTarget Property	596
FocusCellFgColor, GridWndCmdTarget Property.....	596
FocusCellModified, GridWndCmdTarget Property.....	597

FocusCellRow, GridWndCmdTarget Property.....	597
FocusCellText, GridWndCmdTarget Property.....	598
GraphicButtons, GridWndCmdTarget Property	598
InsertBtnText, GridWndCmdTarget Property.....	599
Promptpad,GridWndCmdTarget_Property.....	599
Query, GridWndCmdTarget Property	599
SaveBtnText, GridWndCmdTarget Property	600
SelectAllBtnText, GridWndCmdTarget Property.....	600
ShowCopyBtn, GridWndCmdTarget Property	601
ShowDeleteBtn, GridWndCmdTarget Property	601
ShowInsertBtn, GridWndCmdTarget Property	602
ShowSaveBtn, GridWndCmdTarget Property	602
ShowSelectAllBtn, GridWndCmdTarget Property	603
ShowUpdateBtn, GridWndCmdTarget Property	603
TextFileName, GridWndCmdTarget Property.....	604
TextSeparator, GridWndCmdTarget Property.....	604
UpdateBtnText, GridWndCmdTarget Property.....	605
UpdateVariable, GridWndCmdTarget Property	605
UserName, GridWndCmdTarget Property	605
1.15.11. HisLogWndCmdTarget	606
Events.....	606
OnFilter, HisLogWndCmdTarget Event	606
OnPrint, HisLogWndCmdTarget Event.....	606
OnRefresh, HisLogWndCmdTarget Event.....	606
Functions.....	607
EditCopy, HisLogWndCmdTarget Function	607
EditLayout, HisLogWndCmdTarget Function.....	607
LoadExtSettings, HisLogWndCmdTarget Function.....	608
RecalcLayout, HisLogWndCmdTarget Function	608
Refresh, HisLogWndCmdTarget Function.....	609
SaveExtSettings, HisLogWndCmdTarget Function	609
Properties	610
AutoLayout, HisLogWndCmdTarget Property	610
ButtonPos, HisLogWndCmdTarget Property	610
ButtonSize, HisLogWndCmdTarget Property	611
Clickable, HisLogWndCmdTarget Property	611
EventType, HisLogWndCmdTarget Property.....	612
ExtSettingsFile, HisLogWndCmdTarget Property.....	612
FilterBtnText, HisLogWndCmdTarget Property	613
FilterEvent, HisLogWndCmdTarget Property	613
FilterEventTypeCol, HisLogWndCmdTarget Property	614
FilterEventTypeColCondition, HisLogWndCmdTarget Property	615
FilterFromDate, HisLogWndCmdTarget Property	615
FilterToDate, HisLogWndCmdTarget Property	616
FilterUniqueID, HisLogWndCmdTarget Property	616
FilterUser, HisLogWndCmdTarget Property.....	617
FormatDateTime, HisLogWndCmdTarget Property	618
FormatDuration, HisLogWndCmdTarget Property	618
GraphicButtons, HisLogWndCmdTarget Property	619
IncludeMilliseconds, HisLogWndCmdTarget Property	619
MaxCount, HisLogWndCmdTarget Property	620
NetworkBackupServerName, HisLogWndCmdTarget Property	620
NetworkServerName, HisLogWndCmdTarget Property	621
PrintBtnText, HisLogWndCmdTarget Property	621
Project, HisLogWndCmdTarget Property.....	622
Query, HisLogWndCmdTarget Property.....	622
RefreshBtnText, HisLogWndCmdTarget Property.....	623
ReportFile, HisLogWndCmdTarget Property	624
ShowFilterBtn, HisLogWndCmdTarget Property.....	624
ShowFlatGrid, HisLogWndCmdTarget Property.....	625
ShowPrintBtn, HisLogWndCmdTarget Property	625
ShowRefreshBtn, HisLogWndCmdTarget Property	626
SortBy, HisLogWndCmdTarget Property.....	626
SubItemComment, HisLogWndCmdTarget Property.....	627
SubItemCommentPos, HisLogWndCmdTarget Property	628
SubItemCommentWidth, HisLogWndCmdTarget Property.....	628
SubItemDesc, HisLogWndCmdTarget Property.....	629
SubItemDescPos, HisLogWndCmdTarget Property.....	629
SubItemDescWidth, HisLogWndCmdTarget Property	630
SubItemDuration, HisLogWndCmdTarget Property	630
SubItemDurationPos, HisLogWndCmdTarget Property	631
SubItemDurationWidth, HisLogWndCmdTarget Property	631

SubItemEventId, HisLogWndCmdTarget Property	632
SubItemEventIdPos, HisLogWndCmdTarget Property	632
SubItemEventIdWidth, HisLogWndCmdTarget Property	633
SubItemEventNum, HisLogWndCmdTarget Property	633
SubItemEventNumPos, HisLogWndCmdTarget Property	634
SubItemEventNumWidth, HisLogWndCmdTarget Property	634
SubItemText, HisLogWndCmdTarget Property	635
SubItemTextPos, HisLogWndCmdTarget Property	635
SubItemTextWidth, HisLogWndCmdTarget Property	636
SubItemTime, HisLogWndCmdTarget Property	636
SubItemTimePos, HisLogWndCmdTarget Property	637
SubItemTimeWidth, HisLogWndCmdTarget Property	637
SubItemUser, HisLogWndCmdTarget Property	638
SubItemUserPos, HisLogWndCmdTarget Property	638
SubItemUserWidth, HisLogWndCmdTarget Property	639
<i>1.15.12. HourSelectorCmdTarget.....</i>	<i>639</i>
Events	639
OnAddScheduler, HourSelectorCmdTarget Event	639
OnCancel, HourSelectorCmdTarget Event	640
OnRemoveScheduler, HourSelectorCmdTarget Event	640
OnSave, HourSelectorCmdTarget Event	640
OnSchedulerChanged, HourSelectorCmdTarget Event	640
OnSwitchGridMode, HourSelectorCmdTarget Event	641
OnSwitchHolidays, HourSelectorCmdTarget Event	641
Functions	642
Cancel, HourSelectorCmdTarget Function	642
LoadExtSettings, HourSelectorCmdTarget Function	642
RecalcLayout, HourSelectorCmdTarget Function	643
Save, HourSelectorCmdTarget Function	643
SaveExtSettings, HourSelectorCmdTarget Function	644
Properties.....	644
AddBtnText, HourSelectorCmdTarget Property	644
Border, HourSelectorCmdTarget Property	645
ButtonPos, HourSelectorCmdTarget Property	645
ButtonSize, HourSelectorCmdTarget Property	646
CancelBtnText, HourSelectorCmdTarget Property	646
ColorSelCell, HourSelectorCmdTarget Property	647
DaysText, HourSelectorCmdTarget Property	647
EditMode, HourSelectorCmdTarget Property	647
EndTimeColText, HourSelectorCmdTarget Property	648
ErrorString, HourSelectorCmdTarget Property	649
ExtSettingsFile, HourSelectorCmdTarget Property	649
GraphicButtons, HourSelectorCmdTarget Property	650
GridModeBtnText, HourSelectorCmdTarget Property	650
HolidaysBtnText, HourSelectorCmdTarget Property	651
MaxRow, HourSelectorCmdTarget Property	651
NetworkBackupServerName, HourSelectorCmdTarget Property	652
NetworkServerName, HourSelectorCmdTarget Property	652
PromptPad, HourSelectorCmdTarget Property	653
RemoveBtnText, HourSelectorCmdTarget Property	653
SaveBtnText, HourSelectorCmdTarget Property	654
Scheduler, HourSelectorCmdTarget Property	654
ShowAddBtn, HourSelectorCmdTarget Property	655
ShowCancelBtn, HourSelectorCmdTarget Property	655
ShowColumValue, HourSelectorCmdTarget Property	656
ShowColumVariable, HourSelectorCmdTarget Property	656
ShowComboScheduler, HourSelectorCmdTarget Property	657
ShowHolidaysBtn, HourSelectorCmdTarget Property	657
ShowRemoveBtn, HourSelectorCmdTarget Property	658
ShowSaveBtn, HourSelectorCmdTarget Property	658
StartTimeColText, HourSelectorCmdTarget Property	659
ValueColText, HourSelectorCmdTarget Property	660
VariableColText, HourSelectorCmdTarget Property	660
ValueErrorString, HourSelectorCmdTarget Property	661
<i>1.15.13. IOPortInterface.....</i>	<i>661</i>
<i>1.16. UTILIZZO DELL'IOPORTINTERFACE</i>	<i>661</i>
Functions	662
IOClosePort, IOPortInterface Function	662
IOGetLastError, IOPortInterface Function	662
IOInBufferCount, IOPortInterface Function	663
IOInput, IOPortInterface Function	664

IOOutput, IOPortInterface Function.....	664
IOPortOpen, IOPortInterface Function	665
Properties	666
IOBreak, IOPortInterface Property	666
IOCDHolding, IOPortInterface Property.....	666
IOCTSHolding, IOPortInterface Property	667
IODSRHolding, IOPortInterface Property.....	667
IODTREnable, IOPortInterface Property	668
IOInputLen, IOPortInterface Property.....	668
IOOutBufferCount, IOPortInterface Property	669
IORTSEnable, IOPortInterface Property	669
<i>1.16.1. ListBoxCmdTarget</i>	<i>670</i>
Events.....	670
OnSelected, ListBoxCmdTarget Event.....	670
OnSelecting, ListBoxCmdTarget Event.....	670
Functions.....	671
AddString, ListBoxCmdTarget Function	671
GetCount, ListBoxCmdTarget Function	671
GetSelectedIndex, ListBoxCmdTarget Function	671
GetText, ListBoxCmdTarget Function.....	672
LoadExtSettings, ListBoxCmdTarget Function	672
RefillList, ListBoxCmdTarget Function	673
RemoveString, ListBoxCmdTarget Function	673
SaveExtSettings, ListBoxCmdTarget Function.....	673
Properties	674
ExtSettingsFile, ListBoxCmdTarget Property	674
ListData, ListBoxCmdTarget Property	674
ListVariable, ListBoxCmdTarget Property	675
SortItems,ListBoxCmdTarget Property.....	675
Variable, ListBoxCmdTarget Property	676
<i>1.16.2. NetworkClientCmd.....</i>	<i>676</i>
Functions.....	676
ConnectVariable, NetworkClientCmd Function.....	676
DisconnectVariable, NetworkClientCmd Function.....	677
GetClientRules, NetworkClientCmd Function.....	678
GetRASStation, NetworkClientCmd Function.....	678
IsServerAvailable, NetworkClientCmd Function.....	679
<i>1.16.3. NetworkRedudancyCmd.....</i>	<i>679</i>
Functions.....	679
ActNumRetries, NetworkRedudancyCmd Function	679
CallBackServer, NetworkRedudancyCmd Function	680
ConnectToServer, NetworkRedudancyCmd Function	680
GetTotalPendingMessage, NetworkRedudancyCmd Function	681
IsActiveServer, NetworkRedudancyCmd Function	681
LastInteractionTime, NetworkRedudancyCmd Function	681
PendingStartedDriverOnSecondary, NetworkRedudancyCmd Function	682
SecondaryServerConnected, NetworkRedudancyCmd Function	682
StartedDriverOnSecondary, NetworkRedudancyCmd Function	683
StatusVariable, NetworkRedudancyCmd Function.....	683
Synchronizing, NetworkRedudancyCmd Function	684
Type, NetworkRedudancyCmd Function	684
Properties	685
DriverErrorTimeout, NetworkRedudancyCmd Property	685
MaxHisCacheHits, NetworkRedudancyCmd Property	685
Retries, NetworkRedudancyCmd Property	686
SwitchServerOnDriverError, NetworkRedudancyCmd Property	686
SyncTimeFreq, NetworkRedudancyCmd Property	687
Timeout, NetworkRedudancyCmd Property	687
<i>1.16.4. OPCAECmdTarget.....</i>	<i>688</i>
Functions.....	688
GetEventInQueueAt, OPCAECmdTarget Function.....	688
GetNumEventInQueue, OPCAECmdTarget Function	688
GetNumObjectsInHeap, OPCAECmdTarget Function	688
GetServerStatus, OPCAECmdTarget Function	689
GetServerVendorInfo, OPCAECmdTarget Function.....	689
GetXMLSettings, OPCAECmdTarget Function	689
Reconnect, OPCAECmdTarget Function	689
Refresh, OPCAECmdTarget Function.....	690
Properties	690
MaxEventQueue, OPCAECmdTarget Property	690
Node, OPCAECmdTarget Property	690
ReconnectTime, OPCAECmdTarget Property	691

Server, OPCAECmdTarget Property	691
1.16.5. <i>OPCClientCmdTarget</i>	691
Functions	691
ClearDynOPCItemCache, OPCClientCmdTarget Function	691
DisableDynOPCGroup, OPCClientCmdTarget Function	692
EnableDynOPCGroup, OPCClientCmdTarget Function	693
GetOPCAEClientObject, OPCClientCmdTarget Function	693
GetOPCDAClientObject, OPCClientCmdTarget Function	694
GetXMLSettings, OPCClientCmdTarget Function	695
ReconnectAll, OPCClientCmdTarget Function	695
Properties	696
PoolOPCClientTest, OPCClientCmdTarget Property	696
RefreshRateDynamicOPC, OPCClientCmdTarget Property	696
StartupTimeout, OPCClientCmdTarget Property	697
TimeoutDynamicOperation, OPCClientCmdTarget Property	697
TimeoutOPCClientTest, OPCClientCmdTarget Property	697
1.16.6. <i>OPCClientGroupObjCmdTarget</i>	698
Functions	698
ConnectGroup, OPCClientGroupObjCmdTarget Function	698
GetItemObject, OPCClientGroupObjCmdTarget Function	699
GetName, OPCClientGroupObjCmdTarget Function	699
GetNumObjectsInHeap, OPCClientGroupObjCmdTarget Function	700
GetServerObject, OPCClientGroupObjCmdTarget Function	700
GetXMLSettings, OPCClientGroupObjCmdTarget Function	701
IsGroupConnected, OPCClientGroupObjCmdTarget Function	702
UpdateGroupProperties, OPCClientGroupObjCmdTarget Function	702
Properties	703
Active, OPCClientGroupObjCmdTarget Property	703
DeadBand, OPCClientGroupObjCmdTarget Property	704
EnableGroup, OPCClientGroupObjCmdTarget Property	704
LocalID, OPCClientGroupObjCmdTarget Property	705
NotInUseDisableGroup, OPCClientGroupObjCmdTarget Property	706
NotInUseRefreshRate, OPCClientGroupObjCmdTarget Property	707
TimeBias, OPCClientGroupObjCmdTarget Property	707
UpdateRate, OPCClientGroupObjCmdTarget Property	708
1.16.7. <i>OPCClientItemObjCmdTarget</i>	709
Functions	709
ForceReadData, OPCClientItemObjCmdTarget Function	709
ForceWriteData, OPCClientItemObjCmdTarget Function	709
GetGroupObject, OPCClientItemObjCmdTarget Function	710
GetItemQuality, OPCClientItemObjCmdTarget Function	711
GetItemTimeStamp, OPCClientItemObjCmdTarget Function	712
GetLinkedVariableObject, OPCClientItemObjCmdTarget Function	712
GetNumObjectsInHeap, OPCClientItemObjCmdTarget Function	713
GetXMLSettings, OPCClientItemObjCmdTarget Function	714
IsItemConnected, OPCClientItemObjCmdTarget Function	714
ReconnectItem, OPCClientItemObjCmdTarget Function	715
Properties	716
EnableRead, OPCClientItemObjCmdTarget Property	716
EnableWrite, OPCClientItemObjCmdTarget Property	716
ItemID, OPCClientItemObjCmdTarget Property	717
ItemPath, OPCClientItemObjCmdTarget Property	718
LinkedVariable, OPCClientItemObjCmdTarget Property	718
ReRead, OPCClientItemObjCmdTarget Property	719
SyncDataAtStartup, OPCClientItemObjCmdTarget Property	720
vtType, OPCClientItemObjCmdTarget Property	720
WriteSync, OPCClientItemObjCmdTarget Property	721
1.16.8. <i>OPCClientObjCmdTarget</i>	722
Functions	722
ConnectServer, OPCClientObjCmdTarget Function	722
GetGroupObject, OPCClientObjCmdTarget Function	723
GetNodeName, OPCClientObjCmdTarget Function	723
GetNumObjectsInHeap, OPCClientObjCmdTarget Function	724
GetOPCClientDocObj, OPCClientObjCmdTarget Function	724
GetServerCLSID, OPCClientObjCmdTarget Function	725
GetServerName, OPCClientObjCmdTarget Function	725
GetServerStatus, OPCClientObjCmdTarget Function	726
GetServerVendorInfo, OPCClientObjCmdTarget Function	726
GetXMLSettings, OPCClientObjCmdTarget Function	727
IsConnected, OPCClientObjCmdTarget Function	727
Properties	728
ReconnectTime, OPCClientObjCmdTarget Property	728

ReReadDynamicItems, OPCClientObjCmdTarget Property	729
SyncDynamicItemsAtStartup, OPCClientObjCmdTarget Property	729
WatchdogTime, OPCClientObjCmdTarget Property	730
1.16.9. OPCServerCmdTarget.....	730
Properties	730
ServerStatus, OPCServerCmdTarget Property.....	730
Functions.....	731
FireAEEEvent, OPCServerCmdTarget Function	731
GetNumServingTags, OPCServerCmdTarget Function	731
GetServerName,OPCServerCmdTarget Function	732
GetNumConnectedClients, OPCServerCmdTarget Function	732
1.16.10. OPCUAClientCmdTarget	733
Functions.....	733
GetOPCUAClientObject, OPCUAClientCmdTarget Function	733
GetXMLSettings, OPCUAClientCmdTarget Function	733
Property	734
StartupTimeout, OPCUAClientCmdTarget Property	734
1.16.11. OPCUAClientObjCmdTarget.....	734
Functions.....	734
GetBackupEndpoint, OPCUAClientObjCmdTarget Function	734
GetEndpoint, OPCUAClientObjCmdTarget Function	735
GetNodeName, OPCUAClientObjCmdTarget Function	736
GetOPCUAClientDocObj, OPCUAClientObjCmdTarget Function	736
GetSecurityMode, OPCUAClientObjCmdTarget Function	737
GetServerName, OPCUAClientObjCmdTarget Function	737
GetSecurityPolicy, OPCUAClientObjCmdTarget Function.....	738
GetSessionObject, OPCUAClientObjCmdTarget Function.....	738
GetXMLSettings, OPCUAClientObjCmdTarget Function	739
IsConnected, OPCUAClientObjCmdTarget Function.....	739
Property	740
Endpoint, OPCUAClientObjCmdTarget Property.....	740
Retries, OPCUAClientObjCmdTarget Property	740
StatusVariable, OPCUAClientObjCmdTarget Property	741
BackupEndpoint, OPCUAClientObjCmdTarget Property.....	742
ReconnectTime, OPCUAClientObjCmdTarget Property	742
1.16.12. OPCUAClientSessionObjCmdTarget.....	743
Property	743
ConnectTimeout, OPCUAClientSessionObjCmdTarget Property.....	743
PublishingInterval, OPCUAClientSessionObjCmdTarget Property	744
SamplingInterval, OPCUAClientSessionObjCmdTarget Property.....	744
Functions.....	745
GetItemObject, OPCUAClientSessionObjCmdTarget Function.....	745
GetName, OPCUAClientSessionObjCmdTarget Function.....	746
GetServerObject, OPCUAClientSessionObjCmdTarget Function	746
GetXMLSettings, OPCUAClientSessionObjCmdTarget Function	747
IsSessionConnected, OPCUAClientSessionObjCmdTarget Function.....	748
1.16.13. OPCUAClientItemObjCmdTarget.....	748
Functions.....	748
GetItemQuality, OPCUAClientItemObjCmdTarget Function	748
GetItemTimeStamp, OPCUAClientItemObjCmdTarget Function	749
GetLinkedVariableObject, OPCUAClientItemObjCmdTarget Function	750
GetName, OPCUAClientItemObjCmdTarget Function.....	751
GetSessionObject, OPCUAClientItemObjCmdTarget Function.....	751
GetXMLSettings, OPCUAClientItemObjCmdTarget Function	752
Property	753
LinkedVariable, OPCUAClientItemObjCmdTarget Property	753
NodeID, OPCUAClientItemObjCmdTarget Property	753
EnableRead, OPCUAClientItemObjCmdTarget Property	754
EnableWrite, OPCUAClientItemObjCmdTarget Property.....	755
1.16.14. PmeDocCmdTarget.....	755
1.17. UTILIZZO DELLA PMEDOCMDTARGET	755
Functions.....	756
AckAllAlarms, PmeDocCmdTarget Function	756
AddSysLogMessage, PmeDocCmdTarget Function	756
CreateObjectLic, PmeDocCmdTarget Function	756
CreateRemoteObject, PmeDocCmdTarget Function	757
GetAlarm, PmeDocCmdTarget Function	757
GetAlarmsPath, PmeDocCmdTarget Function.....	758
GetAppTimeZone, PmeDocCmdTarget Function	759
GetChildProject, PmeDocCmdTarget Function	759
GetCurrentListAlarms, PmeDocCmdTarget Function	759

GetDataLoggerRecipe, PmeDocCmdTarget Function	760
GetDataLoggerRecipePath, PmeDocCmdTarget Function	761
GetDataPath, PmeDocCmdTarget Function	761
GetDrawingPath, PmeDocCmdTarget Function	761
GetDriverInterface, PmeDocCmdTarget Function	762
GetEvent, PmeDocCmdTarget Function	762
GetFatherProject, PmeDocCmdTarget Function	763
GetHisLogADOConn, PmeDocCmdTarget Function	763
GetHisLogDNSConnectionString, PmeDocCmdTarget Function	764
GetIOPortInterface, PmeDocCmdTarget Function	765
GetOPCUAClient, PmeDocCmdTarget Function	765
GetLastAlarmOn, PmeDocCmdTarget Function	766
GetLogPath, PmeDocCmdTarget Function	766
GetNetworkClient, PmeDocCmdTarget Function	766
GetNetworkRedundancy, PmeDocCmdTarget Function	767
GetNetworkServer, PmeDocCmdTarget Function	767
GetNetworkUserLogPath, PmeDocCmdTarget Function	768
GetOPCCClient, PmeDocCmdTarget Function	768
GetOPCServer, PmeDocCmdTarget Function	769
GetProjectFileName, PmeDocCmdTarget Function	769
GetProjectTitle, PmeDocCmdTarget Function	770
GetRealTimeDB, PmeDocCmdTarget Function	770
GetResourcePath, PmeDocCmdTarget Function	770
GetSiteCode, PmeDocCmdTarget Function	771
GetScaling, PmeDocCmdTarget Function	771
GetScheduler, PmeDocCmdTarget Function	772
GetSynopticInterface, PmeDocCmdTarget Function	772
GetUserAndGroup, PmeDocCmdTarget Function	773
IsAlarmAreaActive, PmeDocCmdTarget Function	774
IsAlarmAreaON, PmeDocCmdTarget Function	774
IsChildProject, PmeDocCmdTarget Function	774
IsInStoppingMode, PmeDocCmdTarget Function	775
IsRunning, PmeDocCmdTarget Function	775
PlaySoundFile, PmeDocCmdTarget Function	776
ResetAllAlarms, PmeDocCmdTarget Function	776
RunningOnCE, PmeDocCmdTarget Function	777
RunScript, PmeDocCmdTarget Function	777
SendDispatcherMessage, PmeDocCmdTarget Function	778
SetSiteCode, PmeDocCmdTarget Function	778
StartAlarmDispatcher, PmeDocCmdTarget Function	779
StopPlaySoundFile, PmeDocCmdTarget Function	780
UnloadScript, PmeDocCmdTarget Function	780
Properties	780
ActiveLanguage, PmeDocCmdTarget Property	780
ChildProjectActiveNetworkServer, PmeDocCmdTarget Property	781
ChildProjectBackupNetworkServer, PmeDocCmdTarget Property	781
ChildProjectName, PmeDocCmdTarget Property	782
ChildProjectNetworkServer, PmeDocCmdTarget Property	782
ChildProjectStartable, PmeDocCmdTarget Property	783
HisLogAlarmDurationDays, PmeDocCmdTarget Property	783
HisLogAlarmDurationHours, PmeDocCmdTarget Property	783
HisLogAlarmDurationMinutes, PmeDocCmdTarget Property	784
HisLogAlarmTable, PmeDocCmdTarget Property	784
HisLogCommentColName, PmeDocCmdTarget Property	785
HisLogDefVarCharPrecision, PmeDocCmdTarget Property	785
HisLogDescriptionColName, PmeDocCmdTarget Property	786
HisLogDriverDurationDays, PmeDocCmdTarget Property	786
HisLogDriverDurationHours, PmeDocCmdTarget Property	787
HisLogDriverDurationMinutes, PmeDocCmdTarget Property	787
HisLogDriverTable, PmeDocCmdTarget Property	788
HisLogDsn, PmeDocCmdTarget Property	788
HisLogDurationColName, PmeDocCmdTarget Property	789
HisLogEventTypeColName, PmeDocCmdTarget Property	789
HisLogEventTypeNumColName, PmeDocCmdTarget Property	790
HisLogLocalTimeColName, PmeDocCmdTarget Property	791
HisLogMaxCacheBeforeFlush, PmeDocCmdTarget Property	791
HisLogMaxError, PmeDocCmdTarget Property	792
HisLogMaxNumberTrans, PmeDocCmdTarget Property	792
HisLogMSecColName, PmeDocCmdTarget Property	792
HisLogRecycleDBConnection, PmeDocCmdTarget Property	793
HisLogSubEventTypeColName, PmeDocCmdTarget Property	793
HisLogSysTable, PmeDocCmdTarget Property	794

HisLogSystemsDurationDays, PmeDocCmdTarget Property	795
HisLogSystemsDurationHours, PmeDocCmdTarget Property	795
HisLogSystemsDurationMinutes, PmeDocCmdTarget Property	795
HisLogTimeColName, PmeDocCmdTarget Property	796
HisLogTransactionIDColName, PmeDocCmdTarget Property	796
HisLogUseIMDB, PmeDocCmdTarget Property	797
HisLogUser, PmeDocCmdTarget Property	797
HisLogUserColName, PmeDocCmdTarget Property	798
HisLogUniqueIDColName, PmeDocCmdTarget Property	798
ShutdownScript, PmeDocCmdTarget Property	799
StartChildProjectWithFather, PmeDocCmdTarget Property	799
StartupScreen, PmeDocCmdTarget Property	800
StartupScript, PmeDocCmdTarget Property	800
StoreCryptProject, PmeDocCmdTarget Property	801
StoreCryptProjectResources, PmeDocCmdTarget Property	801
StoreCryptProjectStrings, PmeDocCmdTarget Property	802
StoreUnicodeProject, PmeDocCmdTarget Property	802
StoreZippedProject, PmeDocCmdTarget Property	803
StringFromID, PmeDocCmdTarget Property	803
TargetClientJ2ME, PmeDocCmdTarget Property	804
TargetClientJ2SE, PmeDocCmdTarget Property	804
TargetClientWin32, PmeDocCmdTarget Property	804
TargetClientWinCE, PmeDocCmdTarget Property	805
TargetPlatformWin32, PmeDocCmdTarget Property	805
TargetPlatformWinCE, PmeDocCmdTarget Property	806
1.17.1. RASStationInterface.....	806
Functions.....	806
GetXMLSettings, RASStationInterface Function.....	806
Properties	807
ConnectionVariable, RASStationInterface Property	807
DisconnectAfterSecs, RASStationInterface Property	807
EndConnectionTime, RASStationInterface Property	808
IsConnected, RASStationInterface Property.....	809
LastConnectionTime, RASStationInterface Property.....	809
LastRASErrorNumber, RASStationInterface Property	810
LastRASErrorString, RASStationInterface Property	810
NumRetries, RASStationInterface Property.....	811
Password, RASStationInterface Property.....	811
PhoneBookEntry, RASStationInterface Property	812
PhoneNumber, RASStationInterface Property	812
PromptForConnection, RASStationInterface Property	813
RetryAfterSecs, RASStationInterface Property	813
ShowConnectionDlg, RASStationInterface Property	814
StartConnectionTime, RASStationInterface Property.....	815
TotalConnectionTime, RASStationInterface Property.....	815
UserName, RASStationInterface Property	816
1.17.2. RecipeWndCmdTarget.....	816
Events.....	816
OnActivateRecipe, RecipeWndCmdTarget Event.....	816
OnCopyRecipe, RecipeWndCmdTarget Event	817
OnDeleteRecipe, RecipeWndCmdTarget Event	817
OnExportRecipe, RecipeWndCmdTarget Event	817
OnImportRecipe, RecipeWndCmdTarget Event.....	817
OnPasteRecipe, RecipeWndCmdTarget Event.....	817
OnPrintRecipe, RecipeWndCmdTarget Event.....	818
OnReadRecipe, RecipeWndCmdTarget Event	818
OnRecipeIndexChanged, RecipeWndCmdTarget Event.....	818
OnRefreshRecipe, RecipeWndCmdTarget Event.....	818
OnSaveRecipe, RecipeWndCmdTarget Event	818
Functions.....	819
EditLayout, RecipeWndCmdTarget Function.....	819
ExportRecipeToCSV, RecipeWndCmdTarget Function	819
ImportRecipeFromCSV, RecipeWndCmdTarget Function.....	820
LoadExtSettings, RecipeWndCmdTarget Function.....	821
RecalcLayout, RecipeWndCmdTarget Function	821
Reconnect, RecipeWndCmdTarget Function	822
Refresh, RecipeWndCmdTarget Function.....	822
SaveExtSettings, RecipeWndCmdTarget Function	822
Properties.....	823
ActivateBtnText, RecipeWndCmdTarget Property	823
ActivateMessage, RecipeWndCmdTarget Property	823
AutoLayout, RecipeWndCmdTarget Property	824

ButtonPos, RecipeWndCmdTarget Property	824
ButtonSize, RecipeWndCmdTarget Property	825
Clickable, RecipeWndCmdTarget Property	826
CopyBtnText, RecipeWndCmdTarget Property	826
CurrentRecipeVariable, RecipeWndCmdTarget Property	827
DeleteBtnText, RecipeWndCmdTarget Property	827
DeleteMessage, RecipeWndCmdTarget Property	828
ErrorString, RecipeWndCmdTarget Property	828
ExportBtnText, RecipeWndCmdTarget Property	829
ExtSettingsFile, RecipeWndCmdTarget Property	829
GraphicButtons, RecipeWndCmdTarget Property	830
ImpExpSeparator, RecipeWndCmdTarget Property	830
ImportBtnText, RecipeWndCmdTarget Property	831
NetworkBackupServerName, RecipeWndCmdTarget Property	831
NetworkServerName, RecipeWndCmdTarget Property	832
PasteBtnText, RecipeWndCmdTarget Property	832
PrintBtnText, RecipeWndCmdTarget Property	833
Project, RecipeWndCmdTarget Property	833
PromptPad, RecipeWndCmdTarget Property	834
ReadBtnText, RecipeWndCmdTarget Property	834
Recipe, RecipeWndCmdTarget Property	835
RefreshBtnText, RecipeWndCmdTarget Property	835
SaveBtnText, RecipeWndCmdTarget Property	836
SaveMessage, RecipeWndCmdTarget Property	836
ShowActivateBtn, RecipeWndCmdTarget Property	837
ShowCopyBtn, RecipeWndCmdTarget Property	837
ShowDeleteBtn, RecipeWndCmdTarget Property	838
ShowExportBtn, RecipeWndCmdTarget Property	838
ShowImportBtn, RecipeWndCmdTarget Property	839
ShowPasteBtn, RecipeWndCmdTarget Property	839
ShowPrintBtn, RecipeWndCmdTarget Property	840
ShowReadBtn, RecipeWndCmdTarget Property	840
ShowRefreshBtn, RecipeWndCmdTarget Property	841
ShowSaveBtn, RecipeWndCmdTarget Property	841
SubItemDescription, RecipeWndCmdTarget Property	842
SubItemDescriptionPos, RecipeWndCmdTarget Property	842
SubItemDescriptionWidth, RecipeWndCmdTarget Property	843
SubItemMax, RecipeWndCmdTarget Property	843
SubItemMaxPos, RecipeWndCmdTarget Property	844
SubItemMaxWidth, RecipeWndCmdTarget Property	844
SubItemMin, RecipeWndCmdTarget Property	845
SubItemMinPos, RecipeWndCmdTarget Property	845
SubItemMinWidth, RecipeWndCmdTarget Property	846
SubItemUnits, RecipeWndCmdTarget Property	846
SubItemUnitsPos, RecipeWndCmdTarget Property	847
SubItemUnitsWidth, RecipeWndCmdTarget Property	847
SubItemValue, RecipeWndCmdTarget Property	848
SubItemValuePos, RecipeWndCmdTarget Property	848
SubItemValueWidth, RecipeWndCmdTarget Property	849
SubItemVariable, RecipeWndCmdTarget Property	849
SubItemVariablePos, RecipeWndCmdTarget Property	850
SubItemVariableWidth, RecipeWndCmdTarget Property	850
1.17.3. ScalingCmdTarget	851
Functions	851
GetXMLSettings, ScalingCmdTarget Function	851
Reinit, ScalingCmdTarget Function	851
Properties	852
DeadBandValue, ScalingCmdTarget Property	852
Enabled, ScalingCmdTarget Property	852
Name, ScalingCmdTarget Property	853
RawMaxValue, ScalingCmdTarget Property	853
RawMinValue, ScalingCmdTarget Property	853
RawVariableName, ScalingCmdTarget Property	854
ScaledMaxValue, ScalingCmdTarget Property	854
ScaledMinValue, ScalingCmdTarget Property	855
ScaleVariableName, ScalingCmdTarget Property	855
1.17.4. SchedulerCmdTarget	856
Functions	856
AddHoliday, SchedulerCmdTarget Function	856
GetHolidaysString, SchedulerCmdTarget Function	856
GetXMLSettings, SchedulerCmdTarget Function	857
IsHoliday, SchedulerCmdTarget Function	857

RemoveHoliday, SchedulerCmdTarget Function	858
Reset, SchedulerCmdTarget Function	858
SaveRetentive, SchedulerCmdTarget Function	859
Properties	859
CommandList, SchedulerCmdTarget Property	859
CommandListOff, SchedulerCmdTarget Property	860
Enabled, SchedulerCmdTarget Property	860
EnableVariable, SchedulerCmdTarget Property	860
HasHolidays, SchedulerCmdTarget Property	861
HolidaysPlan, SchedulerCmdTarget Property	861
Name, SchedulerCmdTarget Property	862
Plan, SchedulerCmdTarget Property	863
TimeAndDate, SchedulerCmdTarget Property	864
Type, SchedulerCmdTarget Property	864
TreatHolidaysAsSunday, SchedulerCmdTarget Property	865
1.17.5. ScriptMEInterface	866
Events	866
Loading, ScriptMEInterface Event	866
Main, ScriptMEInterface Event	866
Unloading, ScriptMEInterface Event	866
Functions	866
EnterGlobalCriticalSection, ScriptMEInterface Function	866
GetCurrentUser, ScriptMEInterface Function	867
GetInstanceNumber, ScriptMEInterface Function	867
GetParameter, ScriptMEInterface Function	868
HasPreviousInstance, ScriptMEInterface Function	868
IsStopping, ScriptMEInterface Function	869
LeaveGlobalCriticalSection, ScriptMEInterface Function	869
RunningOnServerSide, ScriptMEInterface Function	870
ShowDebuggerWnd, ScriptMEInterface Function	870
Properties	870
MaxInstances, ScriptMEInterface Property	870
ModalDialog, ScriptMEInterface Property	871
RunAtServer, ScriptMEInterface Property	871
SeparateThread, ScriptMEInterface Property	872
SleepExecution, ScriptMEInterface Property	872
StatusVariable, ScriptMEInterface Property	872
SyncroScriptTimeout, ScriptMEInterface Property	873
ThreadPriority, ScriptMEInterface Property	873
UseOwnTrace, ScriptMEInterface Property	874
UseUIInterface, ScriptMEInterface Property	874
1.17.6. SynopticCmdTarget	874
Events	874
Click, Generic Event	874
DbClick, Generic Event	875
KeyDown, Generic Event	875
KeyPress, Generic Event	875
KeyUp, Generic Event	875
KillFocus, Generic Event	876
MouseDown, Generic Event	876
MouseMove, Generic Event	876
MouseUp, Generic Event	877
OnActivate, SynopticCmdTarget Event	877
OnQueryEndSession, SynopticCmdTarget Event	877
OnSize, SynopticCmdTarget Event	878
OnStartSynopsisExecution, SynopticCmdTarget Event	878
OnStopSynopsisExecution, SynopticCmdTarget Event	878
OnTimer, Generic Event	878
SetFocus, Generic Event	879
SynopticLoading, SynopticCmdTarget Event	879
SynopticUnloading, SynopticCmdTarget Event	879
Functions	879
CloseSynoptic, SynopticCmdTarget Function	879
CreateNewSymbol, SynopticCmdTarget Function	880
DestroySymbol, SynopticCmdTarget Function	881
GetAbsoluteSubObject, SynopticCmdTarget Function	882
GetActiveUserObject, SynopticCmdTarget Function	882
GetAlias, SynopticCmdTarget Function	883
GetAliasListName, SynopticCmdTarget Function	883
GetAliasListValue, SynopticCmdTarget Function	884
GetFocus, SynopticCmdTarget Function	884
GetImage, SynopticCmdTarget Function	885

GetInstanceNumber, SynopticCmdTarget Function	885
GetObjectByUniqueID, SynopticCmdTarget Function	886
GetParameterVariable, SynopticCmdTarget Function	886
GetSubObject, SynopticCmdTarget Function	887
GetTimeZone, SynopticCmdTarget Function	887
GetWindowPos, SynopticCmdTarget Function	888
IsRemoteClientView, SynopticCmdTarget Function	889
PrintSynoptic, SynopticCmdTarget Function	889
RemoveAlias, SynopticCmdTarget Function	890
RemoveAllAliases, SynopticCmdTarget Function	890
SaveImageToFile, SynopticCmdTarget Function	891
SetAlias, SynopticCmdTarget Function	892
SetFocusTo, SynopticCmdTarget Function	892
SetRedraw, SynopticCmdTarget Function	893
SetSynopsisVisible, SynopticCmdTarget Function	893
SetWindowPos, SynopticCmdTarget Function	894
ZoomIn, SynopticCmdTarget Function	894
ZoomOut, SynopticCmdTarget Function	895
ZoomTo, SynopticCmdTarget Function	895
Properties	895
BackColor, SynopticCmdTarget Property	895
BackGroundFileBitmap, SynopticCmdTarget Property	896
BackGroundFileBitmapTile, SynopticCmdTarget Property	897
CXBackImage, SynopticCmdTarget Property	897
CYBackImage, SynopticCmdTarget Property	898
CyclicExecution, SynopticCmdTarget Property	899
EnableScrollBars, SynopticCmdTarget Property	899
FastTickCounter, SynopticCmdTarget Property	900
FastTickFrequency, SynopticCmdTarget Property	900
FitInWindow, SynopticCmdTarget Property	901
FrameTitle, SynopticCmdTarget Property	901
GlobalContainerName, SynopticCmdTarget Property	902
GradientColor, SynopticCmdTarget Property	902
GradientFill, SynopticCmdTarget Property	903
KeepAspectRatio, SynopticCmdTarget Property	904
LayerVariable, SynopticCmdTarget Property	904
MaxInstances, SynopticCmdTarget Property	905
NonDestroyable, SynopticCmdTarget Property	905
NumColors, SynopticCmdTarget Property	906
ParameterFileName, SynopticCmdTarget Property	906
ScrollPositionX, SynopticCmdTarget Property	907
ScrollPositionY, SynopticCmdTarget Property	907
SeparateThread, SynopticCmdTarget Property	908
ShowOnMDITabsFlag, SynopticCmdTarget Property	908
SlowTickCounter, SynopticCmdTarget Property	909
SlowTickFrequency, SynopticCmdTarget Property	909
SynopsisExecution, SynopticCmdTarget Property	910
SynopticHeight, SynopticCmdTarget Property	910
SynopticID, SynopticCmdTarget Property	911
SynopticPublicSource, SynopticCmdTarget Property	911
SynopticWidth, SynopticCmdTarget Property	912
UseAntialiasing, SynopticCmdTarget Property	912
UseIMDB,DLRCmdTarget property	913
XBackImage, SynopticCmdTarget Property	913
YBackImage, SynopticCmdTarget Property	914
ZoomFactorX, SynopticCmdTarget Property	914
ZoomFactorY, SynopticCmdTarget Property	915
1.17.7. TraceDBWndCmdTarget	915
Events	915
OnFilter, TraceDBWndCmdTarget Event	915
OnPrint, TraceDBWndCmdTarget Event	915
OnRefresh, TraceDBWndCmdTarget Event	916
Functions	916
EditCopy, TraceDBWndCmdTarget Function	916
EditLayout, TraceDBWndCmdTarget Function	916
LoadExtSettings, TraceDBWndCmdTarget Function	917
RecalcLayout, TraceDBWndCmdTarget Function	917
Refresh, TraceDBWndCmdTarget Function	918
Properties	918
AutoLayout, TraceDBWndCmdTarget Property	918
ButtonPos, TraceDBWndCmdTarget Property	919
ButtonSize, TraceDBWndCmdTarget Property	920

Clickable, TraceDBWndCmdTarget Property	920
ExtSettingsFile, TraceDBWndCmdTarget Property	921
FileReport, TraceDBWndCmdTarget Property	921
FilterBtnText, TraceDBWndCmdTarget Property	922
FilterFromDate, TraceDBWndCmdTarget Property	922
FilterToDate, TraceDBWndCmdTarget Property	923
FilterUser, TraceDBWndCmdTarget Property	923
GraphicButtons, TraceDBWndCmdTarget Property	924
IncludeMilliseconds, TraceDBWndCmdTarget Property	924
MaxCount, TraceDBWndCmdTarget Property	925
NetworkBackupServerName, TraceDBWndCmdTarget Property	925
NetworkServerName, TraceDBWndCmdTarget Property	926
PrintBtnText, TraceDBWndCmdTarget Property	926
Project, TraceDBWndCmdTarget Property	927
Query, TraceDBWndCmdTarget Property	928
RefreshBtnText, TraceDBWndCmdTarget Property	928
SaveExtSettings, TraceDBWndCmdTarget Function	929
ShowFilterBtn, TraceDBWndCmdTarget Property	929
ShowPrintBtn, TraceDBWndCmdTarget Property	930
ShowRefreshBtn, TraceDBWndCmdTarget Property	930
SortBy, TraceDBWndCmdTarget Property	931
SubItemAction, TraceDBWndCmdTarget Property	932
SubItemActionPos, TraceDBWndCmdTarget Property	932
SubItemActionWidth, TraceDBWndCmdTarget Property	933
SubItemAfter, TraceDBWndCmdTarget Property	933
SubItemAfterPos, TraceDBWndCmdTarget Property	934
SubItemAfterWidth, TraceDBWndCmdTarget Property	934
SubItemBefore, TraceDBWndCmdTarget Property	935
SubItemBeforePos, TraceDBWndCmdTarget Property	935
SubItemBeforeWidth, TraceDBWndCmdTarget Property	936
SubItemQuality, TraceDBWndCmdTarget Property	936
SubItemQualityPos, TraceDBWndCmdTarget Property	937
SubItemQualityWidth, TraceDBWndCmdTarget Property	937
SubItemTime, TraceDBWndCmdTarget Property	938
SubItemTimePos, TraceDBWndCmdTarget Property	938
SubItemTimeStamp, TraceDBWndCmdTarget Property	939
SubItemTimeStampPos, TraceDBWndCmdTarget Property	939
SubItemTimeStampWidth, TraceDBWndCmdTarget Property	940
SubItemTimeWidth, TraceDBWndCmdTarget Property	940
SubItemUser, TraceDBWndCmdTarget Property	941
SubItemUserPos, TraceDBWndCmdTarget Property	941
SubItemUserWidth, TraceDBWndCmdTarget Property	942
SubItemValue, TraceDBWndCmdTarget Property	942
SubItemValuePos, TraceDBWndCmdTarget Property	943
SubItemValueWidth, TraceDBWndCmdTarget Property	943
SubItemVarDesc, TraceDBWndCmdTarget Property	944
SubItemVarDescPos, TraceDBWndCmdTarget Property	944
SubItemVarDescWidth, TraceDBWndCmdTarget Property	945
SubItemVarGroup, TraceDBWndCmdTarget Property	945
SubItemVarGroupPos, TraceDBWndCmdTarget Property	946
SubItemVarGroupWidth, TraceDBWndCmdTarget Property	946
SubItemVarName, TraceDBWndCmdTarget Property	947
SubItemVarNamePos, TraceDBWndCmdTarget Property	947
SubItemVarNameWidth, TraceDBWndCmdTarget Property	948
Variable, TraceDBWndCmdTarget Property	948
1.17.8. TrendCmdTarget.....	949
Events.....	949
OnChangingState, TrendCmdTarget Event	949
OnCursorPosChanged, TrendCmdTarget Event	949
OnErrorRecordset, TrendCmdTarget Event	949
OnExpand, TrendCmdTarget Event	949
OnExportEnd, TrendCmdTarget Event	950
OnExportNext, TrendCmdTarget Event	950
OnExportStart, TrendCmdTarget Event	950
OnFailedCreatingThread, TrendCmdTarget Event	950
OnImportEnd, TrendCmdTarget Event	951
OnImportNext, TrendCmdTarget Event	951
OnImportStart, TrendCmdTarget Event	951
OnNext, TrendCmdTarget Event	951
OnPageChanged, TrendCmdTarget Event	952
OnPageEnd, TrendCmdTarget Event	952
OnPageNext, TrendCmdTarget Event	952

OnPagePrev, TrendCmdTarget Event	953
OnPageStart, TrendCmdTarget Event.....	953
OnPositionScrolled, TrendCmdTarget Event	953
OnPrev, TrendCmdTarget Event	953
OnPrint, TrendCmdTarget Event	954
OnRecordsetMoveNext, TrendCmdTarget Event.....	954
OnRecordsetQueryEnd, TrendCmdTarget Event.....	954
OnRecordsetQueryStart, TrendCmdTarget Event	955
OnResetZoom, TrendCmdTarget Event.....	955
OnStartRecording, TrendCmdTarget Event.....	955
OnStartZoom, TrendCmdTarget Event	955
OnStopRecording, TrendCmdTarget Event	956
OnUpdateData, TrendCmdTarget Event	956
OnZoomAreaChanged, TrendCmdTarget Event.....	956
Functions	956
AddPen, TrendCmdTarget Function	956
ClearAllSavedValues, TrendCmdTarget Function	957
ClearSavedValues, TrendCmdTarget Function	958
CloseBackupLink, TrendCmdTarget Function	958
CopyLegendToClipboard, TrendCmdTarget Function.....	959
EditPenProperties, TrendCmdTarget Function	960
ExportToClipboard, TrendCmdTarget Function.....	960
ExportToFile, TrendCmdTarget Function	961
GetCurrentDataLoggerName, TrendCmdTarget Function.....	962
GetCursorDateTime, TrendCmdTarget Function	962
GetCursorDateTimeMsec, TrendCmdTarget Function	963
GetCursorDateTimeString, TrendCmdTarget Function	963
GetCursorPenValue, TrendCmdTarget Function	964
GetCursorPosInLegendArea, TrendCmdTarget Function	964
GetCursorPosInPenArea, TrendCmdTarget Function	965
GetCursorPosInScaleArea, TrendCmdTarget Function.....	966
GetCursorPosInTimeArea, TrendCmdTarget Function	966
GetCursorPosInTrendArea, TrendCmdTarget Function	967
GetDateTimeColumnName, TrendCmdTarget Function	967
GetFirstValidDateTime, TrendCmdTarget Function	968
GetFirstValidDateTimeMs, TrendCmdTarget Function	969
GetLastValidDateTime, TrendCmdTarget Function	969
GetLastValidDateTimeMs, TrendCmdTarget Function.....	970
GetLastValidValuePosition, TrendCmdTarget Function	970
GetMaxPage, TrendCmdTarget Function	971
GetPenNameFromList, TrendCmdTarget Function	971
GetPensNumber, TrendCmdTarget Function	972
GetTimeFontOrientation, TrendCmdTarget Function	972
GetZoomAreaDateTimeFrom, TrendCmdTarget Function.....	973
GetZoomAreaDateTimeMsecFrom, TrendCmdTarget Function.....	973
GetZoomAreaDateTimeMsecTo, TrendCmdTarget Function	974
GetZoomAreaPenValueFrom, TrendCmdTarget Function	974
GetZoomAreaPenValueTo, TrendCmdTarget Function.....	975
GetZoomAreaDateTimeStringFrom, TrendCmdTarget Function	975
GetZoomAreaDateTimeStringTo, TrendCmdTarget Function.....	976
GetZoomAreaDateTimeTo, TrendCmdTarget Function	976
GetZoomAreaScaleValueFrom, TrendCmdTarget Function.....	977
GetZoomAreaScaleValueTo, TrendCmdTarget Function.....	977
ImportFromClipboard, TrendCmdTarget Function	978
ImportFromFile, TrendCmdTarget Function	979
IsCursorPosInLegendArea, TrendCmdTarget Function	980
IsCursorPosInPenArea, TrendCmdTarget Function	980
IsCursorPosInScaleArea, TrendCmdTarget Function	981
IsCursorPosInTimeArea, TrendCmdTarget Function.....	981
IsCursorPosInTrendArea, TrendCmdTarget Function	982
LinkToDataLogger, TrendCmdTarget Function	982
LoadExtSettings, TrendCmdTarget Function	983
MoveCursorToMousePos, TrendCmdTarget Function.....	983
PageEnd, TrendCmdTarget Function	984
PageNext, TrendCmdTarget Function	984
PagePrev, TrendCmdTarget Function	985
PageStart, TrendCmdTarget Function.....	985
PrintTrend, TrendCmdTarget Function	986
Refresh, TrendCmdTarget Function.....	987
RemovePen, TrendCmdTarget Function	987
Requery, TrendCmdTarget Function.....	988
ResetZoom, TrendCmdTarget Function.....	988

RestartStatistic, TrendCmdTarget Function	989
SaveAllCurrentValue, TrendCmdTarget Function	989
SaveCurrentValue, TrendCmdTarget Function.....	990
SaveExtSettings, TrendCmdTarget Function	990
ScrollPosNext, TrendCmdTarget Function.....	991
ScrollPosPrev, TrendCmdTarget Function	991
SetSamplesValue, TrendCmdTarget Function	992
SetTimeFontOrientation, TrendCmdTarget Function	993
StartPanMode, TrendCmdTarget Function	993
StartZoomMode, TrendCmdTarget Function	994
Properties	994
AllBtnText, TrendCmdTarget Property	994
BackupLink, TrendCmdTarget Property	995
BorderLegend, TrendCmdTarget Property	995
BorderLegendRaised, TrendCmdTarget Property	996
BorderPen, TrendCmdTarget Property	996
BorderPenRaised, TrendCmdTarget Property	997
BorderTime, TrendCmdTarget Property.....	997
BorderTimeRaised, TrendCmdTarget Property	998
BorderTrend, TrendCmdTarget Property	998
BorderTrendRaised, TrendCmdTarget Property	999
BrushColor, TrendCmdTarget Property	999
ButtonPos, TrendCmdTarget Property.....	1000
ButtonSize, TrendCmdTarget Property.....	1000
ColumnSeparator, TrendCmdTarget Property	1001
CompareTimeFrameBtnColor, TrendCmdTarget Property	1001
ComposedFileName, TrendCmdTarget Property	1002
CompressData, TrendCmdTarget Property	1002
CurrentMultiplier, TrendCmdTarget Property	1003
CurrentSelectedPen, TrendCmdTarget Property	1004
CurrentTopPen, TrendCmdTarget Property.....	1004
CursorPos, TrendCmdTarget Property.....	1005
DataDefaultQuery, TrendCmdTarget Property.....	1005
DateFromCompareCurrent, TrendCmdTarget Property	1006
DateFromCurrent, TrendCmdTarget Property	1007
DataFileName, TrendCmdTarget Property	1007
DataFilterBy, TrendCmdTarget Property.....	1008
DataSortBy, TrendCmdTarget Property.....	1008
dateFrom, TrendCmdTarget Property	1009
dateFromCompare, TrendCmdTarget Property	1009
dateTo, TrendCmdTarget Property	1010
DateToCompareCurrent, TrendCmdTarget Property.....	1010
DateToCurrent, TrendCmdTarget Property	1011
DayBtnText, TrendCmdTarget Property	1012
dateToCompare, TrendCmdTarget Property.....	1012
DrawGridAfter, TrendCmdTarget Property	1013
ExpandBtnText, TrendCmdTarget Property.....	1013
ExtSettingsFile, TrendCmdTarget Property	1014
FontHeightLegend, TrendCmdTarget Property	1014
FontHeightScale, TrendCmdTarget Property	1015
FontHeightTime, TrendCmdTarget Property.....	1015
FontNameLegend, TrendCmdTarget Property	1015
FontNameScale, TrendCmdTarget Property	1016
FontNameTime, TrendCmdTarget Property.....	1016
FormatTime, TrendCmdTarget Property	1017
Freezed, TrendCmdTarget Property	1017
GeneralGap, TrendCmdTarget Property	1018
HourBtnText, TrendCmdTarget Property	1018
HourRecTime, TrendCmdTarget Property	1019
HourViewTime, TrendCmdTarget Property.....	1019
InvertDrawDirection, TrendCmdTarget Property	1020
LegendBrushColor, TrendCmdTarget Property	1021
LegendBrushVisible, TrendCmdTarget Property.....	1021
LinkedDataLogger, TrendCmdTarget Property.....	1021
MaxFileLength, TrendCmdTarget Property.....	1022
MaxLegendVisiblePen, TrendCmdTarget Property	1022
MaxNumFiles, TrendCmdTarget Property	1023
MeasureBtnText, TrendCmdTarget Property	1023
MeasureTextColor, TrendCmdTarget Property.....	1024
MinBtnText, TrendCmdTarget Property	1025
MinRecTime, TrendCmdTarget Property	1025
MinViewTime, TrendCmdTarget Property	1026

MonthBtnText, TrendCmdTarget Property	1026
MsecRecTime, TrendCmdTarget Property	1027
NetworkBackupServerName, TrendCmdTarget Property	1027
NetworkServerName, TrendCmdTarget Property	1028
NextBtnText, TrendCmdTarget Property	1028
NoneBtnText, TrendCmdTarget Property	1029
NumCacheRecordFile, TrendCmdTarget Property	1029
NumXGridDivision, TrendCmdTarget Property	1030
NumXMinorGridDivision, TrendCmdTarget Property	1030
NumYGridDivision, TrendCmdTarget Property	1031
NumYMinorGridDivision, TrendCmdTarget Property	1031
Page, TrendCmdTarget Property	1032
PageNextBtnText, TrendCmdTarget Property	1032
PagePrevBtnText, TrendCmdTarget Property	1033
PauseRunBtnText, TrendCmdTarget Property	1033
PenAutoscale, TrendCmdTarget Property	1034
PenAverageLineColor, TrendCmdTarget Property	1034
PenBackBrushPattern, TrendCmdTarget Property	1035
PenBrushColor, TrendCmdTarget Property	1035
PenBrushVisible, TrendCmdTarget Property	1036
PenColor, TrendCmdTarget Property	1036
PenDLColumnName, TrendCmdTarget Property	1037
PenDLRName, TrendCmdTarget Property	1038
PenEditable, TrendCmdTarget Property	1038
PenFormatScale, TrendCmdTarget Property	1039
PenLogarithmicScale, TrendCmdTarget Property	1039
PenMaxLineColor, TrendCmdTarget Property	1040
PenMaxValue, TrendCmdTarget Property	1040
PenMinLineColor, TrendCmdTarget Property	1041
PenMinValue, TrendCmdTarget Property	1041
PenPlotType, TrendCmdTarget Property	1042
PenScaleRightBottom, TrendCmdTarget Property	1042
PenShowAverageLine, TrendCmdTarget Property	1043
PenShowMaxLine, TrendCmdTarget Property	1043
PenShowMinLine, TrendCmdTarget Property	1044
PenShowScale, TrendCmdTarget Property	1044
PenSize, TrendCmdTarget Property	1045
PenStyle, TrendCmdTarget Property	1045
PenVariable, TrendCmdTarget Property	1046
PenWidth, TrendCmdTarget Property	1047
PrevBtnText, TrendCmdTarget Property	1047
PrintBtnText, TrendCmdTarget Property	1048
Recording, TrendCmdTarget Property	1048
RecordOnFile, TrendCmdTarget Property	1049
SampleDateTime, TrendCmdTarget Property	1049
SampleDateTimeMs, TrendCmdTarget Property	1050
SamplePerUpdate, TrendCmdTarget Property	1050
Samples, TrendCmdTarget Property	1051
SampleValue, TrendCmdTarget Property	1051
ScrollPosition, TrendCmdTarget Property	1052
SecBtnText, TrendCmdTarget Property	1053
SecRecTime, TrendCmdTarget Property	1053
SecViewTime, TrendCmdTarget Property	1054
ShiftGrid, TrendCmdTarget Property	1054
ShowBreakLines, TrendCmdTarget Property	1055
ShowCompareTimeFrameBtn, TrendCmdTarget Property	1055
ShowDate, TrendCmdTarget Property	1056
ShowExpandBtn, TrendCmdTarget Property	1056
ShowFirstPointBtn, TrendCmdTarget Property	1057
ShowLegend, TrendCmdTarget Property	1057
ShowLastPointBtn, TrendCmdTarget Property	1058
ShowMeasureBtn, TrendCmdTarget Property	1058
ShowMinorXGrid, TrendCmdTarget Property	1059
ShowMinorYGrid, TrendCmdTarget Property	1059
ShowMsec, TrendCmdTarget Property	1059
ShowNextBtn, TrendCmdTarget Property	1060
ShowNextPointBtn, TrendCmdTarget Property	1060
ShowPageNextBtn, TrendCmdTarget Property	1061
ShowPagePrevBtn, TrendCmdTarget Property	1062
ShowPauseRunBtn, TrendCmdTarget Property	1062
ShowPen, TrendCmdTarget Property	1063
ShowPenLabels, TrendCmdTarget Property	1063

ShowPenPoints, TrendCmdTarget Property	1064
ShowPrevBtn, TrendCmdTarget Property	1064
ShowPrevPointBtn, TrendCmdTarget Property	1065
ShowPrintBtn, TrendCmdTarget Property	1065
ShowSavedValues, TrendCmdTarget Property	1066
ShowTime, TrendCmdTarget Property	1066
ShowTimeFrameBtn, TrendCmdTarget Property	1067
ShowTitle, TrendCmdTarget Property	1067
ShowXGrid, TrendCmdTarget Property	1068
ShowYGrid, TrendCmdTarget Property	1068
ShowZoomBtn, TrendCmdTarget Property	1069
StartNewFile , TrendCmdTarget Property	1069
StatAverageValue, TrendCmdTarget Property	1070
StatMaxValue, TrendCmdTarget Property	1070
StatMinValue, TrendCmdTarget Property	1071
TimeBrushColor, TrendCmdTarget Property	1071
TimeBrushVisible, TrendCmdTarget Property	1072
TimeFrameBtnColor, TrendCmdTarget Property	1072
TimeTextColor, TrendCmdTarget Property	1073
TimeScale, TrendCmdTarget Property	1073
TrendQualityFreezeMode, TrendCmdTarget Property	1074
TrendBrushColor, TrendCmdTarget Property	1074
TrendBrushVisible, TrendCmdTarget Property	1075
TrendRunningType, TrendCmdTarget Property	1075
VariableAddValue, TrendCmdTarget Property	1076
VariableCursorPosIn, TrendCmdTarget Property	1077
VariableCursorPosOut, TrendCmdTarget Property	1077
VariableEnabling, TrendCmdTarget Property	1078
VariableFreezedMode, TrendCmdTarget Property	1078
VariableResetAllValues, TrendCmdTarget Property	1079
VariableScrollEnd, TrendCmdTarget Property	1079
VariableScrollNext, TrendCmdTarget Property	1080
VariableScrollNextPage, TrendCmdTarget Property	1081
VariableScrollPrev, TrendCmdTarget Property	1081
VariableScrollPrevPage, TrendCmdTarget Property	1082
VariableScrollStart, TrendCmdTarget Property	1082
Vertical, TrendCmdTarget Property	1083
ViewSamples, TrendCmdTarget Property	1083
Visible, TrendCmdTarget Property	1084
WeekBtnText, TrendCmdTarget Property	1084
XGridColor, TrendCmdTarget Property	1085
XGridLogarithmic, TrendCmdTarget Property	1085
XGridUseNormalLine, TrendCmdTarget Property	1086
XY, TrendCmdTarget Property	1086
YearBtnText, TrendCmdTarget Property	1087
YGridColor, TrendCmdTarget Property	1087
YGridLogarithmic, TrendCmdTarget Property	1088
YGridUseNormalLine, TrendCmdTarget Property	1088
ZoomBtnText, TrendCmdTarget Property	1089
ZoomMode, TrendCmdTarget Property	1089
1.17.9. UIInterface	1090
1.18. UTILIZZO DELLA UIINTERFACE	1090
Functions	1090
AlphaNumericEntry, UIInterface Function	1090
ChooseColor, UIInterface Function	1090
DoSomeEvents, UIInterface Function	1091
EditRuntimeUsers, UIInterface Function	1091
ExecuteCommand, UIInterface Function	1092
GetLastActiveSynoptic, UIInterface Function	1097
GetMonitorCoordinates, UIInterface Function	1097
GetNumMonitors, UIInterface Function	1098
GetPasswordFromLevel, UIInterface Function	1099
GetPasswordFromUser, UIInterface Function	1099
GetVariableNameFromList, UIInterface Function	1100
GetWindowPos, UIInterface Function	1100
HideLongOperationDialog, UIInterface Function	1101
IsLongOperationAborted, UIInterface Function	1102
LoadPicture, UIInterface Function	1102
LogoffActiveUser, UIInterface Function	1103
NumericEntry, UIInterface Function	1103
OpenModalSynoptic, UIInterface Function	1103

OpenSynoptic, UIInterface Function.....	1104
OpenSynopticEx, UIInterface Function	1105
OpenSynopticParameter, UIInterface Function	1105
SayThis, UIInterface Function	1106
SelectResourceFromList, UIInterface Function.....	1106
SetDefPrinterOrient, UIInterface Function.....	1107
SetRedraw, UIInterface Function	1108
SetWindowPos, UIInterface Function.....	1108
ShowHTMLDialog, UIInterface Function	1109
ShowLongOperationDialog, UIInterface Function	1110
ShowMenu, UIInterface Function	1110
Properties.....	1111
ActiveLanguage, UIInterface Property	1111
MainVisible, UIInterface Property	1111
1.18.1. UserAndGroupCmdTarget.....	1112
Functions	1112
GetActiveUserObject, UserAndGroupCmdTarget Function	1112
GetDesignGroupAtPos, UserAndGroupCmdTarget Function	1112
GetGroupObject, UserAndGroupCmdTarget Function.....	1113
GetNumActiveUsers, UserAndGroupCmdTarget Function	1114
GetNumDesignGroups, UserAndGroupCmdTarget Function	1115
GetNumRuntimeGroups, UserAndGroupCmdTarget Function	1115
GetRuntimeGroupAtPos, UserAndGroupCmdTarget Function	1116
GetUserObject, UserAndGroupCmdTarget Function.....	1117
GetXMLSettings, UserAndGroupCmdTarget Function	1117
LogonUser, UserAndGroupCmdTarget Function	1118
ReadRuntimeUsersXML, UserAndGroupCmdTarget Function	1118
SaveRuntimeUsersXML, UserAndGroupCmdTarget Function	1119
Properties.....	1120
AllowResizingForUsersBelowThisLevel, UserAndGroupCmdTarget Property	1120
AllowRuntimeChangesForUsersBelowThisLevel, UserAndGroupCmdTarget Property	1121
DefaultPrivAdminAccessLevel, UserAndGroupCmdTarget Property	1122
DefaultPrivAdminLevel, UserAndGroupCmdTarget Property	1122
DefaultPrivGuestAccessLevel, UserAndGroupCmdTarget Property	1123
DefaultPrivGuestLevel, UserAndGroupCmdTarget Property	1123
DefaultPrivUserAccessLevel, UserAndGroupCmdTarget Property	1124
DefaultPrivUserLevel, UserAndGroupCmdTarget Property	1125
EnableAutoLogoff, UserAndGroupCmdTarget Property.....	1125
EnableNTUserLogin, UserAndGroupCmdTarget Property	1126
EnableRuntimeUsers, UserAndGroupCmdTarget Property	1126
EnableRuntimeUsersSecurity, UserAndGroupCmdTarget Property	1127
MinimumPasswordLength, UserAndGroupCmdTarget Property.....	1127
MinimumUserLength, UserAndGroupCmdTarget Property.....	1128
RuntimeUserAccessCode, UserAndGroupCmdTarget Property	1128
SecsAutoLogoffTimeout, UserAndGroupCmdTarget Property	1129
1.18.2. UserCmdTarget.....	1130
Functions	1130
GetDaysBeforePasswExpires, UserCmdTarget Function.....	1130
GetGroupObject, UserCmdTarget Function.....	1130
GetLastTimeUserAction, UserCmdTarget Function.....	1131
GetListAccessVariable, UserCmdTarget Function.....	1132
GetProp, UserCmdTarget Function	1132
GetXMLSettings, UserCmdTarget Function	1133
IsRemote, UserCmdTarget Function.....	1134
ResetListAccessVariables, UserCmdTarget Function	1134
SetProp, UserCmdTarget Function	1135
Properties.....	1136
AccessLevel, UserCmdTarget Property	1136
AccountDisabled, UserCmdTarget Property	1137
CannotChangePassword, UserCmdTarget Property.....	1137
CommandListLogoff, UserCmdTarget Property	1138
CommandListLogon, UserCmdTarget Property.....	1138
Description, UserCmdTarget Property	1139
Email, UserCmdTarget Property	1140
EnableAutoLoggoff, UserCmdTarget Property	1140
ExpiringDaysPassword, UserCmdTarget Property.....	1141
FaxAreaCode, UserCmdTarget Property	1142
FaxCountryCode, UserCmdTarget Property	1142
FaxPhoneNumber, UserCmdTarget Property.....	1143
Language, UserCmdTarget Property.....	1143
Level, UserCmdTarget Property.....	1144

Locked, UserCmdTarget Property	1145
LogoffScript, UserCmdTarget Property	1145
LogonScript, UserCmdTarget Property	1146
MobileAreaCode, UserCmdTarget Property	1146
MobileCountryCode, UserCmdTarget Property	1147
MobilePhoneNumber, UserCmdTarget Property	1148
MustChangedPasswordLogon, UserCmdTarget Property	1148
Name, UserCmdTarget Property	1149
OnLine, UserCmdTarget Property	1149
Password, UserCmdTarget Property	1150
SecsAutoLoggoffTimeout, UserCmdTarget Property	1151
VoiceAreaCode, UserCmdTarget Property	1151
VoiceCountryCode, UserCmdTarget Property	1152
VoicePhoneNumber, UserCmdTarget Property	1153
WebClientAutoLogoffSecs, UserCmdTarget Property	1153
<i>1.18.3. UserGroupCmdTarget</i>	<i>1154</i>
Functions	1154
GetNumUsers, UserGroupCmdTarget Function	1154
GetUserAtPos, UserGroupCmdTarget Function	1155
GetUserObject, UserGroupCmdTarget Function	1156
GetXMLSettings, UserGroupCmdTarget Function	1157
Properties	1158
CommandListLogoff, UserGroupCmdTarget Property	1158
CommandListLogon, UserGroupCmdTarget Property	1158
DefaultAccessLevel, UserGroupCmdTarget Property	1159
DefaultEnableAutoLoggoff, UserGroupCmdTarget Property	1159
DefaultExpiringDaysPassword, UserGroupCmdTarget Property	1160
DefaultLevel, UserGroupCmdTarget Property	1161
DefaultLogoffScript, UserGroupCmdTarget Property	1162
DefaultLogonScript, UserGroupCmdTarget Property	1162
DefaultSecsAutoLoggoffTimeout, UserGroupCmdTarget Property	1163
Description, UserGroupCmdTarget Property	1163
Language, UserGroupCmdTarget Property	1164
Name, UserGroupCmdTarget Property	1165
WebClientAutoLogoffSecs, UserGroupCmdTarget Property	1165
<i>1.18.4. WorkspaceCmdTarget</i>	<i>1166</i>
Functions	1166
OpenProject, WorkspaceCmdTarget Function	1166
OpenScreen, WorkspaceCmdTarget Function	1166
OpenScript, WorkspaceCmdTarget Function	1167

1. Linguaggio VBA

1.1. Introduzione

Contenuto della Guida alla programmazione dei Linguaggi Basic Script di Movicon.

Il contenuto della Guida alla programmazione dei Linguaggi script Basic VBA™ compatibile (Visual Basic for Applications™) contiene tutte le informazioni necessarie allo sviluppatore per la realizzazione di routines "Basic Script" di Movicon.

La guida comprende una parte relativa alle istruzioni inerenti ai comandi specifici per Movicon, ed una seconda parte contenente la sintassi delle istruzioni standard compatibile VBA™, facente parte della guida "WinWrap Basic Language" incorporata in Movicon.

1.1.1. Premesse

Tutte le informazioni contenute nella documentazione di Movicon presuppongono:

- Windows si riferisca all'apposito marchio registrato da Microsoft inc. nelle versioni Windows 32/64 bit
- Movicon si riferisca al sistema di supervisione sviluppato da Progea tutelato dalle leggi internazionali sul Copyright
- VBA si riferisca a Microsoft Visual Basic for Application. Il Linguaggio VBA di Movicon è compatibile VBA al 100%
- Ms Access e SQL Server si riferiscano ai prodotti registrati da Microsoft inc
- Ogni altro prodotto o marchio citato sia registrato o tutelato dal relativo proprietario

1.2. Concetti Generali

1.2.1. Basic Script nel Progetto

1.3. Basic Script nel Progetto

All'interno di un progetto è possibile disporre delle funzioni basic script in diverse circostanze e modalità.

E' bene farne utilizzo in quei frangenti in cui le stesse operazioni non siano disponibili con altre risorse o metodi: logica generale, proprietà di esecuzione dei disegni, Eventi. L'utilizzo incondizionato dei basic script all'interno di un progetto potrà essere molto comodo in fase di progettazione ma potrà rendere meno veloce e più dispendiosa di risorse l'esecuzione del progetto.

Il codice basic può essere utilizzato in più punti del progetto: come risorsa, direttamente nelle proprietà di esecuzione di un oggetto disegno, come codice associato agli eventi di un oggetto (allarme, disegno o simbolo), nei sinottici.

Nelle apposite sezioni sono spiegati in dettaglio queste funzionalità.

1.3.1. WinWrap Basic Language

Movicon integra al suo interno un componente software, WinWrap Basic, che permette di integrare all'interno dell'applicazione delle routine in linguaggio Basic.

Di seguito vengono elencate alcune delle principali caratteristiche del WinWrap Basic:

- permette di creare delle routine in linguaggio VBA (Visual Basic for Application) compatibili con il codice BASIC (Beginners All-purpose Symbolic Instruction Code)
- permette di estendere il set di istruzioni con funzioni e metodi personalizzati
- permette di creare delle dialog box utilizzando i controlli standard di windows (pulsanti, checkboxes, groupboxes, listboxes, option buttons, immagini e caselle di testo)
- supporta i controlli ActiveX
- permette di dichiarare e richiamare le funzioni API e WMI di windows



Attenzione! la definizione Declare del winwrap basic non è supportata su sistemi CE.

1.4. Supporto VB.NET e Unicode

Supporto al VB.NET

Usando la chiave speciale **'#Language "WWB.NET"** è possibile passare alla modalità di programmazione di .NET. Questo consente di accedere direttamente alle .NET assemblies con il codice VB. La finestra di dialogo per aggiungere gli oggetti di riferimento da usare nel codice tiene conto di questa chiave. Nel caso sia impostato l'uso del linguaggio .NET, allora sarà possibile vedere l'elenco di tutti gli assemblies selezionabili.

Esempio di codice:

```
'#Reference #System.Windows.Forms, Version=2.0.0.0, Culture=neutral,
PublicKeyToken=b77a5c561934e089, processorArchitecture=MSIL

'#Language "WWB.NET"

Dim WithEvents t As System.Windows.Forms.Timer

Sub Main
    t = New System.Windows.Forms.Timer
    t.Interval = 1000
    t.Enabled = True
    Wait 1
End Sub

Private Sub t_Tick(ByVal sender As Object, ByVal e As System.EventArgs) Handles
t.Tick
    Debug.Print Now
End Sub
```

A tale scopo sono disponibili tutta una serie di istruzioni nuove per rendere il codice compatibile alla programmazione VB.NET.

Limitazioni:

- La programmazione .NET non è supportata nella versione Windows CE
- Le Subroutines negli oggetti non possono essere rinominate così come consentirebbe la programmazione .NET
- Non è possibile collegare più eventi alla stessa procedura usando una lista di "Handles"
- Non è possibile gestire eventi sulla modifica del valore di una variabile del progetto in uno script WWB.NET
- Nella versione 9 del WinWrap Basic, il basic script è single thread e il codice script deve essere eseguito dal thread che l'ha creato. Ad esempio non è possibile passare un delegate di .NET ad un oggetto che gestirà la chiamata di questo delegate in un altro thread. Un esempio concreto sono alcune delle funzioni della classe "System.IO.FileSystemWatcher" che creano dei thread aggiuntivi per monitorare le modifiche dei file. In questo caso i metodi di tale classe non sono supportati in quanto richiamano altri thread. La stessa cosa vale anche per la classe "System.Net.Sockets"

- Nella versione 9 del WinWrap Basic, il basic script non supporta i "nested types", cioè un tipo definito nell'ambito di un altro tipo. Ad esempio non è possibile usare le funzioni del tipo "System.Net.WebRequestMethods.Ftp".

Supporto Unicode

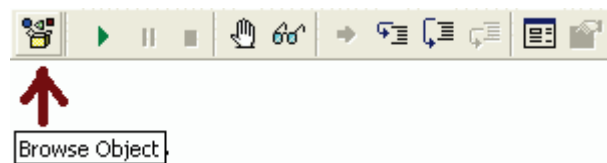
L'editor del codice consente di inserire delle stringhe in formato Unicode. Quindi è possibile visualizzare nelle finestre di dialogo basic script delle stringhe Unicode oppure assegnare a delle variabili stringa del progetto testi Unicode.

Esiste la possibilità di leggere e/o scrivere file Unicode UTF8 o UTF16. A tale scopo sono state aggiunte due nuove costanti "vbUTF8BOM" e "vbUTF16BOM". Scrivendo una di questa costanti nel primo carattere di un file di testo ne verrà determinata la codifica.

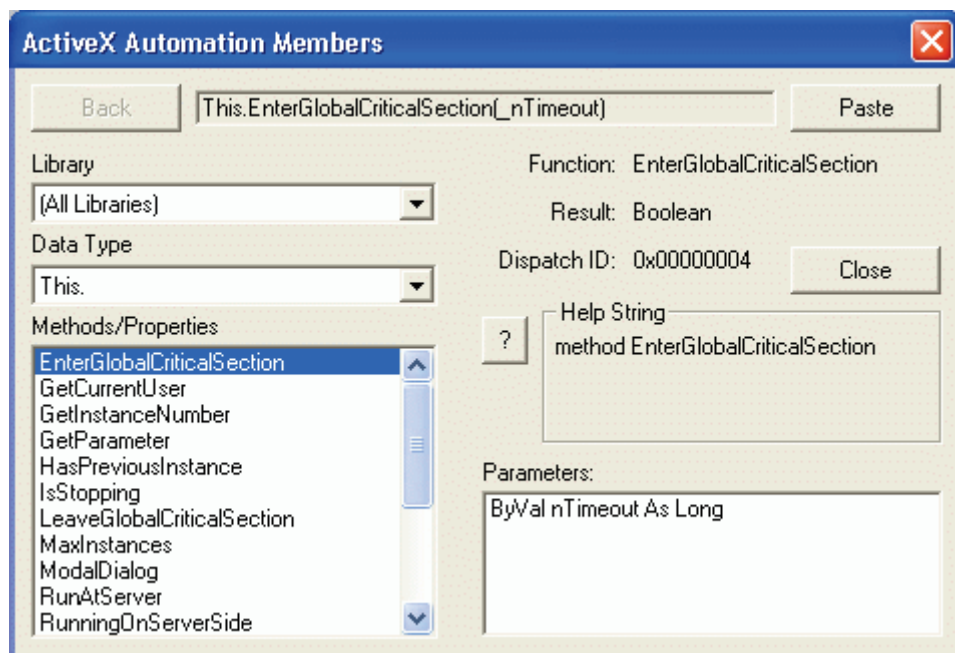
1.4.1. Librerie Basic Script

Oltre alle funzioni base messe a disposizione dal WinWrap Basic, all'interno dei Basic Script è possibile utilizzare una serie di funzioni supplementari che fanno parte delle librerie di Movicon e che permettono di interagire con il progetto. Queste funzioni permettono per esempio di leggere e scrivere le variabili del Real Time DB di Movicon, di eseguire dei cambi pagina, di interagire con le proprietà dei simboli di Movicon, ecc. Queste librerie di funzioni di Movicon vengono definite come Librerie di Interfaccia Basic, e ogni interfaccia raccoglie una serie di funzioni specifiche per un determinato componente del progetto.

Le interfacce basic di Movicon sono numerose, e per accedere all'elenco delle funzioni di queste librerie si può utilizzare il pulsante "Browse" delle funzioni della barra utensili dei basic (disponibile dopo avere aperto il codice basic di un qualsiasi componente):



La finestra che si apre riporta il nome dell'interfaccia nella casella "Data Type" e la lista dei metodi e proprietà relativi all'interfaccia nella lista "Methods/Proprieties":



E' possibile selezionare un'altra interfaccia agendo sulla lista "Data Type".

Per ottenere l'help di una funzione è sufficiente selezionare la funzione e premere il punto interrogativo se ci si trova nella finestra del Browse delle funzioni, oppure se si è all'interno del codice basic è sufficiente selezionare la funzione e premere il tasto F1. Il significato dei vari campi della Finestra di Browse è il seguente:

Back

Ritorna di un passo indietro nella scala gerarchica della libreria. Alcune librerie definiscono degli oggetti ai quali sono associate delle proprietà e dei metodi, se stiamo visualizzando le proprietà di un oggetto con un clic sul pulsante back si ritornerebbe all'elenco degli oggetti definiti nella libreria.

Paste

Copia il contenuto della casella a lato, nella posizione in cui si trova il cursore sulla finestra di editor dello Script. Se il comando è disabilitato indica l'impossibilità di copiare il contenuto nella posizione in cui il cursore si trova.

Close

Chiude la finestra di browse e ritorna attivo l'editor dello Script.

Library

Permette di selezionare una delle librerie dall'elenco proposto. Se si aggiungono ActiveX/OCX attraverso i references questi sono disponibili nell'elenco.

Data Type

Permette di selezionare un tipo di dato dall'elenco proposto. L'elenco si riferisce alla libreria selezionata nella casella Library.

Methods/Properties

Permette di selezionare un metodo o una proprietà dall'elenco proposto. L'elenco si riferisce al tipo di dato selezionato nella casella Data Type.

?

Accede all'help della proprietà o del metodo selezionato nella casella a fianco. Alcune librerie esterne, non definite da Progea e abilitate attraverso i references, non installano i loro file di help.

Parameters

Viene visualizzato l'eventuale elenco di parametri definiti per il metodo selezionato nella casella Data Type.

1.4.2. Sub, Function, Eventi, Metodi e Proprietà

Sub e Function

All'interno dei basic script si possono creare delle Routine, ovvero porzioni di codice racchiuse in un blocco, che possono essere fondamentalmente di due tipi:

- Sub
- Function

la differenza tra questi due tipi di routine è in realtà minima. Entrambe possono essere richiamate passando loro dei parametri. La "Function" però a differenza della "Sub" può restituire un valore di tipo definito dal programmatore (Bool, Int, String, ecc.)

Esempio 1:

All'interno della Sub Main di un Basic Script viene richiamata la Sub Prova:

```
Sub Main
    Call Prova()
End Sub

Sub Prova()
    MsgBox("Prova Sub", vbInformation + vbOkOnly, GetProjectTitle)
End Sub
```

Esempio 2:

All'interno della Sub Main di un Basic Script viene richiamata la Function Prova e viene stampato il risultato del tasto premuto dall'utente sulla MsgBox:

```
Sub Main
    Debug.Print Prova()
End Sub

Function Prova() As String
```

```

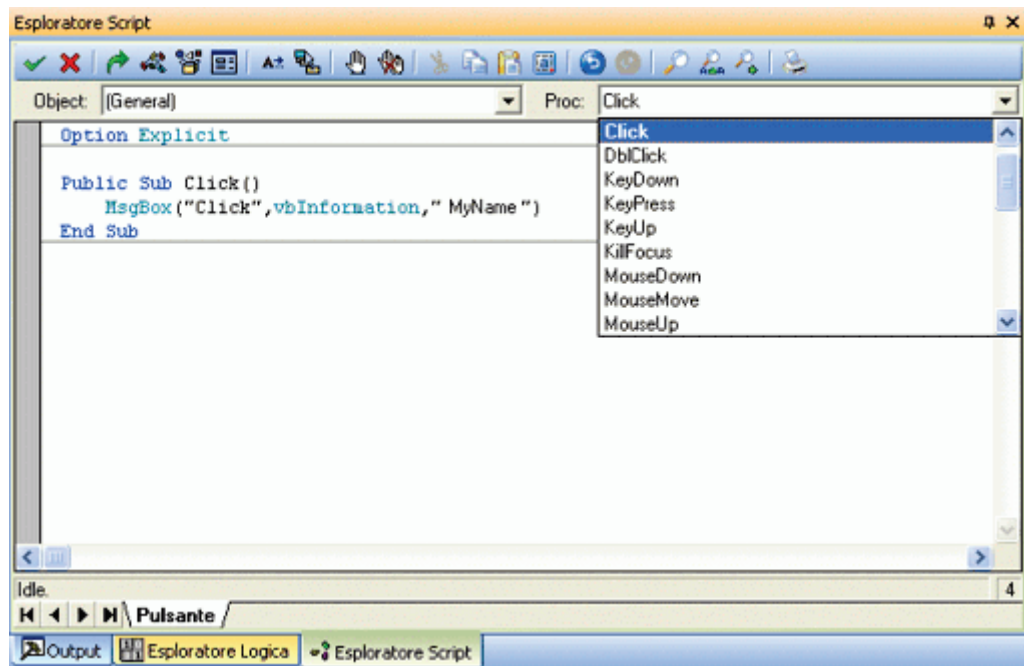
If MsgBox("Prova Function",vbInformation + vbOkCancel,GetProjectTitle) = vbOK
Then
    Prova = "OK"
Else
    Prova = "Annulla"
End If
End Function

```

Le routine sopra descritte possono essere richiamate da altre routine, quindi si creano delle chiamate annidate tra loro. Nel caso della Risorsa Basic Script però dovrà sempre esistere la "**Sub Main**" che è la routine che viene eseguita automaticamente dal Basic Script quando questo viene messo in esecuzione. Sarà poi cura del programmatore inserire il codice opportuno all'interno di questa routine ed eventualmente richiamare altre Sub o Function create dal programmatore stesso. Una volta che le istruzioni contenute all'interno della Sub Main() sono state tutte eseguite il basic script viene terminato e si dovrà richiamarlo nuovamente per eseguirlo un'altra volta. E' possibile ad ogni modo inserire dei loop all'interno della Sub Main() in modo da mantenere il Basic Script sempre in esecuzione.

Eventi

Esistono delle routine di tipo Sub, messe a disposizione del sistema (Movicon o eventuali componenti ActiveX inseriti) che vengono richiamate automaticamente dal sistema a fronte di determinati eventi. Queste routine, che vengono appunto definite "Eventi", possono essere inserite negli script e sarà cura del programmatore aggiungere al loro interno il codice desiderato.



Esempio:

Inserendo il seguente codice all'interno dello script di un disegno rettangolo, ogni volta che si eseguirà un Click col mouse sul rettangolo apparirà una MsgBox con la scritta "Click":

```

Public Sub Click()
    MsgBox("Click",vbInformation,GetProjectTitle)
End Sub

```

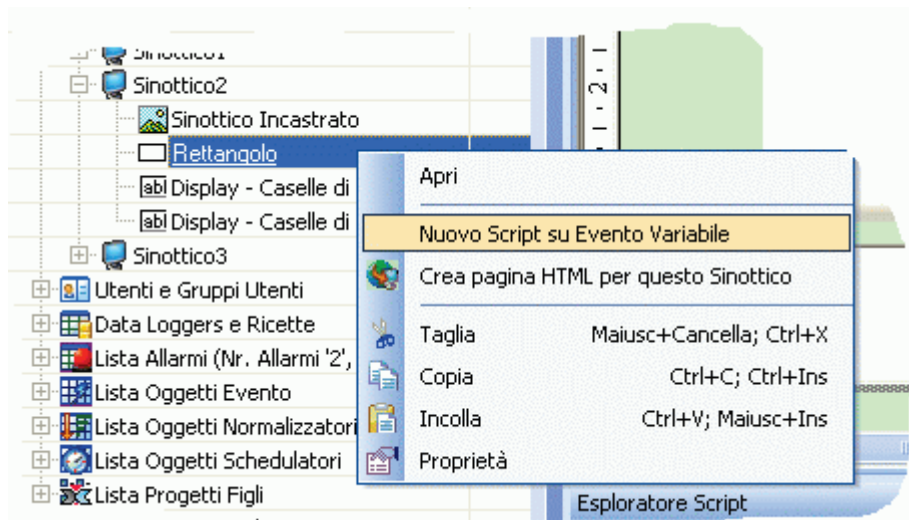
In questo caso la Public Sub Click() è un evento scatenato dal sistema quando si esegue un click sul rettangolo, mentre il codice al suo interno deve essere inserito dal programmatore.

All'interno del codice basic dei simboli, dei sinottici e degli allarmi si possono selezionare una serie di eventi resi disponibili dal sistema. Questi eventi riguardano esclusivamente il componente sul quale si sta editando il codice, ovvero gli eventi in questione riguardano eventi scatenati dal componente. Questi eventi sono elencati nella lista "Proc:" quando è selezionata la voce "(General)" nella lista "Object", come mostrato nella figura sopra.

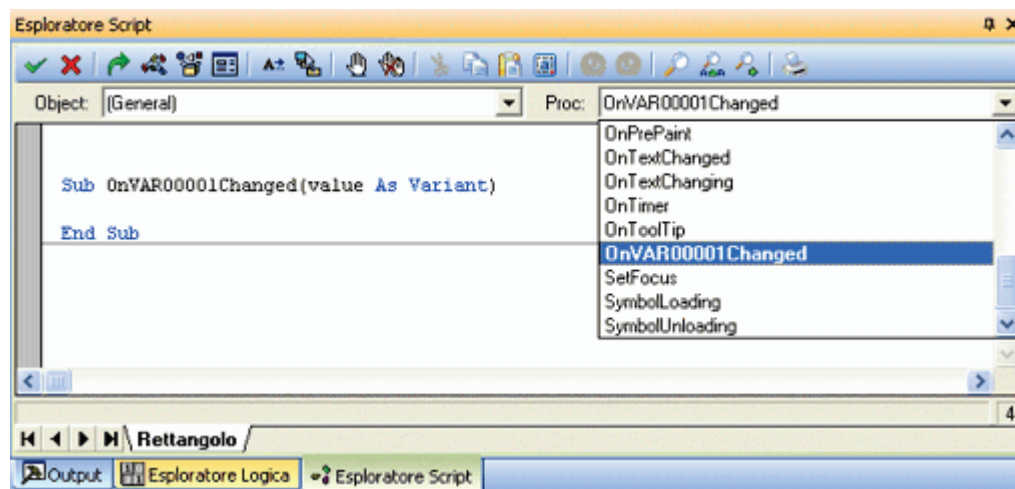
Per la categoria dei Simboli e Disegni e per i Sinottici è possibile creare degli eventi personalizzati su specifiche variabili del Real Time DB di Movicon. Ad esempio è possibile inserire un evento che viene richiamato ogni volta che una determinata variabile del progetto cambia di stato. E' una procedura

molto comoda per tenere monitorato il cambiamento di stato di una variabile senza appesantire troppo il sistema. La procedura per creare un evento di questo tipo è la seguente:

- dalla finestra "Esploratore Progetto" cliccare col tasto destro del mouse sul componente o sul sinottico al quale si vuole associare l'evento, e dal menù contestuale che verrà aperto selezionare il comando "Nuovo Script su Evento Variabile":



- a questo punto verrà aperta la finestra di Browse delle variabili del progetto dal quale si dovrà selezionare la variabile interessata. Una volta selezionata la variabile all'interno del codice script del componente o del sinottico verrà automaticamente inserito un nuovo evento nella lista denominato come "OnNomeVariabileChanged" (On + nome variabile + Changed. ad esempio se si aggiunge la variabile VAR00001, l'evento diventerà "OnVAR00001Changed").



Inserendo questo nuovo evento nel codice questo verrà richiamato ogni volta che la variabile "VAR0001" subirà un cambiamento del suo valore e il parametro "value" riporterà il valore attuale della variabile. Naturalmente è possibile associare più variabili ad ogni componente.



Attenzione! Gli Eventi delle interfacce basic non vengono eseguiti se contengono uno spazio fra il nome della sub e l'inizio dei parametri.

Ad esempio:

Public Sub OnSetAlarm (bSet As Boolean, bRet As Boolean) -> Non viene eseguita

Public Sub OnSetAlarm(bSet As Boolean, bRet As Boolean) -> Viene eseguita

Metodi e Proprietà

Le funzioni che si possono inserire dalle interfacce basic di Movicon (o da altre librerie esterne come ActiveX/OCX) possono essere definite come "Metodi" e "Proprietà". Le proprietà identificano una

caratteristica dell'oggetto in questione, ad esempio il colore, il titolo, ecc., mentre il metodo esegue una funzione di quell'oggetto.

1.4.3. Variabili nei Basic Script

All'interno dei Basic Script di Movicon possono essere utilizzate sia le variabili del Real Time DB di Movicon, che risulteranno essere comunque variabili globali per tutto il progetto, sia le variabili locali del Basic Script che possono essere visibili solo all'interno della routine Basic e che vengono distrutte una volta terminato il Basic Script.

Le variabili del progetto (Real Time DB di Movicon) possono essere utilizzate direttamente con il loro nome oppure utilizzando le apposite funzione della "DBVariableCmdTarget" (GetVariableValue(), SetVariableValue()). Nel caso delle variabili struttura, quando vengono utilizzate in modo diretto, il carattere di delimitazione fra il nome della variabile e il nome del membro non sono più i due punti (":") ma l'underscore ("_") oppure il punto ("."). Per utilizzare i due punti è necessario racchiudere la variabile tra parentesi quadre ([Struttura:Membro]). Il carattere standard dei due punti viene invece utilizzato in tutte le funzioni basic che richiedono una variabile come parametro (ad esempio GetVariableValue(), SetVariableValue(), ecc.).

Per maggiori informazioni sulla sintassi e l'uso delle variabili di tipo Struttura nei Basic Script vedere il capitolo "Puntamento a Variabili Struttura".



Non è possibile utilizzare in modo diretto, con il solo nome, variabili dichiarate nel Real Time DB del progetto che iniziano con un numero o con un carattere underscore ("_"). Il motivo sta nel fatto che l'interprete VBA non permette dichiarazioni di variabili di questo tipo. E' necessario in questi casi utilizzare sempre le funzioni GetVariableValue(), SetVariableValue().



Le variabili interne del Basic Script hanno priorità su quelle del progetto. Quindi utilizzando il nome di una variabile di progetto direttamente nel codice basic script, se esiste una variabile interna al basic con lo stesso nome, viene impostata o letta questa e non quella di progetto. In questo caso occorre appoggiarsi alle funzioni basic GetVariableValue(), SetVariableValue().

Le variabili di tipo bit, quando sono utilizzate direttamente con il loro nome, sono convertite in boolean (true o false). Questo significa che all'interno del Basic Script una variabile del progetto che ha valore "1" verrà letta come variabile "true" e quindi con un valore numerico uguale a "-1". Per quanto riguarda il valore "0" che viene interpretato dal Basic Script come "false" risulta essere sempre un valore numerico uguale a "0". Questo meccanismo è valido sia per la lettura che per la scrittura delle variabili di tipo bit all'interno dei Basic Script. Per avviare a questo meccanismo è sempre possibile utilizzare le funzioni basic GetVariableValue() e SetVariableValue() per leggere e scrivere le variabili di tipo bit. In questo modo il valore di ritorno della funzione sarà il valore numerico del bit e non il valore booleano, quindi "1" e "0".

Esempio:

volendo leggere la variabile bProva dichiarata nel DataBase di Movicon come bit e impostata al valore "1", risulterà:

```
Sub Main()  
    Debug.Print bProva 'Risultato = -1  
    Debug.Print GetVariableValue("bProva") 'Risultato = 1  
End Sub
```

Utilizzo dell'IntelliSense per le Variabili del RealTimeDB

L'IntelliSense del VBA può essere popolato anche con le variabili globali del progetto (le variabili del RealTimeDB). Questo consente di recuperare una variabile usando la lista popup che appare sulla pressione dei tasti "Ctrl+Spazio" e di accedere in modo diretto anche alle variabili di tipo Struttura e Array di Byte.

Nelle proprietà di una variabile è stata aggiunta un'opzione chiamata "IntelliSense" che serve appunto per abilitare la variabile ad essere aggiunta nell'intellisense del VBA. Di default tutte le variabili vengono aggiunte senza questa opzione, che può però essere marcata per tutte quelle variabili che si vogliono avere a disposizione nell'IntelliSense.

All'interno del codice basic script è possibile accedere agli elementi di una variabile di tipo "Array di Byte" o ai membri di una "Variabile Struttura" (se la proprietà "IntelliSense" è stata abilitata) definite nel RealTimeDB del progetto utilizzando la seguente sintassi:

NomeVarArray.eIndiceMembro (dove IndiceMembro va da "0" a "dim. array -1". Es. "Motore.e5" indica il sesto elemento dell'array "Motore")
NomeVarStruttura.NomeMembro



LIMITAZIONI: L'IntelliSense non riporta mai le variabili locali di un sinottico, e non viene popolato con le variabili globali se usato nella finestra "Esploratore Proprietà Dinamiche".



E' buona norma Impostare l'opzione "IntelliSense" soltanto per quelle variabili per cui si ritiene necessario. Infatti l'operazione di popolamento dell'IntelliSense in apertura di uno script o sull'esecuzione dello stesso, potrebbe essere rallentata nel caso in cui il numero di variabili con l'opzione "IntelliSense" sia dell'ordine di decine di migliaia di variabili.

Variabili senza segno

Il WinWrap non prevede l'utilizzo di variabili senza segno quindi una variabile del RealTimeDB di tipo WORD (senza segno) o DWORD (senza segno) per essere utilizzata nel Basic Script deve essere convertita al fine di non causare un errore di tipo overflow. L'errore di overflow infatti si manifesterebbe quando la variabile usata nello script supera il valore di tipo INT (ad esempio 32767 per una word) perchè il motore basic non gestisce le variabili di tipo senza segno. **Per evitare questo problema una variabile di tipo WORD viene passata al WinWrap sempre come Long e una variabile di tipo DWORD viene passata al WinWrap di tipo "Hauge_" che significa un "64 bit integer value".**

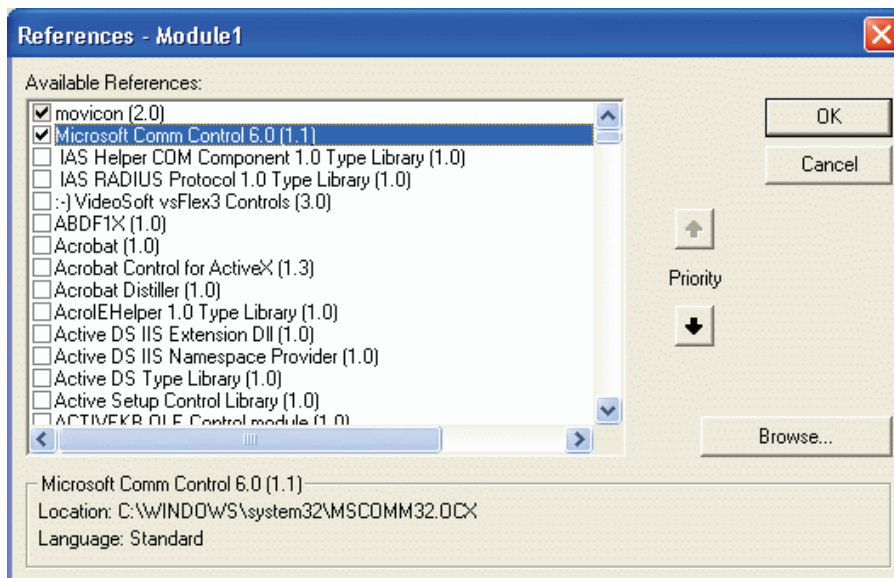


Per quanto riguarda l'ambiente WinCE si possono gestire le variabili di tipo WORD ma non è possibile gestire le variabili di tipo DWORD (quando superano il valore di un Long) perchè WinCE non ha il supporto al tipo di dato "VT_I8".

1.4.4. Programmazione Rapida

Durante la digitazione del codice di uno script si possono utilizzare alcuni tasti per velocizzare l'inserimento delle funzioni. Ad esempio premendo i tasti "CTRL+Spazio" verrà aperta una lista a tendina di tutte le funzioni (del WinWrap Basic o di Movicon) che si possono inserire nel contesto di quello script.

Quando si utilizzano delle variabili di tipo Oggetto, e quindi vengono da queste messi a disposizione una serie di metodi e proprietà, è sufficiente scrivere il nome dell'Oggetto seguito dal "." (punto) per vedere la lista di queste funzioni. Per quanto riguarda gli ActiveX/OCX, questo tipo di meccanismo funziona solo se nei "references" del basic è stata abilitata la libreria che contiene il controllo in questione e l'Oggetto viene dichiarato non come "Object" generico ma con il nome della classe che viene implementata. Ad esempio volendo accedere alle proprietà di un ActiveX MsComm bisognerà abilitare il reference "Microsoft Comm Control 6.0".



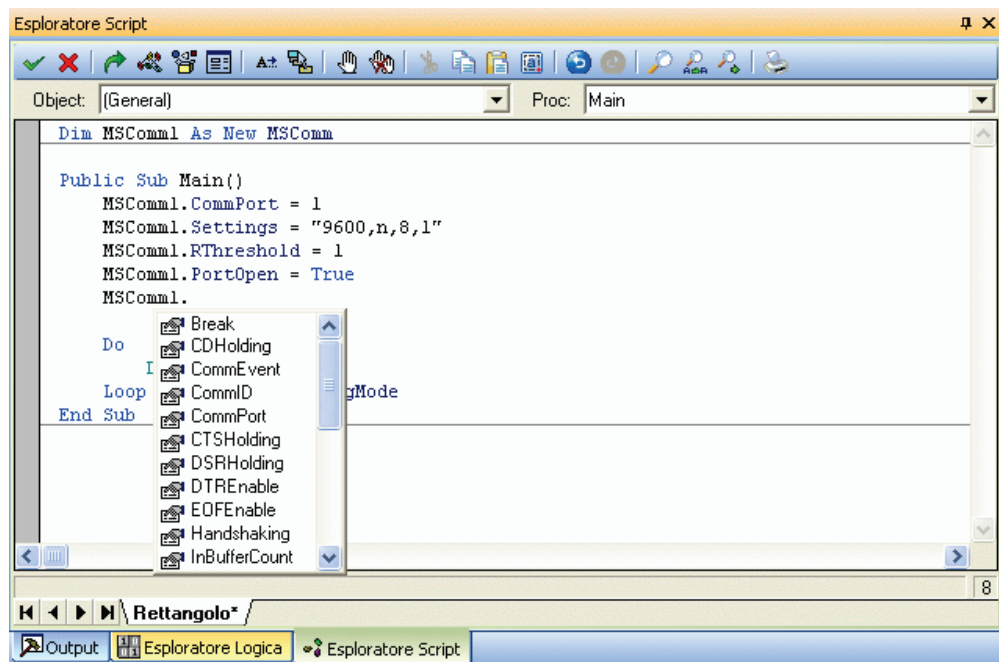
poi all'interno del Basic Script si dovrà dichiarare l'oggetto come segue:

```
Dim MSComm1 As New MSComm

Public Sub Main()
    MSComm1.CommPort = 1
    MSComm1.Settings = "9600,n,8,1"
    MSComm1.RThreshold = 1
    MSComm1.PortOpen = True

    Do
        DoEvents
    Loop Until IsInStoppingMode
End Sub
```

dichiarando la variabile MSComm1 nel modo sopra riportato, ovvero non come Object ma come MSComm (classe dell'activex da utilizzare), digitando il nome Oggetto seguito dal punto apparirà la seguente finestra:



Per maggiori chiarimenti sull'utilizzo di ActiveX/OCX all'interno dei Basic Script si rimanda al paragrafo "ActiveX/OCX nei Basic Script".

1.5. Codice nei Basic Script e nei Simboli

Esiste una fondamentale differenza di funzionamento tra le risorse Basic Script e il codice che si inserisce all'interno dei simboli, dei sinottici e degli allarmi. Nei Basic Script deve esistere la "**Sub Main**" che viene eseguita al richiamo del Basic Script. Nel caso invece dei simboli (o sinottici o allarmi) non esiste una Sub Main, ma la gestione del codice è fatta esclusivamente in base agli eventi (SymbolLoading, Click, ecc.) disponibili all'interno del codice e che sono stati inseriti dal programmatore. Movicon richiamerà tali eventi soltanto finché il simbolo (o sinottico) saranno attivi, quindi quando il sinottico è caricato in ram.

1.5.1. Basic Script come Risorsa

Se si desidera inserire un Basic Script nel progetto, occorre procedere all'inserimento di una nuova risorsa. Per l'inserimento di una nuova risorsa Basic Script, selezionare prima il punto desiderato nella

struttura ad albero del gruppo "Basic Script" della finestra "Esploratore Progetto", quindi cliccando con il pulsante destro del mouse selezionare il comando "Nuovo Script" dal menù contestuale che verrà aperto.

Alla conferma dell'operazione, apparirà nel gruppo o nel punto selezionato nella struttura delle Risorse, la nuova risorsa Basic Script, e verrà aperta la finestra di editor del codice relativa al Basic Script appena inserito. A questo punto è possibile procedere all'introduzione del codice VBA™ comp. come descritto nei paragrafi specifici.

Alla risorsa potrà successivamente essere assegnato il Nome utilizzando il mouse facendo clic sulla risorsa e digitando il nome in sostituzione a quello provvisorio.

Una risorsa Basic Script deve contenere al suo interno la procedura Main (Sub Main). Le istruzioni contenute in tale subroutine vengono eseguite nel momento in cui il basic è lanciato dalle logiche del progetto. Al termine della subroutine, in assenza di cicli loop programmati, il basic viene terminato ed è pronto per un successivo richiamo.

La procedura Main non presenta dei parametri configurabili ma è possibile associare dei parametri al momento della chiamata del Basic Script. Nel comando che esegue il richiamo del basic script dovranno essere indicati i parametri interponendo fra ciascuno di loro il carattere "virgola", e all'interno del basic script la funzione GetParameter() potrà essere utilizzata per leggere il valore degli ultimi parametri con cui il basic è stato richiamato.

Dopo che una risorsa Basic Script è stata messa in esecuzione la prima volta, anche dopo che la Sub Main() è terminata, la risorsa rimane attiva, quindi eventuali eventi definiti al suo interno saranno eseguiti all'occorrenza. Ad esempio è possibile inserire all'interno del Basic Script un "Evento su Variabile", che verrà eseguito al cambiamento della variabile anche se il Basic Script ha già terminato la Sub Main(). Se si vuole invece terminare completamente l'esecuzione di un Basic Script è necessario eseguire il comando di "Stop" disponibile ad esempio nella "Lista Comandi" di Movicon. I due eventi "Loading" e "Unloading" del Basic Script vengono richiamati rispettivamente al primo avvio dello Script e alla sua terminazione (tramite il comando di "Stop" o alla chiusura del progetto).



Il comando di stop di una risorsa basic script scarica dalla memoria soltanto quelle risorse basic che sono in esecuzione in un thread separato. Le altre risorse basic script che sono in esecuzione nello stesso thread vengono solo fermate. Ne consegue che l'evento "Unloading" non viene più eseguito a seguito di un comando di stop per quelle risorse basic script che non sono in thread separato. Inoltre le variabili utilizzate dal basic script rimangono sempre in uso una volta che il basic script è stato eseguito almeno una volta.

Una risorsa Basic Script introdotta nel progetto può essere mandata in esecuzione in vari modi, in funzione delle necessità. I paragrafi seguenti ne illustrano le metodologie.

Esecuzione su comando

L'esecuzione su comando è facilmente impostabile tramite le proprietà "**Comandi**" degli oggetti, dei menù o delle risorse che prevedono la possibilità di eseguire dei comandi.

Ad esempio, se ad un pulsante si desidera associare l'esecuzione di una routine Basic Script, occorre selezionare dalla proprietà "Comandi" il tipo di comando "Comandi Script", selezionando la risorsa Basic desiderata tra la lista proposta riferita a quelle introdotte, e definendo poi le altre impostazioni come desiderato.

L'esecuzione su comando di un Basic Script può essere eseguita anche dalla risorsa "**Lista Oggetti Evento**", quindi di fatto l'avvio della routine basic non verrà gestita da un comando da operatore, ma sarà eseguita a fronte di un determinato evento.

Esecuzione allo startup

E' possibile eseguire una routine Basic Script automaticamente all'avvio in runtime del progetto applicativo. Per utilizzare questa funzionalità, occorre accedere all'apposita impostazione "**Script di Avvio**" dalle "Impostazioni Esecuzione del Progetto".

Attraverso l'apposita finestra di selezione, occorre selezionare la risorsa Basic Script tra quelle precedentemente inserite nel progetto.

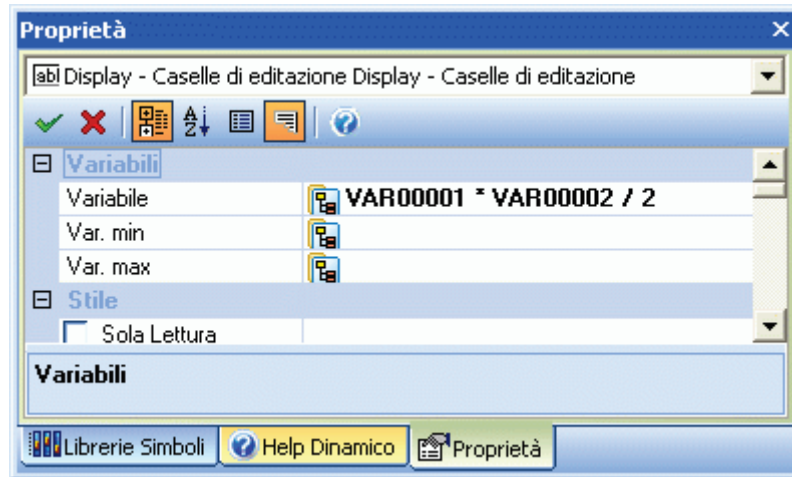
Esecuzione da un altro basic

L'esecuzione di un basic script può essere comandata da un altro Basic Script con la funzione "RunScript".

Se entrambi i basic script, quello contenente la funzione e quello lanciato, non sono in un thread separato, l'esecuzione del basic script sarà bufferizzata ed eseguita appena ve ne sarà l'opportunità.

1.5.2. Espressioni Basic Script nelle Proprietà degli Oggetti

L'impostazione delle caratteristiche di animazione disponibili negli oggetti di Movicon (Display, Rettangoli, ecc.) permette di stabilire la variabile e le relative soglie per l'attivazione delle funzioni grafiche. Movicon permette di sfruttare una caratteristica particolare per attivare le funzionalità grafiche: l'associazione di una stringa di espressione Basic Script per l'esecuzione del comando. Seguendo questa filosofia, al posto della variabile sarà possibile associare un'intera espressione VBA, completa di variabili e condizioni logiche, per l'esecuzione della funzione grafica desiderata. In tal modo, il valore restituito dall'espressione sarà gestito come valore interpretato al posto della variabile.



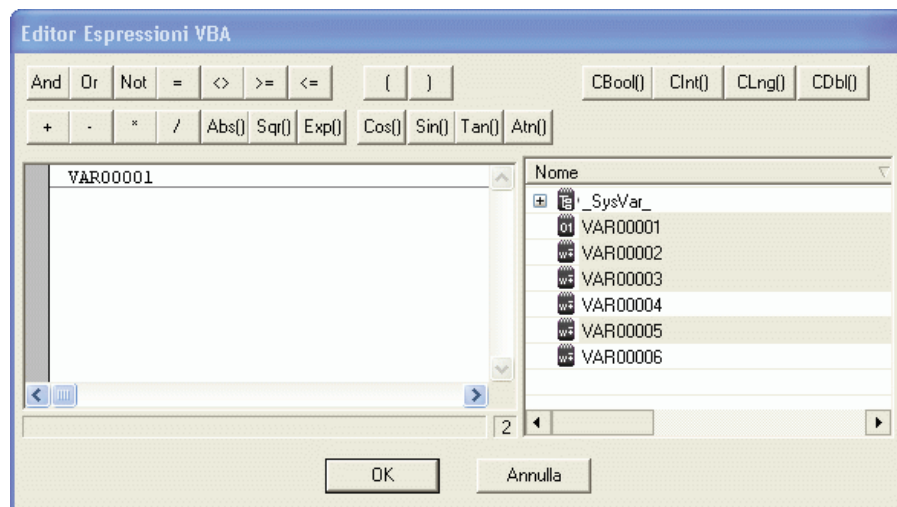
L'illustrazione mostra come è possibile inserire una espressione Basic Script al posto della variabile. Il valore restituito dall'espressione sarà interpretato al posto della variabile.



Attenzione! Non è possibile utilizzare funzioni script di Movicon come ad esempio `GetVariableValue()`, `SetVariableValue`, `QualityOf()`, ecc. nelle espressioni basic. Le espressioni basic supportano infatti soltanto gli operatori matematici del WinWrap riportati di seguito.

Editor delle Espressioni Basic

Le espressioni basic possono essere inserite direttamente al posto della variabile, ma è anche disponibile una finestra di editor che può aiutare il programmatore ad inserire queste espressioni. La finestra di editor delle espressioni basic si può aprire tramite l'apposito pulsante "Espressione..." presente nella finestra "Esploratore Tag":



Questa finestra di editor risulta essere molto comoda perchè mette già a disposizioni le operazioni ammesse per le espressioni e quindi consente un maggior controllo sugli errori.

Una volta aperta la finestra per inserire una variabile si esegue il doppio click sul nome della variabile nella lista di destra. Per inserire una funzione matematica si usa il pulsante corrispondente. Le funzioni disponibili sono:

Operatore	Descrizione
And	operatore binario AND
Or	operatore binario OR
Not	operatore logico NOT (inverte il valore booleano della variabile o dell'espressione che lo segue)
=	operatore binario uguaglianza
<>	operatore binario disuguaglianza
>=	operatore binario maggiore o uguale
<=	operatore binario minore o uguale
(parentesi aperta
)	parentesi chiusa
+	addizione
-	sottrazione
*	moltiplicazione
/	divisione
Abs()	valore assoluto (modulo)
Sqr()	radice quadrata
Exp()	potenza a base e (2,718)
Cos()	coseno (argomento espresso in radianti)
Sin()	seno (argomento espresso in radianti)
Tan()	tangente (argomento espresso in radianti)
Atn()	arcotangente
CBool()	conversione a booleano (boolean)
CInt()	conversione a intero (integer)
CLng()	conversione a intero lungo (long)
CDbl()	conversione a numero decimale a doppia precisione (double)

E' possibile inserire espressioni anche lunghe andando a capo con il tasto Enter. Se l'espressione inserita non è valida, in fase di conferma premendo il tasto OK un messaggio avvertirà dell'errore.



L'espressione potrebbe non essere validata perchè produce una divisione per zero. Nel validare un'espressione le variabili vengono tutte considerate al valore "1". Quindi ad esempio l'espressione "VAR00001/(VAR00002-1)" non verrà validata dall'editor delle espressioni perchè appunto produce una divisione per zero.

L'uso delle espressioni VBA nei controlli è molto veloce anche su WinCE. Inoltre richiede molte meno risorse rispetto a scrivere del codice VBA nel controllo. Infatti un solo thread si occupa di gestire tutte l'espressioni vba dei controlli caricati in memoria e queste espressioni sono valutate soltanto quando una variabile, contenuta all'interno, cambia di valore. Può risultare anche più veloce dell'uso della logica IL, perchè le espressioni vengono valutate solo quando le variabili usate all'interno cambiano di valore; mentre la logica IL viene eseguita in modo ciclico.



Questa funzionalità permette di estendere ulteriormente le potenzialità grafiche del disegno. La variabile associata a ciascuna delle proprietà grafiche del simbolo o disegno potrà infatti essere sostituita in realtà con espressioni Basic che potranno contenere combinazioni di più variabili o espressioni logiche o matematiche.



Quando un simbolo viene aggiunto in un sinottico, se questo simbolo contiene un'espressione VBA, ad esempio "VAR00001 + VAR00002", Movicon riconosce l'uso delle due variabili "VAR00001" e "VAR00002" e le inserisce entrambe nel progetto.



Attenzione! Questa funzionalità passa l'espressione inserita direttamente all'interprete Basic Script, quindi non sarà ammessa in questo caso l'uso di variabili che iniziano con caratteri numerici o caratteri particolari compresi gli underscore ("_").



Attenzione! Le espressioni basic non supportano la formattazione di tipo decimale (per esempio "x.x") ma solo "x". Pertanto è opportuno utilizzare una divisione per ordini decimali direttamente nella espressione (es: (VAR00001+VAR00002)/10).



Attenzione! Le Variabili Locali al Sinottico non sono supportate all'interno di Espressioni Basic.

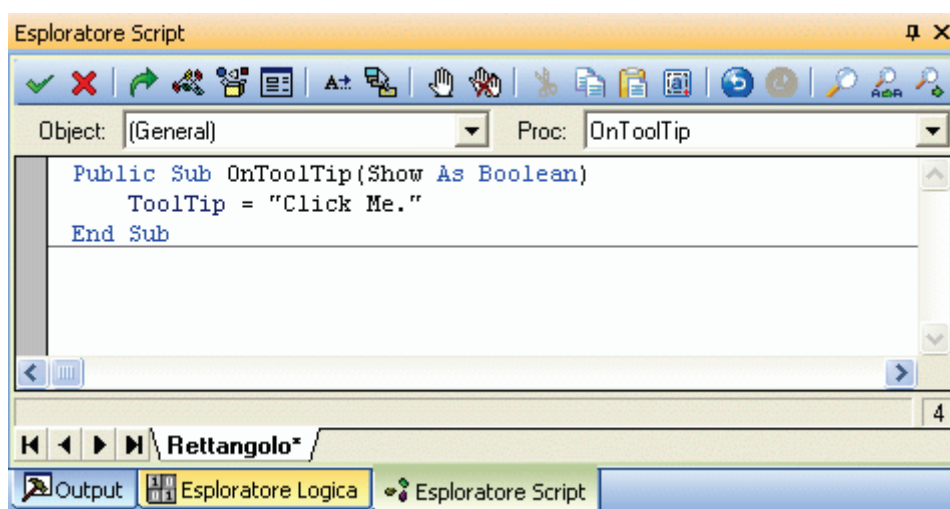


Attenzione! Nelle Espressioni Basic non è possibile utilizzare la sintassi <NomeVariabile>.<NumeroBit> ma è necessario eseguire una mascheratura per estrarre il valore del Bit voluto.

1.5.3. Basic Script VBA™ nel Codice degli Oggetti e Allarmi

Le soglie di allarme i disegni o simboli all'interno dei sinottici e i sinottici possono contenere del codice script. La gestione di questo codice basic inserito negli oggetti è differente da quella delle risorse Basic Script, la cui esecuzione è comandata nelle logiche del progetto. All'interno dell'oggetto sono resi disponibili una serie di eventi che se selezionati e quindi inseriti all'interno dell'editor basic script consentono di inserirvi del codice. Con il progetto in esecuzione in concomitanza della generazione di questi eventi nel progetto verranno eseguite le funzioni basic programmate al loro interno.

In base all'oggetto in cui si edita del codice è possibile avere a disposizione degli eventi oltre che delle proprietà e dei metodi diversi. Per una spiegazione dettagliata di ciascuno di questi si rimanda all'elenco presente nei paragrafi specifici.



Importante è ricordare che il codice associato ad un evento di un oggetto è eseguito solamente se l'oggetto è caricato in memoria e quindi gestito da Movicon. Per esempio un disegno a cui viene associato del codice è eseguito solamente quando il sinottico contenitore è caricato.

Va considerato che al caricamento della pagina sinottico il codice all'interno dei disegni non viene subito inizializzato ma soltanto quando è necessario. Se un simbolo contiene l'evento "SymbolLoading" allora Movicon è costretto ad inizializzare da subito il codice basic script contenuto in quel disegno. Questo significa che il caricamento della pagina risulterà più veloce se i disegni ad essa associata non contengono l'evento "SymbolLoading". Naturalmente questo non significa che l'evento "SymbolLoading" non debba essere utilizzato, ma conviene gestirlo solo se necessario. In particolar modo è consigliato considerare questo aspetto quando si realizzano i Template, soprattutto se poi dovranno essere utilizzati su Windows CE.



Il codice script dei disegni viene caricato solo quando si rende necessario il suo utilizzo e non quindi al caricamento della pagina.

Codice Script nelle Soglie Allarme

E' anche possibile associare del codice script alle singole soglie di un allarme. Per fare questo è sufficiente selezionare la soglia con il mouse e utilizzare la finestra "Esploratore Script". Anche nel codice delle soglie di allarme sono disponibili una serie di eventi che se selezionati e quindi inseriti

all'interno dell'editor basic script consentono di inserirvi del codice. Tali eventi sono ad esempio "AlarmLoading", "OnSetAlarm", ecc.. Per maggiori informazioni consultare l'apposita sezione dell'interfaccia script "AlarmThresholdCmdTarget". Si consiglia comunque di utilizzare lo script negli allarmi soltanto se necessario in quanto ci sono alcune limitazioni, come ad esempio il fatto che lo script negli allarmi non è gestito in ridondanza. Inoltre consultare le limitazioni di utilizzo riportate nel capitolo "Gestione Allarmi" delle "Specifiche Funzionali" di Movicon.

1.6. Basic Script Condivisi

L'utilizzo di codice basic script potrebbe richiedere la necessità di utilizzare singole funzioni o subroutine o procedure in più parti del progetto. Per evitare di scrivere più volte lo stesso codice script è quindi possibile rendere "condiviso" un Basic Script, in modo che le funzioni contenute al suo interno possano essere richiamate anche da altri codici script. Questo consente di effettuare eventuali modifiche una volta soltanto all'interno dell'editor di tale risorsa, velocizzando notevolmente le fasi di editing.

Per poter accedere da un Basic Script alle funzioni di un altro Basic Script è necessario utilizzare la funzione "Uses". Questa funzione di fatto consente di implementare all'interno del Basic Script le funzioni contenute nel Basic Script specificato. La sintassi completa è la seguente:

'#Uses "*"NomeBasicScript"

Il carattere "*" prima del nome della routine basic consente a Movicon di interpretare il codice XML contenuto nel file del Basic Script.



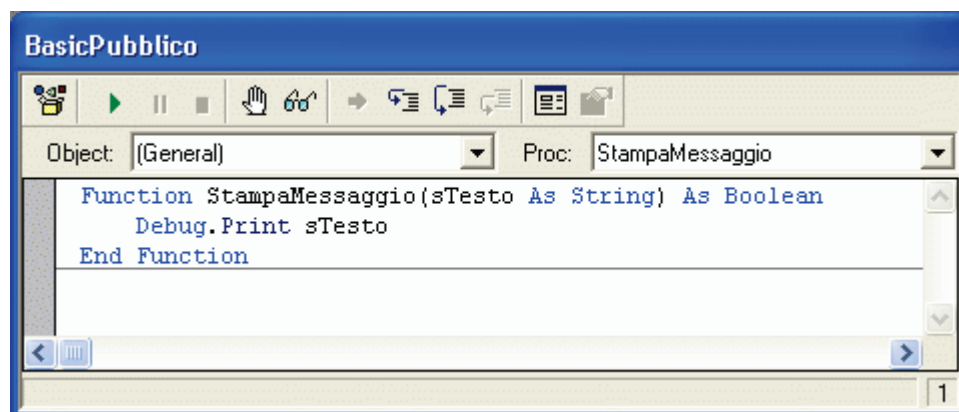
ATTENZIONE! L'istruzione '#Uses nel codice di un disegno può essere usata solo per functions, subroutines, public var o public constant. Non può contenere eventi del disegno perchè l'attuale gestione del "delay load" del codice non permette di sapere a priori che in un basic script condiviso ci sono degli eventi da processare per il disegno.

Il comando **"Apri #Uses"** presente nella finestra "Esploratore Script" o nel "Menù Modifica (Basic Script)" consente di avere le funzioni e gli oggetti dichiarati nei moduli "#Uses" di uno script disponibili all'intellisense (con comando: CTRL + Space). In questo modo tramite il comando CTRL + Spazio verrà proposto l'elenco di tutte le funzioni disponibili che sarà composto sia da quelle standard dello script sia da quelle definite nel modulo #Uses.

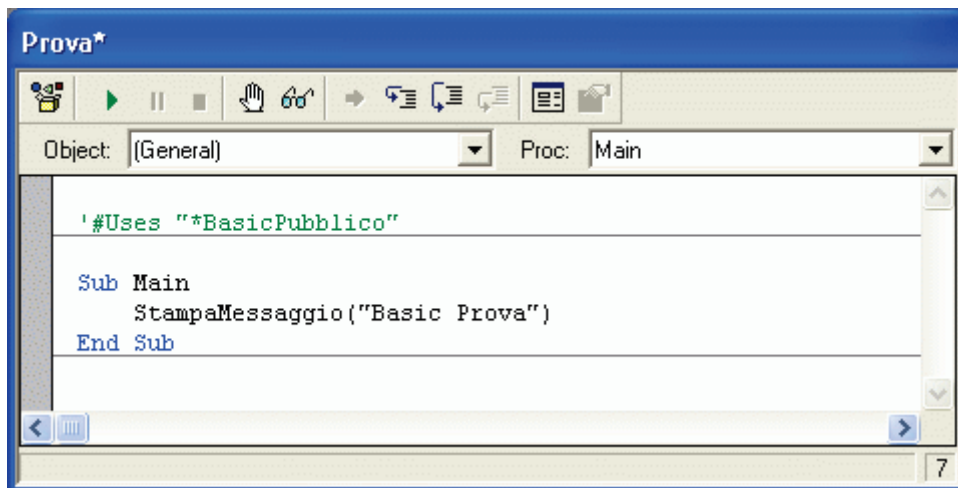
Un modulo richiamato tramite l'istruzione "#Uses" può a sua volta richiamarne un'altro. In questo caso riapplicando il comando "Apri #Uses" sono rese disponibili all'intellisense anche le funzioni contenute in questo ulteriore modulo.

Esempio:

Si supponga di avere creato un basic script denominato "BasicPubblico", all'interno del quale è stata inserita la funzione "StampaMessaggio()" che segue la stampa del testo che viene passato come parametro:



A questo punto se si vuole richiamare la funzione "StampaMessaggio()" da un'altra routine basic, o anche dal codice script di un disegno o di un sinottico, ecc. sarà sufficiente utilizzare la sintassi sopra descritta all'interno dello script. Supponendo quindi di voler implementare la funzione nello script "Prova" il codice potrebbe risultare il seguente:



ATTENZIONE! Se un Basic Script utilizza una funzione di un altro Basic Script, questo di fatto implementa al suo interno tutto il codice del Basic Script condiviso, e non soltanto la funzione utilizzata. Quindi non potranno essere dichiarate funzioni, routine o costanti con lo stesso nome in entrambi i Basic Script.



E' possibile creare più routine Basic Script da condividere in uno stesso progetto. Questo permette di poter raggruppare e gestire in modo più strutturato le eventuali funzioni pubbliche di cui ci si dovrà servire.

Condivisione Annidata di Basic Script

E' possibile anche gestire ad albero i Basic Script condivisi, ovvero un Basic Script ne può condividere un altro che a sua volta ne condivide un terzo. In questa configurazione però ogni Basic Script potrà utilizzare solo le routine o costanti o variabili del Basic Script che condivide.

Ad esempio, si supponga di avere creato il BasicScript1 che condivide il BasicScript2 e che il BasicScript2 condivide il BasicScript3. In questa situazione il BasicScript1 potrà richiamare le routine del BasicScript2, ma non quelle del BasicScript3, che invece potranno essere utilizzate dal BasicScript2.

Condivisione di Basic Script tra progetti Padre e Figlio

E' possibile condividere le risorse Basic Script anche tra progetti Padre e Figlio. In questo caso la sintassi da utilizzare sarà la seguente:

Condivisione di una risorsa Basic Script in un progetto figlio dal progetto padre:

```
'#Uses "*ChildProjectName\BasicScriptName"
```

Condivisione di una risorsa Basic Script in un progetto padre da un progetto figlio:

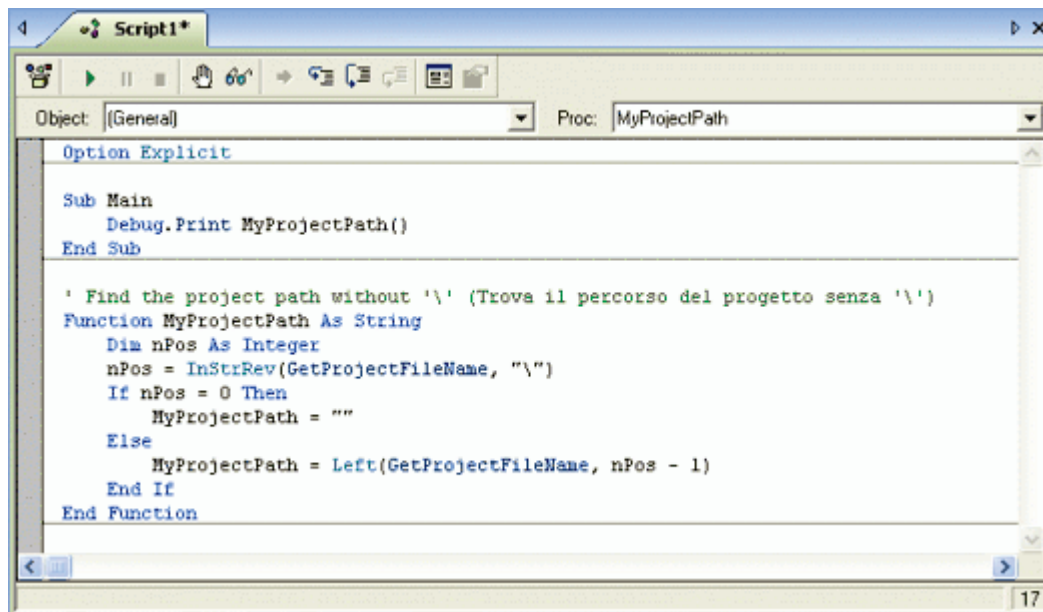
```
'#Uses "..\BasicScriptName"
```

1.6.1. Editor Basic Script

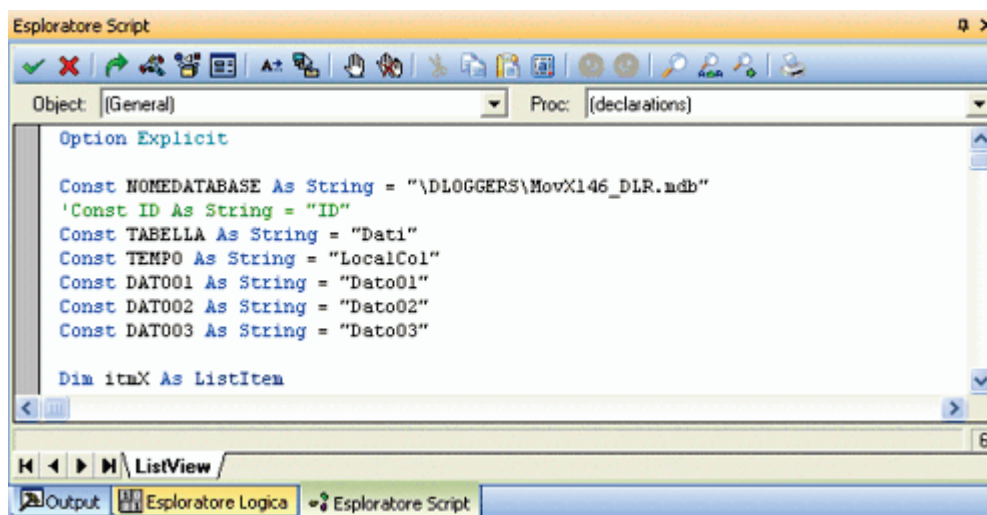
1.7. Editor Basic Script

L'editor basic script si compone di una finestra comprendente un'area con sfondo bianco in cui è possibile editare il codice, più una serie di comandi descritti di seguito.

La dimensione della finestra può essere ridimensionata e la possibilità di aprire contemporaneamente più finestre permette di eseguire rapidamente operazioni di Copia-Incolla anche da progetti diversi. Per aprire l'editor è sufficiente fare un doppio click con il mouse sulla risorsa Basic Script.



L'editazione del codice script degli oggetti, dei sinottici e degli allarmi invece viene fatta tramite la finestra **"Esploratore Script"** di Movicon. Il contenuto di questa finestra infatti cambia dinamicamente in base al componente selezionato, mostrando il codice script ad esso associato.



Entrambe le finestre di editing sopra mostrate mettono a disposizione due liste a discesa, **"Object:"** e **"Proc:"**.

Nella lista **"Object:"** è visualizzato l'elenco degli oggetti basic definiti in Movicon che corrispondono a **"(General)"** e **"On_Events"**. La selezione di un oggetto rispetto ad un altro modifica l'elenco delle procedure nella lista **"Proc:"** disponibili per quello specifico oggetto.

L'oggetto **"(General)"** identifica le procedure ed eventi programmate per quel specifico oggetto e l'elenco può cambiare a seconda di dove stiamo editando il codice: Risorsa Basic Script, Disegno, Sinottico o Allarme.

L'oggetto **"On_Events"** identifica gli eventi di Movicon generali ed è disponibile solo quando si edita il codice associato al progetto.

E' possibile poi trovare nella lista **"Object:"** altre voci che corrispondono ad eventuali componenti ActiveX/OCX inseriti nel codice basic e che mettono a disposizione dei propri eventi.

Nella lista **"Proc:"** è visualizzato l'elenco degli eventi disponibili in base alla selezione fatta nella lista **"Object:"**.

1.7.1. Menù Modifica (Basic Script)

All'apertura di una finestra di editazione del codice di una risorsa Basic Script, nel menù vengono visualizzati una serie di comandi specifici per l'editazione del codice, oltre ai classici comandi di Windows:

Annulla (Ctrl+Z; Alt+Backspace)

Esegue l'annullamento dell'ultima operazione eseguita (es. cancellazione di una risorsa, modifica di un oggetto, ecc.).

Questo comando è disponibile anche nella **"Barra Strumenti"** di Movicon.

Ripristina

Esegue il ripristino dell'ultima operazione Annullata (es. cancellazione di una risorsa, modifica di un oggetto, ecc.).

Questo comando è disponibile anche nella **"Barra Strumenti"** di Movicon.

Taglia (Ctrl+X; Maiuscolo+Cancella)

Taglia l'oggetto, la risorsa o qualsiasi cosa sia stata selezionata. In questo caso l'oggetto viene cancellato ma mantenuto in memoria negli appunti di Windows.

Questo comando è disponibile anche nella **"Barra Strumenti"** di Movicon.

Copia (Ctrl+C; Ctrl+Ins)

Copia l'oggetto, la risorsa o qualsiasi cosa sia stata selezionata in memoria negli appunti di Windows.

Questo comando è disponibile anche nella **"Barra Strumenti"** di Movicon.

Incolla (Ctrl+V; Maiuscolo+Ins)

Incolla l'oggetto, la risorsa o qualsiasi cosa sia stata precedentemente Copiata o Tagliata dagli appunti di Windows nella zona dell'area di lavoro selezionata dal mouse.

Questo comando è disponibile anche nella **"Barra Strumenti"** di Movicon.

Elimina (Cancella)

Elimina l'oggetto selezionato senza copiarlo negli appunti di Windows.

Trova (ALT+F3)

Qualora disponibile in seguito all'apertura di una risorsa predisposta, il comando Trova permette di specificare un carattere o una stringa di testo da ricercare all'interno della risorsa selezionata.

Secondo gli standard, sarà possibile specificare se il testo da ricercare dovrà rispettare i caratteri Maiuscoli o Minuscoli.

Il pulsante Successivo avvia una nuova ricerca, mentre il pulsante Annulla chiude la ricerca.

Al comando Trova si accede anche da tastiera con ALT+F3, oppure, se disponibile, dalla **"Barra Strumenti"** tramite la casella di editazione Trova.

Questo comando è disponibile anche nella **"Barra Strumenti"** di Movicon.

Trova Prossima (F3)

Dopo avere eventualmente avviato una ricerca con il comando Trova, è possibile ricercare il testo specificato successivo con il comando Trova Prossima, eseguibile anche con il tasto F3

Questo comando è disponibile anche nella **"Barra Strumenti"** di Movicon.

Sostituisci

Il comando Sostituisci consente di specificare un testo da ricercare e un testo da sostituire a quello trovato.

Seleziona Tutto

Il comando Seleziona Tutto permette di eseguire simultaneamente la selezione di tutto il contenuto della risorsa attualmente attiva, qualora questo lo consenta.

Carattere

Imposta il tipo di carattere con cui visualizzare il codice Basic Script all'interno della finestra di editazione. Questa voce è disponibile soltanto dopo avere aperto l'editor Basic Script.

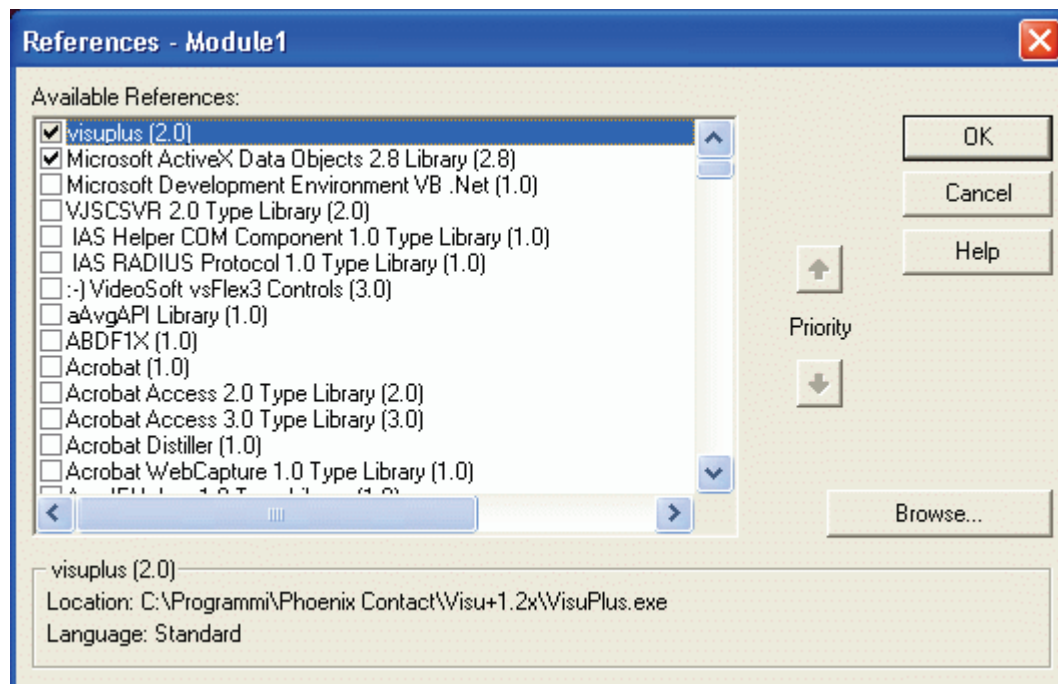
Colori nella Sintassi

Permette di modificare i colori standard della sintassi Basic Script che corrispondono a:

- **Verde azzurro:** Funzioni del linguaggio motore VBA integrato.
- **Verde:** Righe di commento.
- **Rosso:** Istruzioni in errore.
- **Blu Scuro:** Funzioni aggiunte in Movicon o estese tramite librerie esterne ActiveX/OCX.
- **Blu:** Riservato sempre alle funzioni del motore VBA integrato.

Riferimenti...

Permette di accedere all'elenco dei references (ActiveX/OCX) installati sul PC nel quale è caricato il sistema Movicon. Inoltre dalla finestra che appare è possibile attivarne uno o più di uno contemporaneamente e disporre quindi delle funzioni che questo implementa.



Completa la Parola (CTRL+Barra Spaziatrice).

Visualizza una lista dei comandi contenente il testo parziale in editazione. Il comando rapido per accedere a questa funzionalità è il Ctrl+Spazio. La pressione del tasto Spazio riporta sull'area di editazione il comando selezionato dalla lista.

Informazione sui Parametri (CTRL+SHIFT+Barra Spaziatrice)

Posizionando il cursore su una funzione e attivando il comando viene visualizzato un tooltip che riporta il tipo di parametri necessari per la funzione. Questa funzione è sempre attiva anche in fase di editazione.

Apri #Uses

Questo comando consente di avere le funzioni e gli oggetti dichiarati nei moduli "#Uses" di uno script disponibili all'intellisense (con comando: CTRL + Space).

1.7.2. Menù Debug

I comandi del "Menù Debug" riguardano esclusivamente l'esecuzione delle routine Basic Script, e sono comandi che normalmente si trovano anche nella "**Barra Utensili Basic Script**".

Esegui Istruzione

Questo comando esegue passo-passo una funzione alla volta ad ogni pressione del comando. Questo comando è disponibile anche nella "Barra Utensili Basic Script" di Movicon.

Esegui Istruzione/Routine

Questo comando esegue passo-passo una funzione alla volta ad ogni pressione del comando. Questo comando è disponibile anche nella "Barra Utensili Basic Script" di Movicon.

Esci da Istruzione/Routine

Questo comando esegue le funzioni presenti fino ad arrivare nella linea in cui è presente il cursore. Questo comando è disponibile anche nella "Barra Utensili Basic Script" di Movicon.

Esegui fino al cursore

Questo comando esegue le funzioni presenti fino ad arrivare nella linea in cui è presente il cursore.

Imposta/Rimuovi punto di interruzione

Questo comando inserisce o elimina un break point sulla riga in cui è presente il cursore. Questa funzione è disponibile anche con un click sull'estremo bordo sinistro della finestra di editor in coincidenza della riga nella quale si vuole inserire il break point. Nel momento in cui il basic deve eseguire una riga di codice in cui è presente un break point l'esecuzione viene messa automaticamente in pausa.

Questo comando è disponibile anche nella "Barra Utensili Basic Script" di Movicon.

Rimuovi punti di interruzione

Questo comando elimina tutti i Break Point inseriti nel codice.

Controllo Immediato

Questo comando restituisce su una finestra di dialogo o nella finestra di debug (finestra di Watch) del basic il risultato della funzione che è stata evidenziata.

Questo comando è disponibile anche nella "Barra Utensili Basic Script" di Movicon.

Aggiungi Espressione di Controllo

Questo comando restituisce su una finestra di dialogo o nella finestra di debug (finestra di Immediate) del basic il risultato della funzione che è stata evidenziata.

Visualizza Oggetti

Questo comando apre la finestra di browse delle funzioni basic script di Movicon.

Imposta Istruzione Successiva

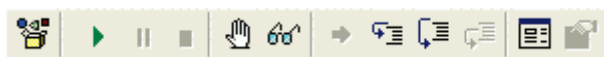
Questo comando consente di impostare quale sarà la prossima istruzione che verrà eseguita durante la fase di debug della routine. Attivando il comando verrà impostata come istruzione da eseguire la riga sulla quale si trova il cursore del mouse in quel momento.

Mostra Istruzione Successiva

Questo comando consente di posizionare il cursore all'altezza della prossima istruzione che dovrà essere eseguita durante la fase di debug della routine.

1.7.3. Barra Utensili Basic Script

Aprendo la finestra di editazione del codice basic script sulla parte superiore viene mostrata la barra utensile Basic Script.



Alcuni comandi sono eseguibili anche dal "Menù Debug".

La funzione dei tasti è di seguito elencata:

Browse



Questo comando consente di accedere all'elenco delle funzioni e proprietà disponibili nel basic script. La finestra che si apre è chiamata Browse delle funzioni.

Start/Resume



Questo comando avvia l'esecuzione del basic script. Se il codice è riferito ad una risorsa basic del progetto viene eseguito il contenuto della procedura Sub Main altrimenti se è riferito ad un oggetto abilita la gestione degli eventi su di esso.

Pause



Questo comando mette in pausa il basic script che è in esecuzione posizionandosi e colorando di giallo la linea di codice che era in esecuzione.

End



Questo comando arresta l'esecuzione del basic script. Il codice deve essere in pausa per agire su questo comando.

Break Point



Questo comando inserisce o elimina un break point sulla riga in cui è presente il cursore. Questa funzione è disponibile anche con un click sull'estremo bordo sinistro della finestra di editor in coincidenza della riga nella quale si vuole inserire il break point. Nel momento in cui il basic deve eseguire una riga di codice in cui è presente un break point l'esecuzione viene messa automaticamente in pausa.

Quick Watch



Questo comando restituisce su una finestra di dialogo o nella finestra di debug (finestra di Watch) del basic il risultato della funzione che è stata evidenziata.

Show Current Statement



Questo comando consente di posizionare il cursore all'altezza della prossima istruzione che dovrà essere eseguita durante la fase di debug della routine.

Step Into



Questo comando esegue passo-passo una funzione alla volta ad ogni pressione del comando.

Step Over



Questo comando esegue passo-passo una funzione alla volta ad ogni pressione del comando.

Step Out



Questo comando esegue le funzioni presenti fino ad arrivare nella linea in cui è presente il cursore.

Edit UserDialog



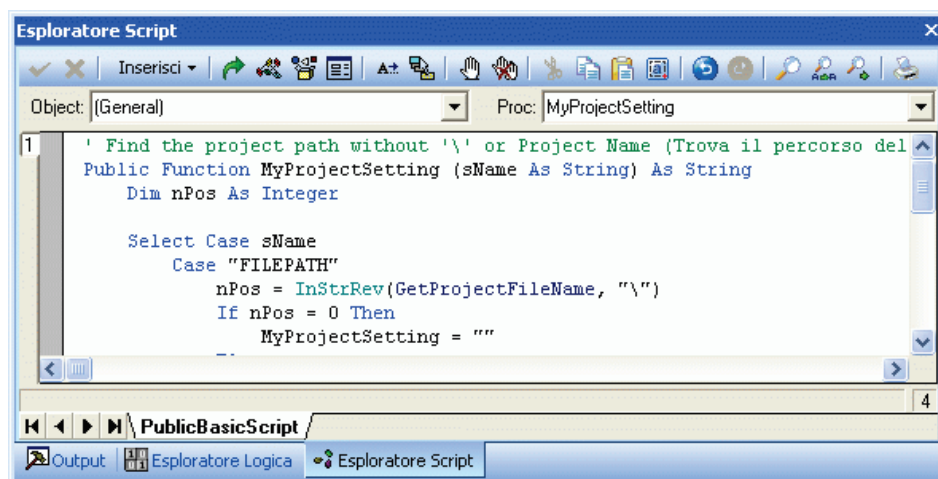
Questo comando apre il tool per la creazione grafica delle finestre di dialogo basic script. All'uscita con conferma viene tradotto ciò che è stato creato graficamente in codice. Posizionandosi su questo e attivando il comando, viene nuovamente riprodotto graficamente ciò che è espresso in codice basic script.

Edit Module Properties



1.8. Finestra Esploratore Script

La finestra "Esploratore Script" è di fondamentale importanza per l'editazione del codice script degli oggetti del progetto. Questa finestra infatti, che risulta essere un editor di testo, consente di inserire il codice script per i disegni, sinottici, allarmi e anche delle risorse Basic Script. Il contenuto della finestra cambia in modo dinamico a seconda dell'oggetto o della risorsa del progetto selezionati in quel momento. E' infatti possibile associare il codice script sia al progetto, sia ad ogni singolo simbolo o disegno di Movicon. Selezionando quindi un simbolo, nella finestra "Esploratore Script" apparirà il codice relativo a quel simbolo, sempre che sia stato inserito.



E' inoltre disponibile il tasto funzione F8 per richiamare la finestra "Esploratore Tag" utile quando si deve inserire una variabile.

Tramite la barra comandi posta in alto alla finestra "Esploratore Script" è possibile eseguire comandi di editing e di debug del codice:



Il significato delle icone presenti nella barra comandi è il seguente:



OK. La pressione di questo pulsante di validazione rende attive le eventuali modifiche eseguite nel codice.



Annulla. La pressione di questo pulsante elimina le eventuali modifiche eseguite nel codice. In questo caso vengono annullate soltanto le modifiche apportate dopo l'ultimo comando di validazione eseguito.



Questo comando consente di inserire nel Basic Script una variabile del RealTimeDB del progetto. La selezione avverrà tramite l'apertura della finestra di Browse delle Variabili.



Run/Sintassi. La pressione di questo pulsante esegue la verifica del codice inserito, segnalando eventuali errori (ad esempio di sintassi).



Riferimenti. La pressione di questo pulsante esegue l'apertura della finestra dei "Reference" per la selezione di componenti ActiveX/OCX da inserire.



Visualizza. Questo comando consente di accedere all'elenco delle funzioni e proprietà disponibili nel basic script. La finestra che si apre è chiamata Browse delle funzioni.



Finestra di Dialogo Script. Questo comando apre il tool per la creazione grafica delle finestre di dialogo basic script. All'uscita con conferma viene tradotto ciò che è stato creato graficamente in codice. Posizionandosi su questo e attivando il comando, viene nuovamente riprodotto graficamente ciò che è espresso in codice basic script.



Completa la Parola. Questo comando visualizza una lista dei comandi contenente il testo parziale in editazione. Il comando rapido per accedere a questa funzionalità è il Ctrl+Spazio. La pressione del tasto Spazio riporta sull'area di editazione il comando selezionato dalla lista.



Informazione sui Parametri. Posizionando il cursore su una funzione e attivando il comando viene visualizzato un tooltip che riporta il tipo di parametri necessari per la funzione. Questa funzione è sempre attiva anche in fase di editazione.



Punto di Interruzione. Questo comando inserisce o elimina un break point sulla riga in cui è presente il cursore. Questa funzione è disponibile anche con un click sull'estremo bordo sinistro della finestra di editor in coincidenza della riga nella quale si vuole inserire il break point. Nel momento in cui il basic deve eseguire una riga di codice in cui è presente un break point l'esecuzione viene messa automaticamente in pausa.



Rimuovi Punti di Interruzione. Questo comando elimina tutti i Break Point inseriti nel codice.



Apri #Uses. Questo comando consente di avere le funzioni e gli oggetti dichiarati nei moduli "#Uses" di uno script disponibili all'intellisense (con comando: CTRL + Space).



Taglia. Taglia il testo selezionato. In questo caso il testo viene cancellato ma mantenuto in memoria negli appunti di Windows.



Copia. Copia il testo selezionato in memoria negli appunti di Windows.



Inserisci. Incolla il testo precedentemente Copiato o Tagliato dagli appunti di Windows nella zona dell'area di lavoro selezionata dal mouse.



Seleziona tutto. Esegue la selezione di tutto il testo contenuto nella finestra "Esploratore Script".



Annulla. Esegue l'annullamento dell'ultima operazione eseguita (es. cancellazione di un testo, ecc.).



Ripristina. Esegue il ripristino dell'ultima operazione Annullata (es. cancellazione di un testo, ecc.).



Trova. Il comando Trova permette di specificare un carattere o una stringa di testo da ricercare all'interno della finestra "Esploratore Script". Secondo gli standard, sarà possibile specificare se il testo da ricercare dovrà rispettare i caratteri Maiuscoli o Minuscoli.



Sostituisci. Dopo avere eventualmente avviato una ricerca con il comando Trova, è possibile ricercare il testo specificato successivo con il comando Trova Prossima.



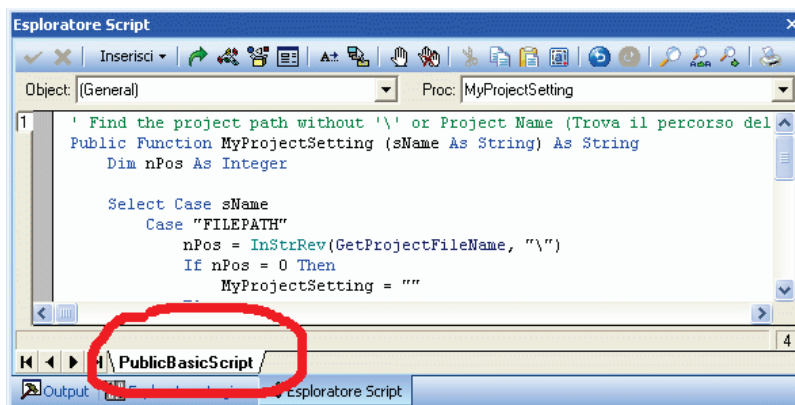
Trova Prossima. Il comando Sostituisci consente di specificare un testo da ricercare e un testo da sostituire a quello trovato.



Stampa. Tramite questo pulsante è possibile mandare in stampa il codice visualizzato. Si aprirà comunque la finestra di selezione ed impostazione Stampanti.

Oggetto in Editazione

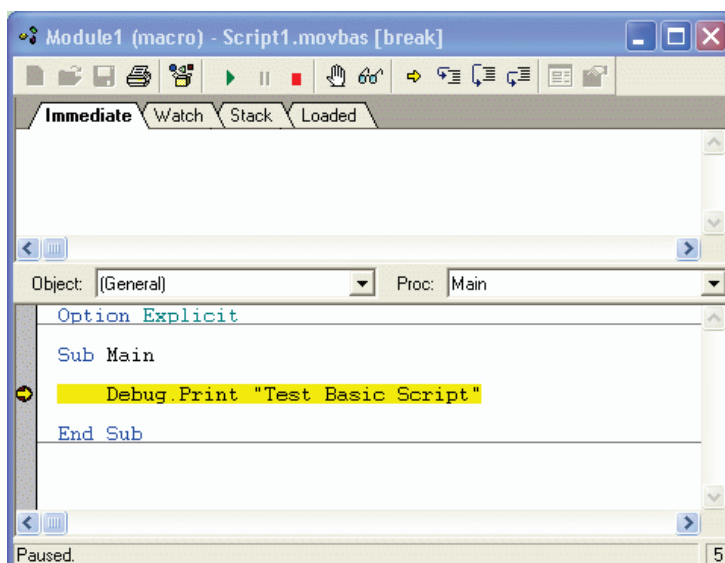
Quando si seleziona uno script, il suo nome o il nome del componente associato viene riportato nella barra in basso alla finestra. Questo consente di poter capire qual'è il componente a cui fa riferimento lo script visualizzato nella finestra:



1.9. Debug dei Basic Script

Movicon consente il completo debug delle logiche basic script del progetto, siano esse risorse del progetto o contenute nei simboli grafici dei sinottici o degli allarmi.

Nella fase di progettazione è possibile debuggare solo i Basic Script definite come Risorse, mentre non è possibile testare gli script associati agli oggetti o agli allarmi in quanto questi sono gestiti ad evento. Durante la fase di Run-Time invece è possibile debuggare tutti gli script. In fase di programmazione è necessario inserire dei Break-Point all'interno del codice. Quando in Run-Time lo script verrà eseguito si aprirà la finestra di debug e l'esecuzione del codice sarà ferma alla posizione del Break-Point. A questo punto tramite la barra di debug posta in alto alla finestra sarà possibile eseguire lo script step a step, ecc.





L'inserimento di Break-Point all'interno degli script è permanente, ovvero vengono salvati nelle proprietà dello script, e quindi saranno attivi anche in modalità Run-Time. Si raccomanda quindi di inserire i Break-Point solo durante la fase di debug e di rimuoverli subito dopo avere eseguito i test.



Alcune funzioni basic script verificate in programmazione possono ritornare con valori diversi se eseguite in runtime, quindi si raccomanda di utilizzare il debug in programmazione solamente per verificare in maniera grossolana il codice e ripetere il test in maniera accurata lanciando in esecuzione il progetto.

1.9.1. Proprietà Basic Script

1.10. Proprietà Basic Script

Ogni routine **"Basic Script"** inserita come risorsa nella finestra "Esploratore Progetto" può ricevere in associazione delle Proprietà, che ne determinano il tipo di funzionamento durante l'esecuzione. Per fare questo è sufficiente selezionare il "Basic Script" desiderato e quindi modificare le impostazioni tramite la **"Finestra delle Proprietà"** di Movicon.

1.10.1. Proprietà Generali di un Basic Script

Tramite le proprietà Generali è possibile definire il nome della Risorsa Basic Script selezionata nella finestra "Esploratore Progetto". Per fare questo è sufficiente selezionare il Basic Script desiderato e quindi modificare le impostazioni tramite la **"Finestra delle Proprietà"** di Movicon.

Nome

Tramite questa proprietà è possibile leggere o modificare il nome del Basic Script.

1.10.2. Proprietà Modo di un Basic Script

Tramite le proprietà "Modo" è possibile definire la modalità con cui verrà eseguito lo script. Per modificare le proprietà Modo di uno Script, selezionare l'oggetto con il mouse e utilizzare la **"Finestra delle Proprietà"** di Movicon.

Esegui sul Server

Questa proprietà serve quando si utilizza la Ridondanza tra due progetti. Se infatti viene abilitata, lo script verrà eseguito soltanto sul progetto che ha il controllo in quel momento. Se ad esempio entrambi i progetti, Primario e Secondario, sono in esecuzione il controllo sarà del progetto Primario. In questo caso anche se l'esecuzione del Basic Script viene comandata dal Secondario questo verrà eseguito sul Primario.

Questa gestione avviene anche se lo script viene eseguito da un "Progetto Figlio" impostato per collegarsi ad un "Server di Rete". Anche in questa situazione lo script verrà eseguito sul progetto "Server di Rete" anziché sul Progetto Figlio.

Utilizza l'Interfaccia Utente

Questa impostazione consente di aggiungere la gestione dell'interfaccia utente al basic script. E' necessario abilitare questa proprietà quando si devono utilizzare controlli come "MsgBox" o "Dialog Box" all'interno del Basic Script.



Le funzioni del WinWrap elencate di seguito non sono supportate quando la proprietà "Utilizza l'Interfaccia Utente" dello script non è attivata e provocheranno la generazione di un errore in fase di caricamento del codice script: AppActivate, AboutWinWrapBasic, Shell, ShowPopupMenu.

Thread Separato

Questa selezione consente di eseguire la routine Basic in un Thread separato, quindi anche simultaneamente ad un'altra routine basic o processo sullo stesso thread (finestre di dialogo).

Questa possibilità va utilizzata quando il basic script viene lanciato mentre un altro è già in esecuzione. Se in entrambi non è marcata tale opzione allora l'esecuzione del secondo viene bufferizzata ed eseguita nel momento in cui termina il primo. Ricordiamo che non tutte le funzioni Movicon sono disponibili se eseguite in un Thread separato. Fare riferimento alla guida di ogni singola istruzione per verificare se può essere eseguita oppure no in un Thread separato.



Bisogna fare attenzione agli Script che non usano un Thread separato, ma eseguono operazioni lunghe. Questi Script infatti finiscono tutti in uno stesso Thread che condividono (consumo risorse limitato), ma però si mettono in coda di esecuzione uno rispetto all'altro. Per gli Script che eseguono operazioni lunghe, quindi, occorre marcare l'opzione "Thread Separato".

Crea un Proprio Trace Tab

Abilitando questa casella di selezione, durante la fase di runtime verrà creata una TAB nella finestra di Output personalizzata per la routine Basic Script, dove verranno stampati i messaggi inerenti allo script, come ad esempio i Debug.Print.

Dialog Modali

Abilitando questa casella di selezione sarà possibile rendere modali le finestre di dialogo aperte dallo script. In caso contrario le finestre di dialogo non saranno modali, e quindi cliccando fuori dall'area della finestra di dialogo questa andrà in secondo piano.

1.10.3. Proprietà Esecuzione di un Basic Script

Tramite le proprietà "Esecuzione" è possibile definire i parametri con cui verrà eseguito lo script. Per modificare le proprietà Modo di uno Script, selezionare l'oggetto con il mouse e utilizzare la "Finestra delle Proprietà" di Movicon.

Priorità

L'esecuzione di una risorsa basic script può essere eseguito con diverse priorità: alta, normale e bassa:

Alta: Massima priorità
Normale: Priorità normale
Bassa: Valore di default

Tag di Stato

E' possibile assegnare una delle variabili del Real Time DB di Movicon alla routine Basic Script. In tal caso, il sistema scriverà lo stato di esecuzione della routine come codice del valore di stato sulla variabile desiderata.

La variabile associata può essere dichiarata di qualsiasi tipo (byte, word, ecc.). Il contenuto sarà così indicato:

Bit 0 = Basic in run
Bit 1 = Basic in pausa
Bit 2 = Basic in errore
Bit 3 = Basic bufferizzato
Bit 4 = Esecuzione pendente
Bit 5 = Basic sospeso

Gli altri bit non sono gestiti, ma è opportuno non utilizzarli per eventuali gestioni future. Notare che se la variabile definita è di tipo Bit, verrà fornita solo l'informazione dello stato di Run.

Numero Massimo Istanze

Questa proprietà consente di definire il numero massimo di istanze concorrenti permesse per il Basic Script. Questo significa che la routine basic può essere richiamata anche più volte nello stesso momento e verranno quindi create n istanze. Il risultato finale sarà comunque differente a seconda di come si è impostata la proprietà "Thread Separato":

Proprietà Thread Separato abilitata: se lo script viene eseguito più volte contemporaneamente, verranno creati diversi thread concorrenti (fino a un massimo definito dalla proprietà "Numero Massimo Istanze") ed eseguiti contemporaneamente.

Proprietà Thread Separato NON abilitata: se lo script viene eseguito più volte contemporaneamente, le richieste verranno bufferizzate ed eseguite successivamente non appena lo script in esecuzione sarà terminato. Se lo script infatti non è impostato in Thread Separato potrà essere eseguita soltanto un'istanza per volta.

Se poi viene richiesta l'esecuzione di uno script e questo è già in esecuzione con il numero massimo di istanze definite, allora quest'ultima richiesta non verrà eseguita e ne verrà data notifica con un messaggio nel log storico:

"Non è possibile eseguire lo Script 'Script1'. Lo Script è già in esecuzione oppure è stato raggiunto il numero massimo di istanze allocabili"

Sleep (ms)

Questo tempo di sleep serve per rendere meno onerosa l'esecuzione del basic script per il processore. Più alto sarà il tempo di sleep più il basic script impegnerà meno il processore e di conseguenza rallenterà la sua esecuzione.

Timeout Sincronismo (ms)

Tempo di timeout per i basic script eseguiti come sincroni. Questo tempo è inteso come il tempo massimo di attesa prima dell'errore di timeout durante l'esecuzione dello script.

Nota: Se lo script risorsa è eseguito come basic script di start-up, esso viene eseguito sempre sincrono rispetto alle risorse del progetto avviate (anche se è attiva l'opzione Thread Separato), attendendo quindi, al massimo, un tempo uguale al valore impostato nel parametro.

1.10.4. Proprietà Script Debug Output di un Basic Script

Tramite le proprietà "Script Debug Output" è possibile scrivere su file di log o mandare in stampa i testi inseriti nell'istruzione "Debug.Print" del Basic Script.

Per modificare le proprietà Script Debug Output di uno Script, selezionare l'oggetto con il mouse e utilizzare la "**Finestra delle Proprietà**" di Movicon.

Barra di Stato

Abilitando questa proprietà i messaggi eseguiti dalla funzione Debug.Print verranno stampati anche nella barra di stato di Movicon, oltre che sull'eventuale TAB di debug dello script se questa è stata abilitata tramite la proprietà "Crea un proprio Trace Tab".

Log Storico

Abilitando questa proprietà i messaggi eseguiti dalla funzione Debug.Print verranno stampati anche nel Log Storico del progetto.

Spooler

Abilitando questa proprietà i messaggi eseguiti dalla funzione Debug.Print verranno inviati allo spooler di stampa del sistema.

1.10.5. ActiveX/OCX

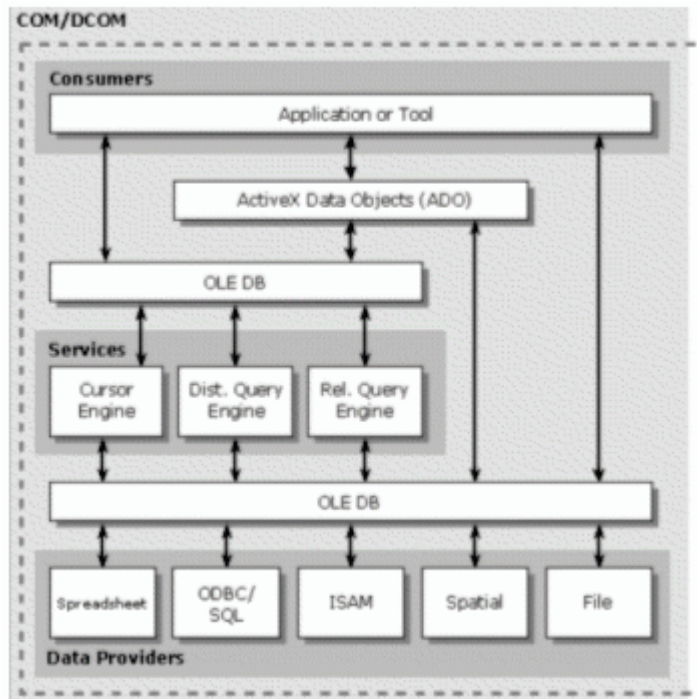
1.11. ADO nei Basic Script

Una risorsa Basic di Movicon può liberamente fare uso degli oggetti Microsoft ADO (ActiveX Data Object) per l'accesso diretto ai dati contenuti in un database.

L'abilitazione delle funzioni base ADO può avvenire attraverso i references selezionando la voce "Microsoft ActiveX Data Object Library...".

Le funzioni ADO permettono di interfacciarsi al sistema OLE DB il quale si occupa di reperire le informazioni dai database. I DataBase possono così essere gestiti in maniera trasparente, come qualsiasi oggetto, all'interno di un Basic Script.

L'architettura generale dell'integrazione degli ADO all'interno di un sistema è quella illustrata nella figura sottostante.



Si riporta di seguito un piccolo esempio di codice basic script che utilizza ADO per l'accesso a un DataBase:

```
Option Explicit
Const PATH_DB As String = "|DatiAdo|"
Const FILE_DB As String = "DatiAdo.mdb"

'Data for ADO architecture (dati per architettura ADO)
Dim Conn1 As New ADODB.Connection
Dim Rs1 As New ADODB.Recordset

Sub Main
    Dim sAccessConnect As String
    Dim lNumCariche As Long
    Dim sQuery As String

    'ODBC connection parameters (parametri per connessione ODBC)
    sAccessConnect = "Provider=Microsoft.Jet.OLEDB.4.0;Data " & _
        "Source=" & CurDir & PATH_DB & FILE_DB & _
        ";User ID=Admin;Password=;"

    'Open Connection (apertura connessione)
    Conn1.ConnectionString = sAccessConnect
    Conn1.Open

    sQuery = "SELECT Cariche.* " & _
        "FROM Cariche " & _
        "WHERE ID=1;"

    'Recordset Creation (creazione recordset)
    Rs1.CursorType = adOpenKeyset
    Rs1.LockType = adLockOptimistic
    Rs1.Open sQuery, Conn1, , , adCmdText

    If Not Rs1.EOF Then
        SetVariableValue("VAR00001",Rs1.Fields("RicettaAT").Value)
    End If

    Rs1.Close
    Set Rs1 = Nothing
    Conn1.Close
    Set Conn1 = Nothing
End Sub
```


In questo esempio e' possibile vedere come accedere tramite le funzionalità ADO ai dati contenuti in una tabella MS Access.

Il file "DatiAdo.mdb" contiene la tabella "Cariche", la quale a sua volta contiene il campo "RicettaAT". Mediante la sintassi SQL possiamo selezionare il record della carica con ID=1 ed estrarre il relativo codice di ricetta AT.

Le variabili necessarie sono la "Conn1" di tipo "Connection" destinata a puntare il file mediante il metodo "Open" e la variabile "Rs1" di tipo "Recordset" destinata a contenere il record o l'insieme di record che soddisfino la query Sql di selezione.

Al termine della routine è bene distruggere le variabili "Conn1" e "Rs1" mediante i rispettivi metodi "Close" e "Nothing" al fine di rilasciare la memoria allocata per la creazione delle stesse.

Il motore ADO è thread safe e può essere utilizzato in più thread contemporaneamente.



Nei Sistemi WindowsCE non è possibile nei VB Script utilizzare i References alla classe ADODB ma è necessario creare direttamente gli oggetti ADO tramite la funzione VB CreateObject().

Il parametro della funzione deve corrispondere al nome di registrazione della classe ADOCE all'interno del sistema operativo Windows CE.

Pertanto la dichiarazione degli oggetti ADO deve essere generica e successivamente devono essere istanziati tramite la CreateObject().

Ad esempio:

```
Sub Main()  
Dim oDBConn As Object  
Dim oRS As Object  
Set oDBConn = CreateObject("ADOCE.Connection.3.1")  
Set oRS = CreateObject("ADOCE.Recordset.3.1")  
'...  
End Sub
```

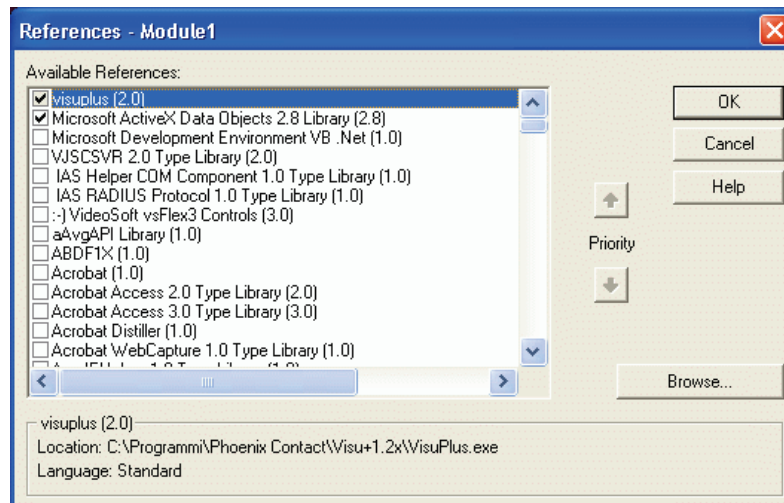
1.12. ActiveX/OCX nei Basic Script

Gli oggetti ActiveX/OCX possono essere utilizzati all'interno delle routines Basic Script. Una risorsa Basic di Movicon può liberamente fare uso di oggetti ActiveX/OCX, oggetti realizzati da terze parti ed indipendenti da Movicon per la realizzazione di funzioni più svariate.

I vantaggi derivanti dall'uso della tecnologia ActiveX/OCX di Microsoft sono, come nel caso dei **ADO**, enormi ed indiscutibili, consentendo ai programmatori di riutilizzare il loro codice all'interno di applicazioni Container diverse.

Prima di gestire un oggetto esterno all'interno di un Basic Script, occorre creare sempre il collegamento, utilizzando la funzione CreateObject o, ancora meglio, utilizzando il collegamento del References. In questo ultimo modo, infatti, le proprietà dell'oggetto saranno disponibili direttamente all'interno del Browser della libreria funzioni di Movicon.

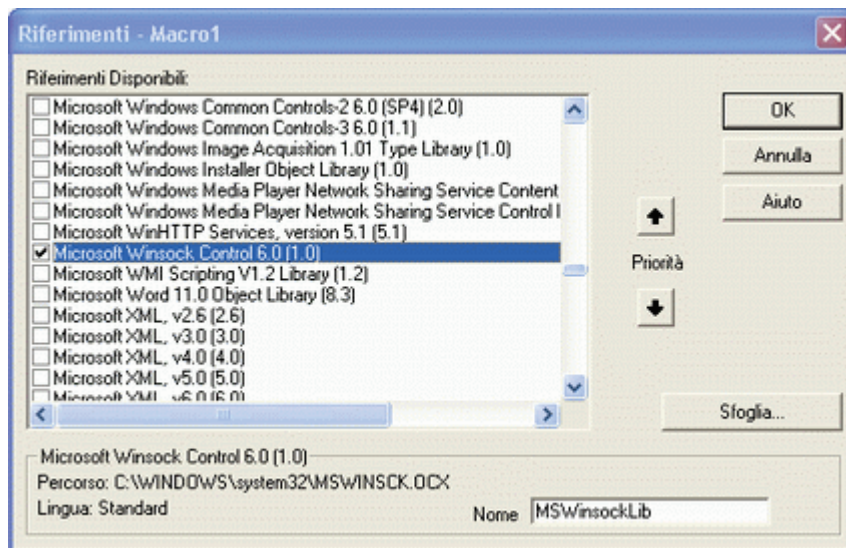
Per eseguire il collegamento, selezionare la voce References dal menu Modifica di Movicon quando è attiva la risorsa Basic. Verrà aperta la finestra illustrata sotto attraverso la quale è possibile "marcare" il collegamento all'oggetto desiderato.



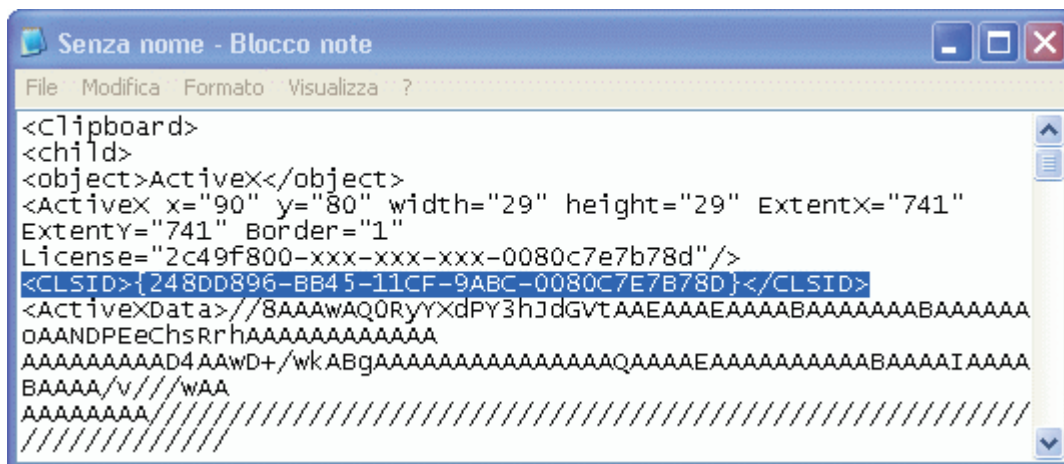
1.12.1. Come utilizzare ActiveX con Licenza

Alcuni oggetti ActiveX hanno bisogno di una licenza per funzionare. In questo caso è possibile utilizzare la funzione script "CreateObjectLic()" per creare un oggetto ActiveX con licenza. Tale funzione necessita di due parametri, il nome del Server ActiveX e la sua licenza. Se non si hanno già a disposizione questi due parametri si possono recuperare nel modo seguente:

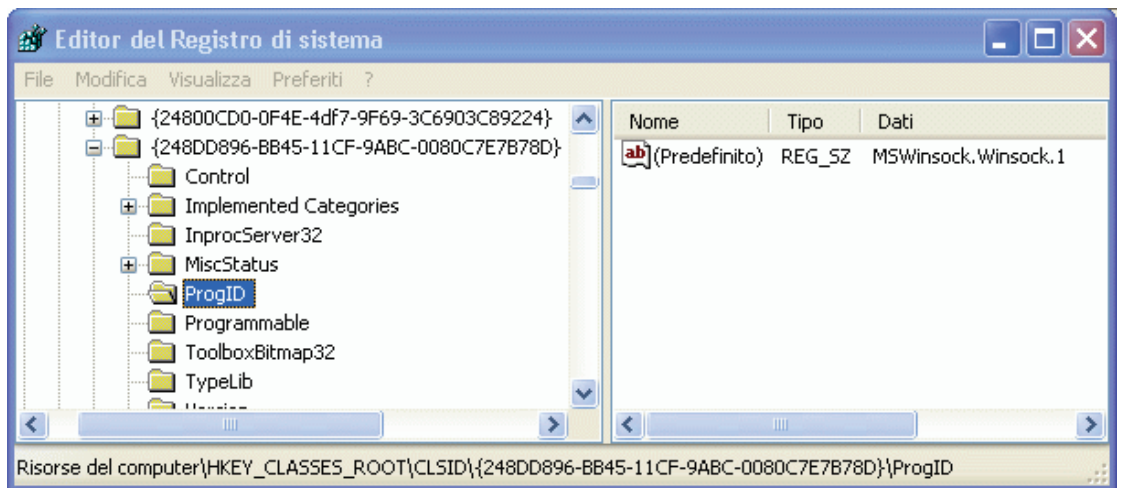
1. Nel Basic Script abilitare il Reference dell'ActiveX da utilizzare. Ad esempio Microsoft WinSock:



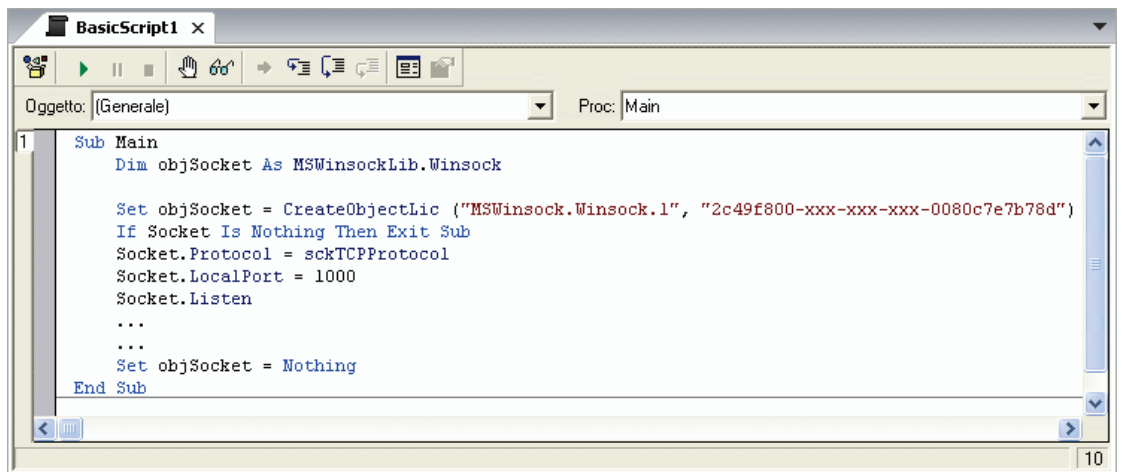
2. Inserire temporaneamente l'ActiveX in un sinottico. Dal gruppo "Oggetti Speciali" della Finestra Oggetti inserire un ActiveX e selezionare quello desiderato, ad esempio Microsoft WinSock. Dalla finestra delle proprietà, tramite il comando "Inserimento Licenza ActiveX" verrà visualizzata la licenza nella proprietà "Licenza ActiveX". Questo sarà il codice da inserire nel parametro "IpszLicense" della funzione "CreateObjectLic()".
3. Per recuperare invece il nome del Server ActiveX, copiare l'oggetto ActiveX inserito nel sinottico e incollarlo su un qualsiasi editor di testo, come il Blocco Note. Cercare a questo punto il CLSID dell'oggetto:



4. Ora aprire il registro di Windows e cercare la chiave corrispondente al CLSID e poi la sottochiave "ProgID". Nel valore predefinito viene riportato il nome del Server ActiveX che andrà poi inserito nel parametro "IpszServerName" della funzione "CreateObjectLic()":



5. A questo punto cancellare l'oggetto ActiveX dal sinottico ed inserire il codice necessario nel Basic Script:



1.12.2. Esempio di utilizzo ActiveX/OCX nei Basic Script

L'esempio riportato sotto illustra l'utilizzo dell'oggetto Microsoft "MSCOMM32.OCX" per la gestione del canale seriale all'interno del Basic Script.

Option Explicit

```
Const BASESTX As Byte = &H20
Const COM_SCRIVI As String = "Q"
Const TIMEOUT As Long = 3 'secondi
Const MAX_TENTATIVI As Long = 5
```

Sub Main

```
'#####
'CREAZIONE STRINGA MESSAGGIO
'#####
```

```
Dim message As String
Dim dati As Long
Dim address As Byte
Dim numByte As Byte
Dim chk As Long
Dim i As Integer
```

```
Begin Dialog UserDialog 270,105,"INSERIMENTO PARAMETRI" ' %GRID:10,5,1,1
Text 10,10,90,20,"Address",.Add
```

```

    TextBox 140,10,110,20,.address
    Text 10,40,90,20,"Dato",.dato
    TextBox 140,40,110,20,.dati
    OKButton 30,70,90,25
    CancelButton 150,70,90,25
    End Dialog
    Dim dlg As UserDialog

    If Dialog (dlg) = 0 Then Exit Sub
    If IsNumeric(CVar(dlg.dati)) And IsNumeric(CVar(dlg.address)) Then
        address = CByte (dlg.address)
        dati = CLng (dlg.dati)
    Else
        MsgBox ("PARAMETRI NON VALIDI", vbCritical, "ERRORE")
        Exit Sub
    End If

    'calcolo num. byte dati come lunghezza della stringa hex di dati diviso come intero per 4
    'caratt. Hex in una word) approssimandolo per eccesso (+3) il tutto moltiplicato per 2 (byte
    per word)
    numByte = ((Len(Hex(dati))+3)\4)*2
    message = Chr(BASESTX + numByte) & COM_SCRIVI
    message = message & Format(address,"00")
    message = message & String((numByte*2-Len(Hex(dati))), "0") & Hex(dati)

    chk = 0
    For i = 2 To Len(message) 'escludo il primo carattere (STX)
        chk = chk + Asc(Mid(message,i,1))
    Next i

    message = message & Right(Hex(chk),2) & vbCr

    '#####
    'INVIO SERIALE
    '#####
    'oggetto per l'OCX MSComm32
    Dim MSComm1 As Object
    'buffer per la stringa di input
    Dim InString As String
    'data e ora di invio messaggio
    Dim oraInvio As Date
    'numero tentativi di invio
    Dim numTentativi As Long

    Set MSComm1 = CreateObject("MSCOMMLib.MSComm.1")

    ' Use COM2.
    MSComm1.CommPort = 2
    ' 9600 baud, no parity, 8 data, and 1 stop bit.
    MSComm1.Settings = "9600,N,8,1"
    'apertura porta seriale
    MSComm1.PortOpen = True
    'predisposizione alla lettura dell'intero buffer di input
    MSComm1.InputLen = 0
    'azzerò contatore tentativi invio con timeout
    numTentativi = 0

Invio:
    'resetto buffer di input
    MSComm1.InBufferCount = 0
    'invio stringa messaggio
    MSComm1.Output = message
    'setto l'ora di invio messaggio
    oraInvio = Now

    'attesa per ritorno di tutti i dati alla porta seriale
    Do
        DoEvents
    Loop Until MSComm1.InBufferCount = Len(message) Or Now >
    DateAdd("s",TIMEOUT,oraInvio)

    If MSComm1.InBufferCount <> 0 Then
        'lettura dati dal buffer di input

```

```

InString = MSComm1.Input
If InString <> message Then
    numTentativi = numTentativi + 1
    If numTentativi < MAX_TENTATIVI Then
        GoTo Invio
    Else
        MsgBox("Errore trasmissione",vbCritical,"ERRORE")
    End If
End If
Else
    MsgBox("Time Out comunicazione",vbCritical,"ERRORE")
End If
' chiusura porta seriale
MSComm1.PortOpen = False
End Sub

```

1.13. API Interfacce Basic

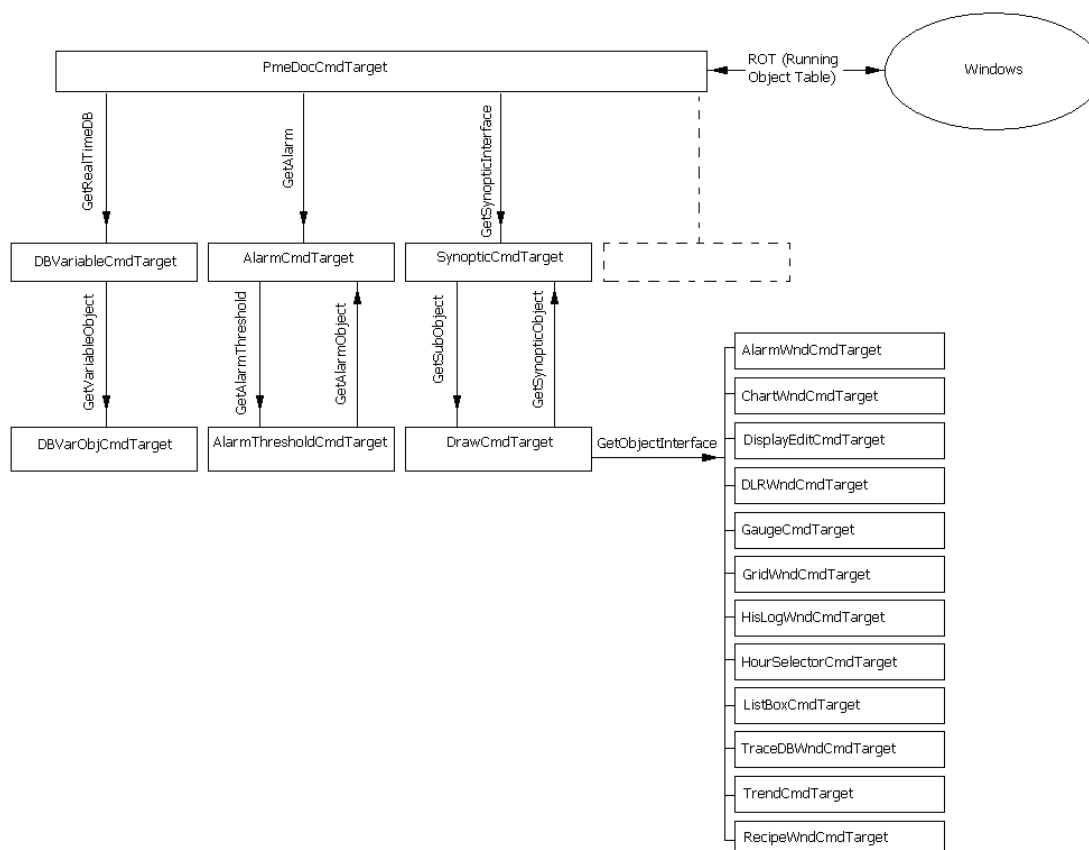
1.14. Utilizzo delle interfacce Basic Script

Movicon consente di poter utilizzare una serie di metodi e proprietà basic script per la gestione del progetto, delle risorse e oggetti in esso contenuti. In questo modo è possibile poter leggere/modificare quasi tutte le proprietà delle risorse e degli oggetti del progetto durante la fase di runtime. A questo scopo ogni risorsa e oggetto del progetto mette a disposizione un set di metodi e proprietà che vengono raggruppati in quelle che vengono chiamate Interfacce Basic Script degli oggetti.

A seconda del contesto basic script in cui ci si trova, ad esempio se si sta editando del codice in una risorsa Basic Script, oppure in un oggetto sinottico, ecc., si avranno a disposizione i metodi e le proprietà dell'interfaccia relativa a quell'oggetto. Per poter avere a disposizione anche i metodi e le proprietà di una diversa interfaccia sarà necessario creare un oggetto referenziato alla nuova interfaccia da utilizzare. Ad esempio, se si sta editando del codice script all'interno di un sinottico sarà disponibile l'interfaccia **"SynopticCmdTarget"**. Per poter accedere all'interfaccia **"DrawCmdTarget"**, ad esempio per modificare il titolo di un simbolo contenuto nel sinottico, si dovrà creare un oggetto di tipo **"DrawCmdTarget"** utilizzando la funzione **"GetSubObject"**.

Si tenga conto che i metodi e le proprietà delle interfacce principali "PmeDocCmdTarget" (interfaccia generale del progetto), "UIInterface" (interfaccia utente del progetto), "DBVariableCmdTarget" (database RealTimeDB delle variabili di progetto) e "IOPortInterface" (interfaccia per accesso alle porte seriali della macchina) sono sempre disponibili indipendentemente dal contesto in cui ci si trova.

L'interfaccia **"PmeDocCmdTarget"** contiene una serie di metodi che consentono di referenziare tutte le risorse del progetto (Allarmi, RealTimeDB, Sinottici, DataLogger/Ricette, ecc.). Nello schema sotto riportato viene mostrato come si possa accedere ad alcuni oggetti o risorse (sono riportati solo alcuni esempi ma il principio di funzionamento è valido per tutte le interfacce basic script) partendo dalla **"PmeDocCmdTarget"**.



Di seguito viene riportato l'elenco di queste Interfacce Basic Script e per ognuna verrà specificato a che tipo di oggetto fa riferimento.

Interfaccia Basic Script	Descrizione
AlarmCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà di un oggetto "Allarme".
AlarmThresholdCmdTarget	Questa interfaccia raggruppa gli eventi, i metodi e le proprietà di un oggetto "Soglia Allarme".
AlarmWndCmdTarget	Questa interfaccia raggruppa gli eventi, i metodi e le proprietà di un oggetto "Finestra Allarmi".
ChartWndCmdTarget	Questa interfaccia raggruppa gli eventi, i metodi e le proprietà di un oggetto "Grafico".
ClientRulesInterface	Questa interfaccia raggruppa i metodi e le proprietà di un oggetto "Regole Client" di Networking.
CommandLanguageCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di modificare i comandi di tipo "Lingua" della Lista Comandi di un Pulsante o di un Oggetto del sinottico.
CommandsListCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di modificare la Lista Comandi di un Pulsante o di un Oggetto del sinottico.
ButtonCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di modificare le proprietà di un oggetto Pulsante.

CommandUsersCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di modificare i comandi di tipo "Utenti" della Lista Comandi di un Pulsante o di un Oggetto del sinottico.
CommandAlarmCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di modificare i comandi di tipo "Allarme" della Lista Comandi di un Pulsante o di un Oggetto del sinottico.
CommandVariableCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di modificare i comandi di tipo "Variabile" della Lista Comandi di un Pulsante o di un Oggetto del sinottico.
CommandBaseCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà di base comuni a tutte le interfacce script utilizzate per la modifica dei singoli comandi della Lista Comandi di un Pulsante o di un Oggetto del sinottico.
CommandHelpCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di modificare i comandi di tipo "Help" della Lista Comandi di un Pulsante o di un Oggetto del sinottico.
CommandMenuCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di modificare i comandi di tipo "Menu" della Lista Comandi di un Pulsante o di un Oggetto del sinottico.
CommandReportCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di modificare i comandi di tipo "Report/Ricette" della Lista Comandi di un Pulsante o di un Oggetto del sinottico.
CommandScriptCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di modificare i comandi di tipo "Script" della Lista Comandi di un Pulsante o di un Oggetto del sinottico.
CommandSynopticCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di modificare i comandi di tipo "Sinottico" della Lista Comandi di un Pulsante o di un Oggetto del sinottico.
CommandSystemCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di modificare i comandi di tipo "Sistema" della Lista Comandi di un Pulsante o di un Oggetto del sinottico.
CommandEventCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di modificare i comandi di tipo "Evento" della Lista Comandi di un Pulsante o di un Oggetto del sinottico.
DBVariableCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà per accedere alle impostazioni del RealTimeDB.
DBVarObjCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà di un oggetto "Variabile (Tag)" del RealTimeDB.
DisplayEditCmdTarget	Questa interfaccia raggruppa gli eventi, i metodi e le proprietà di un oggetto "Display". Si consideri che un oggetto "List Box" o "Combo Box" ereditano i metodi e le proprietà sia dell'interfaccia "DisplayEditCmdTarget" che "ListBoxCmdTarget".
DLRCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà di un oggetto "DataLogger" o di un oggetto "Ricetta".
DLRColumnCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà di un oggetto "Colonna" di un DataLogger o di una Ricetta.
DLRWndCmdTarget	Questa interfaccia raggruppa gli eventi, i metodi e le proprietà di un oggetto "Finestra DataLogger-Ricetta".

DrawCmdTarget	Questa interfaccia raggruppa gli eventi, i metodi e le proprietà generiche disponibili per tutti gli oggetti che possono essere inseriti all'interno di un sinottico.
EventCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà di un oggetto "Evento" delle risorse di progetto.
GaugeCmdTarget	Questa interfaccia raggruppa gli eventi, i metodi e le proprietà di un oggetto "Potenziometro".
GenericEvents	Questo gruppo di eventi sono generici e possono essere disponibili, tutti o in parte, per tutti gli oggetti che possono essere inseriti all'interno di un sinottico.
GridWndCmdTarget	Questa interfaccia raggruppa gli eventi, i metodi e le proprietà di un oggetto "Griglia".
HisLogWndCmdTarget	Questa interfaccia raggruppa gli eventi, i metodi e le proprietà di un oggetto "Finestra Log Storico".
HourSelectorCmdTarget	Questa interfaccia raggruppa gli eventi, i metodi e le proprietà di un oggetto "Finestra Scheduler".
IOPortInterface	Questa interfaccia raggruppa i metodi e le proprietà che consentono di accedere alle porte seriali COM del PC.
ListBoxCmdTarget	Questa interfaccia raggruppa gli eventi, i metodi e le proprietà di un oggetto "List Box" o "Combo Box".
NetworkClientCmd	Questa interfaccia raggruppa i metodi e le proprietà che consentono di accedere alle impostazioni "Network Client" del progetto.
NetworkRedundancyCmd	Questa interfaccia raggruppa i metodi e le proprietà che consentono di accedere alle impostazioni di "Ridondanza" del progetto.
OPCAECmdTarget	Interfaccia al momento non supportata.
OPCClientCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di accedere alle impostazioni generali dell'OPC Client del progetto.
OPCClientGroupObjCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di accedere alle impostazioni di un "Gruppo" OPC Client del progetto.
OPCClientItemObjCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di accedere alle impostazioni di un "Item" OPC Client del progetto.
OPCClientObjCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di accedere alle impostazioni di uno specifico OPC Client del progetto.
OPCServerCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà che consentono di accedere alle impostazioni dell'OPC Server Movicon.
PmeDocCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà generiche del progetto.
RASStationInterface	Questa interfaccia raggruppa i metodi e le proprietà di una "Stazione RAS" di Networking.

RecipeWndCmdTarget	Questa interfaccia raggruppa gli eventi, i metodi e le proprietà di un oggetto "Finestra Gestione Ricette".
ScalingCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà di un oggetto "Normalizzatore" delle risorse di progetto.
SchedulerCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà di un oggetto "Schedulatore" delle risorse di progetto.
ScriptMEInterface	Questa interfaccia raggruppa gli eventi, i metodi e le proprietà di una risorsa "Basic Script" del progetto.
SynopticCmdTarget	Questa interfaccia raggruppa gli eventi, i metodi e le proprietà di una risorsa "Sinottico" del progetto.
TraceDBWndCmdTarget	Questa interfaccia raggruppa gli eventi, i metodi e le proprietà di un oggetto "Finestra DB Trace".
TrendCmdTarget	Questa interfaccia raggruppa gli eventi, i metodi e le proprietà di un oggetto "Trend" o "Data Analysis".
UIInterface	Questa interfaccia raggruppa i metodi e le proprietà generiche del progetto che riguardano l'interfaccia utente.
UserAndGroupCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà generiche della "Gestione Utenti" del progetto.
UserCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà di un oggetto "Utente" del progetto.
UserGroupCmdTarget	Questa interfaccia raggruppa i metodi e le proprietà di un oggetto "Gruppo Utenti" del progetto.

1.14.1. AlarmCmdTarget

Functions

GetAlarmThreshold, AlarmCmdTarget Function

Syntax GetAlarmThreshold(_IpszName)

Description Restituisce un oggetto di tipo AlarmThresholdCmdTarget per la gestione delle soglie dell'allarme.

Parameter	Description
IpszName As String	Nome della soglia da reperire.

Result Object
Viene restituito un oggetto di tipo AlarmThresholdCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub AlarmLoading()
    Dim objAlarm As AlarmCmdTarget
    Dim objThreshold As AlarmThresholdCmdTarget
```

```

Set objAlarm = GetAlarmObject
Set objThreshold = objAlarm.GetAlarmThreshold("High")
Debug.Print objThreshold.BackColor
Set objThreshold = Nothing
Set objAlarm = Nothing
End Sub

```

GetXMLSettings, AlarmCmdTarget Function

Syntax GetXMLSettings()

Description La funzione restituisce la stringa di definizione dell'oggetto allarme in formato XML.

Parameter	Description
None	None

Result String

Example:

```

Public Sub AlarmLoading()
    Dim objAlarm As AlarmCmdTarget
    Dim sResult As String
    Set objAlarm = GetAlarmObject
    sResult = objAlarm.GetXMLSettings
    MsgBox("XMLSetting: " & sResult, vbOkOnly, GetProjectTitle)
    Set objAlarm = Nothing
End Sub

```

Properties

AlarmOnQualityGood, AlarmCmdTarget Property

Syntax AlarmOnQualityGood = _Boolean

Description Abilita o disabilita la gestione dell'allarme solamente quando la qualità della variabile ad esso collegata è buona. Questo significa, ad esempio, che gli allarmi associati a variabili che arrivano direttamente dal PLC o da un qualsiasi altro dispositivo di campo, saranno automaticamente disabilitati nel caso in cui la comunicazione si interrompa.

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub AlarmLoading()
    Dim objAlarm As AlarmCmdTarget
    Dim bResult As Boolean
    Set objAlarm = GetAlarmObject

```

```

        bResult = objAlarm.AlarmOnQualityGood
        Debug.Print bResult
        Set objAlarm = Nothing
    End Sub

```

DeviceName, AlarmCmdTarget Property

Syntax DeviceName

Description Questa proprietà imposta o restituisce il nome del dispositivo associato all'allarme.
Questa proprietà è di sola lettura.

Parameter	Description
None	None

Result String

Example:

```

Public Sub AlarmLoading()
    Dim objAlarm As AlarmCmdTarget
    Dim sResult As String
    Set objAlarm = GetAlarmObject
    sResult = objAlarm.DeviceName
    Debug.Print sResult
    Set objAlarm = Nothing
End Sub

```

Enabled, AlarmCmdTarget Property

Syntax Enabled = _Boolean

Description Abilita o disabilita l'allarme.

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub AlarmLoading()
    Dim objAlarm As AlarmCmdTarget
    Dim bResult As Boolean
    Set objAlarm = GetAlarmObject
    bResult = objAlarm.Enabled
    Debug.Print bResult
    Set objAlarm = Nothing
End Sub

```

EnableDispatchingVariableName, AlarmCmdTarget Property

Syntax _EnableDispatchingVariableName

Description Imposta o restituisce il nome della variabile associata alla Proprietà "Tag abilitazione Invio messaggi".

Parameter	Description
None	None

Result String

Example:

Option Explicit

```
Public Sub Click()  
    Dim objAlarm As AlarmCmdTarget  
    Dim sResult As String  
    Set objAlarm = GetAlarm("Analog Alarm")  
    sResult = objAlarm.EnableDispatchingVariableName  
    MsgBox sResult  
    Set objAlarm = Nothing  
End Sub
```

EnableVariableName, AlarmCmdTarget Property

Syntax EnableVariableName

Description Questa proprietà restituisce, in formato stringa, il nome della variabile di abilitazione associata all'allarme. Questa proprietà e' di sola lettura.

Parameter	Description
None	None

Result String

Example:

```
Public Sub AlarmLoading()  
    Dim objAlarm As AlarmCmdTarget  
    Dim sResult As String  
    Set objAlarm = GetAlarmObject  
    sResult = objAlarm.EnableVariableName  
    Debug.Print sResult  
    Set objAlarm = Nothing  
End Sub
```

Isteresis, AlarmCmdTarget Property

Syntax Isteresis = _Long

Description Questa proprietà restituisce il valore di isteresi dell'allarme. Questa proprietà e' di sola lettura.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub AlarmLoading()  
    Dim objAlarm As AlarmCmdTarget  
    Dim IResult As Long  
    Set objAlarm = GetAlarmObject  
    IResult = objAlarm.Isteresis  
    Debug.Print sResult  
    Set objAlarm = Nothing  
End Sub
```

Name, AlarmCmdTarget Property

Syntax Name

Description Questa proprietà imposta o restituisce il nome dell'allarme.

Parameter	Description
None	None

Result String

Example:

```
Public Sub AlarmLoading()  
    Dim objAlarm As AlarmCmdTarget  
    Dim sResult As String  
    Set objAlarm = GetAlarmObject  
    sResult = objAlarm.Name  
    Debug.Print sResult  
    Set objAlarm = Nothing  
End Sub
```

ThresholdExclusive, AlarmCmdTarget Property

Syntax ThresholdExclusive = _Boolean

Description L'abilitazione di questa proprietà comporterà la scomparsa di un allarme riferito ad una soglia quando comparirà l'allarme riferito alla soglia successiva. In caso contrario, i due allarmi saranno mantenuti attivi entrambi.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Dim objAlarm As AlarmCmdTarget  
    Dim bResult As Boolean  
    Set objAlarm = GetAlarmObject  
    bResult = objAlarm.ThresholdExclusive  
    Debug.Print bResult  
    Set objAlarm = Nothing  
End Sub
```

VariableName, AlarmCmdTarget Property

Syntax VariableName

Description Questa proprietà restituisce, in formato stringa, il nome della variabile associata all'allarme. Questa proprietà e' di sola lettura.

Parameter	Description
None	None

Result String

Example:

```
Public Sub AlarmLoading()  
    Dim objAlarm As AlarmCmdTarget  
    Dim sResult As String  
    Set objAlarm = GetAlarmObject  
    sResult = objAlarm.VariableName  
    Debug.Print sResult  
    Set objAlarm = Nothing  
End Sub
```

1.14.2. AlarmThresholdCmdTarget

Events

AlarmLoading, AlarmThresholdCmdTarget Event

Description Evento notificato quando l'oggetto allarme viene inizializzato cioè al run del progetto.

Parameter	Description
None	None

AlarmUnloading, AlarmThresholdCmdTarget Event

Description Evento notificato quando l'oggetto allarme viene scaricato dalla memoria cioè allo stop del progetto.

Parameter	Description
None	None

OnAckAlarm, AlarmThresholdCmdTarget Event

Description Evento notificato al riconoscimento dell'allarme cioè quando viene eseguito il comando di ack sull'allarme stesso. La variabile booleana bRet viene acquisita dall'evento e consente o meno l'esecuzione del riconoscimento dell'allarme, cioè se bRet viene impostata False l' allarme non viene riconosciuto.

Parameter	Description
bRet As Boolean	Abilitazione al riconoscimento dell'allarme

OnCommentAlarm, AlarmThresholdCmdTarget Event

Description Evento notificato quando sull'oggetto allarme/messaggio viene aggiunto o modificato il commento associato. Questa operazione è possibile dall'oggetto finestra allarmi o messaggi tramite gli appositi comandi. La variabile bRet permette di interrompere questo evento ed impedire quindi che venga aggiunto un commento.

Parameter	Description
bRet As Boolean	Abilitazione all'inserimento del commento

OnHelpAlarm, AlarmThresholdCmdTarget Event

Description Evento notificato quando sull'oggetto allarme/messaggio viene richiesta la visualizzazione della stringa di help. Questa operazione è possibile dall'oggetto finestra allarmi o messaggi tramite gli appositi comandi.

La variabile bRet permette di interrompere questo evento ed impedire quindi la visualizzazione del testo di help.

Parameter	Description
bRet As Boolean	Abilitazione alla visualizzazione del testo di help

OnResetAlarm, AlarmThresholdCmdTarget Event

Description Evento notificato al reset dell'allarme cioè quando viene eseguito il comando di reset sull'allarme stesso. La variabile booleana bRet consente o meno l'esecuzione del reset dell'allarme, cioè se bRet viene impostata False l'allarme non viene resettato.

Parameter	Description
bRet As Boolean	Abilitazione al reset dell'allarme

OnSetAlarm, AlarmThresholdCmdTarget Event

Description Evento notificato all'innescarsi o al disinnescarsi dell'allarme. La variabile booleana bSet ritorna lo stato dell'allarme (ON = True, OFF = False) mentre la variabile booleana bRet consente o meno l'innescarsi o il disinnescarsi dell'allarme, cioè se bRet viene impostata False non sarà possibile che l'allarme vada ON se e' OFF o viceversa.

Parameter	Description
bSet As Boolean	Stato dell'allarme
bRet As Boolean	Abilitazione al cambio di stato dell'allarme

Functions

GetTransactionID, AlarmThresholdCmdTarget Function

Syntax GetTransactionID()

Description Questa proprietà restituisce il numero di transazione TransactionID che l'allarme ha raggiunto.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub OnAckAlarm()  
    Dim objAlarm As AlarmCmdTarget  
    Dim objAlarmThreshold As AlarmThresholdCmdTarget  
    Set objAlarm = GetAlarm("Alarm01")  
    Set objAlarmThreshold = objAlarm.GetAlarmThreshold("Threshold01")  
    MsgBox " GetTransactionID = " &  
    CStr(objAlarmThreshold.GetTransactionID),vbInformation,GetProjectTitle  
    Set objAlarm = Nothing  
    Set objAlarmThreshold = Nothing  
End Sub
```

GetUniqueID, AlarmThresholdCmdTarget Function

Syntax GetUniqueID()

Description Questa proprietà restituisce il numero univoco UniqueID associato all'allarme.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub OnAckAlarm()  
    Dim objAlarm As AlarmCmdTarget  
    Dim objAlarmThreshold As AlarmThresholdCmdTarget  
    Set objAlarm = GetAlarm("Alarm01")  
    Set objAlarmThreshold = objAlarm.GetAlarmThreshold("Threshold01")  
    MsgBox " GetUniqueID = " &  
    CStr(objAlarmThreshold.GetUniqueID),vbInformation,GetProjectTitle  
    Set objAlarm = Nothing  
    Set objAlarmThreshold = Nothing  
End Sub
```

GetTotNumReset, AlarmThresholdCmdTarget Function

Syntax GetTotNumReset()

Description Questa proprietà restituisce il numero di volte che l'allarme è stato resettato.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub OnAckAlarm()  
    Dim objAlarm As AlarmCmdTarget  
    Dim objAlarmThreshold As AlarmThresholdCmdTarget  
    Set objAlarm = GetAlarm("Alarm01")  
    Set objAlarmThreshold = objAlarm.GetAlarmThreshold("Threshold01")  
    MsgBox " GetTotNumAck = " &  
        CStr(objAlarmThreshold.GetTotNumReset),vbInformation,GetProjectTitle  
    Set objAlarm = Nothing  
    Set objAlarmThreshold = Nothing  
End Sub
```

GetTotNumAck, AlarmThresholdCmdTarget Function

Syntax GetTotNumAck()

Description Questa proprietà restituisce il numero di volte che l'allarme è stato riconosciuto.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub OnAckAlarm()  
    Dim objAlarm As AlarmCmdTarget  
    Dim objAlarmThreshold As AlarmThresholdCmdTarget  
    Set objAlarm = GetAlarm("Alarm01")  
    Set objAlarmThreshold = objAlarm.GetAlarmThreshold("Threshold01")  
    MsgBox " GetTotNumAck = " &  
        CStr(objAlarmThreshold.GetTotNumAck),vbInformation,GetProjectTitle  
    Set objAlarm = Nothing  
    Set objAlarmThreshold = Nothing  
End Sub
```

GetTotNumOn, AlarmThresholdCmdTarget Function

Syntax GetTotNumOn()

Description Questa proprietà restituisce il numero di volte che l'allarme è andato ad ON.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub OnAckAlarm()  
    Dim objAlarm As AlarmCmdTarget  
    Dim objAlarmThreshold As AlarmThresholdCmdTarget  
    Set objAlarm = GetAlarm("Alarm01")  
    Set objAlarmThreshold = objAlarm.GetAlarmThreshold("Threshold01")  
    MsgBox " GetTotNumOn = " &  
        CStr(objAlarmThreshold.GetTotNumOn),vbInformation,GetProjectTitle  
    Set objAlarm = Nothing  
    Set objAlarmThreshold = Nothing  
End Sub
```

ResetAlarm, AlarmThresholdCmdTarget Function

Syntax ResetAlarm()

Description Resetta l'allarme relativo alla soglia referenziata.

Parameter	Description
None	None

Result None

Example:

```
Public Sub OnAckAlarm()  
    ..  
    ResetAlarm  
    ..  
End Sub
```

GetAlarmObject, AlarmThresholdCmdTarget Function

Syntax GetAlarmObject()

Description Reperisce l'oggetto allarme relativo alla soglia referenziata.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo AlarmCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub AlarmLoading()
```

```

Dim objAlarm As AlarmCmdTarget
Dim sResult As String
Set objAlarm = GetAlarmObject
sResult = objAlarm.GetXMLSettings
MsgBox("XMLSetting: " & sResult, vbOkOnly, GetProjectTitle)
Set objAlarm = Nothing
End Sub

```

GetXMLSettings, AlarmThresholdCmdTarget Function

Syntax GetXMLSettings()

Description La funzione restituisce la stringa di definizione dell'oggetto allarme in formato XML.

Parameter	Description
None	None

Result String

Example:

```

Public Sub AlarmLoading()
Dim sResult As String
sResult = GetXMLSettings
MsgBox("XMLSetting: " & sResult, vbOkOnly, GetProjectTitle)
End Sub

```

AckAlarm, AlarmThresholdCmdTarget Function

Syntax AckAlarm()

Description Esegue il riconoscimento (Ack) dell'allarme relativo alla soglia referenziata.

Parameter	Description
None	None

Result None

Example:

```

Public Sub AlarmLoading()
AckAlarm
End Sub

```

Properties

AlarmArea, AlarmThresholdCmdTarget Property

Syntax AlarmArea = _String

Description Questa proprietà imposta o restituisce l'area allarmi o l'area messaggi a cui appartiene la stringa associata al testo della soglia. Accetta un parametro di tipo stringa.

Parameter	Description
None	None

Result String

Example:
Public Sub AlarmLoading()
 Debug.Print **AlarmArea**
End Sub

Attachment, AlarmThresholdCmdTarget Property

Syntax Attachment = _String

Description Tramite questa proprietà è possibile reperire o impostare il file (uno o più di uno) da allegare al messaggio di posta da spedire al destinatario. Nel caso i file siano più di uno, utilizzare il carattere ";" come separatore (Es. File1.zip;File2.Zip;File3.zip).

Parameter	Description
None	None

Result String

Example:
Public Sub AlarmLoading()
 Debug.Print **Attachment**
End Sub

BackColor, AlarmThresholdCmdTarget Property

Syntax BackColor = _Long

Description Questa proprietà imposta o restituisce il colore di sfondo relativo all'allarme o messaggio visualizzato nella apposita finestra.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub AlarmLoading()
    Debug.Print BackColor
End Sub
```

Beep, AlarmThresholdCmdTarget Property

Syntax Beep = _Boolean

Description Questa proprietà permette di attivare o disattivare l'avviso acustico generato dal cicalino del computer quando l'allarme va ON.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()
    Debug.Print Beep
End Sub
```

BlinkBackColor, AlarmThresholdCmdTarget Property

Syntax BlinkBackColor = _Long

Description Questa proprietà imposta o restituisce il colore da associare allo sfondo relativo all'allarme o messaggio visualizzato nella apposita finestra durante la fase di lampeggio o blinking.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub AlarmLoading()
    Debug.Print BlinkBackColor
End Sub
```

BlinkOnNewAlarm, AlarmThresholdCmdTarget Property

Syntax BlinkOnNewAlarm = _Boolean

Description Tramite questa proprietà è possibile determinare se attivare o meno la funzione di lampeggio dell'allarme al momento della sua attivazione.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print BlinkOnNewAlarm  
End Sub
```

BlinkTextColor, AlarmThresholdCmdTarget Property

Syntax BlinkTextColor = _Long

Description Questa proprietà imposta o restituisce il colore da associare al testo relativo all'allarme o messaggio visualizzato nella apposita finestra durante la fase di lampeggio o blinking.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub AlarmLoading()  
    Debug.Print BlinkTextColor  
End Sub
```

CommentOnAck, AlarmThresholdCmdTarget Property

Syntax CommentOnAck = _Boolean

Description Consente di impostare o resettare l'opzione per l'inserimento di un commento sul riconoscimento dell'allarme.

Parameter	Description
-----------	-------------

None	None
------	------

Result Boolean

Example:

```
Public Sub Click()
    Dim objAlarmWnd As AlarmWndCmdTarget
    Dim objAlarmThr As AlarmThresholdCmdTarget
    Set objAlarmWnd = objAlarmWnd =
    GetSynopticObject.GetSubObject("AlarmWnd").GetObjectInterface
    Set objAlarmThr = objAlarmWnd.GetSelectedAlarm
    If Not objAlarmThr Is Nothing Then
        objAlarmThr.CommentOnAck = Not objAlarmThr.CommentOnAck
    End If
End Sub
```

CommandList..., AlarmThresholdCmdTarget Property

Syntax CommandList... = _String

Description Questa proprietà restituisce la stringa XML di progetto contenente la definizione della "Lista Comandi" associata alla soglia di allarme referenziata, che dovrà essere eseguita sul rispettivo evento:

CommandList: la lista comandi viene eseguita su richiesta dell'operatore quando l'allarme è attivo. Solo in questo caso, dalla Finestra Allarmi, eseguendo un doppio click con il mouse sull'allarme e tenendo allo stesso tempo premuto il tasto CTRL, la Lista Comandi verrà eseguita.

CommandListAck: la lista comandi viene eseguita sull'evento di riconoscimento dell'allarme.

CommandListOff: la lista comandi viene eseguita sull'evento di disattivazione (OFF) dell'allarme.

CommandListOn: la lista comandi viene eseguita sull'evento di attivazione (ON) dell'allarme.

CommandListReset: la lista comandi viene eseguita sull'evento di reset dell'allarme.

Parameter	Description
None	None

Result String

Example:

```
Public Sub AlarmLoading()
    Debug.Print CommandList
    Debug.Print CommandListAck
    Debug.Print CommandListOff
    Debug.Print CommandListOn
    Debug.Print CommandListReset
End Sub
```


Condition, AlarmThresholdCmdTarget Property

Syntax Condition = _Integer

Description Questa proprietà imposta o restituisce la condizione per cui si verifica l'allarme referenziato.
Le possibili configurazioni sono:

- 0 = maggiore-uguale (>=)
- 1 = minore-uguale (<=)
- 2 = uguale (=)
- 3 = Decremento Valore nel Tempo
- 4 = Incremento Valore nel Tempo
- 5 = Diverso (<>)
- 6 = Compreso tra (Between)

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub AlarmLoading()  
    MsgBox "Alarm Condition = " & cstr(Condition), vbInformation, GetProjectTitle  
End Sub
```

DateTimeACK, AlarmThresholdCmdTarget Property

Syntax DateTimeACK = _Date

Description Questa proprietà imposta o restituisce la data e l'ora in cui è stato effettuato il riconoscimento dell'allarme. Accetta un parametro di tipo data. Questa proprietà modifica solo il valore visualizzato nella finestra allarmi, non influisce sul log storico.

Parameter	Description
None	None

Result Date

Example:

```
Public Sub OnResetAlarm(bRet As Boolean)  
    Debug.Print DateTimeAck  
End Sub
```

DateTimeACKMs, AlarmThresholdCmdTarget Property

Syntax DateTimeACKMs = _Integer

Description Questa proprietà imposta o restituisce i millisecondi del secondo in cui è stato effettuato il riconoscimento dell'allarme. Per avere il quadro completo utilizzare abbinata a DateTimeAck facendo attenzione al tipo di variabili differente (formato data e intero per questa proprietà). Questa proprietà modifica solo il valore visualizzato nella finestra allarmi, non influisce sul log storico.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub OnResetAlarm(bRet As Boolean)
    Debug.Print "ON = " & Format(DateTimeOn,"yyyy/mm/dd hh.nn.ss") & "," &
    DateTimeOnMs    ' ON = 2001/03/16 11.27.17,10
    Debug.Print "ACK = " & Format(DateTimeAck,"yyyy/mm/dd hh.nn.ss") & "," &
    DateTimeACKMs   ' ACK = 2001/03/16 11.27.24,210
End Sub
```

DateTimeFromTimeStamp, AlarmThresholdCmdTarget Property

Syntax DateTimeFromTimeStamp = _Boolean

Description Questa proprietà imposta o restituisce lo stato di attivazione della proprietà della soglia dell'allarme "Use TimeStamp della Variabile" tramite la quale è possibile far coincidere il TimeStamp dell'allarme a quello della variabile.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim objAlarm As AlarmCmdTarget
    Dim objAlarmThreshold As AlarmThresholdCmdTarget
    Set objAlarm = GetAlarm("Alarm01")
    Set objAlarmThreshold = objAlarm.GetAlarmThreshold("Threshold01")
    MsgBox " DateTimeFromTimeStamp = " &
    CStr(objAlarmThreshold.DateTimeFromTimeStamp),vbInformation,GetProject
    Title
    Set objAlarm = Nothing
    Set objAlarmThreshold = Nothing
End Sub
```

DateTimeOFF, AlarmThresholdCmdTarget Property

Syntax DateTimeOFF = _Date

Description Questa proprietà imposta o restituisce la data e l'ora in cui l'allarme è andato OFF. Accetta un parametro di tipo data. Questa proprietà modifica solo il valore visualizzato nella finestra allarmi, non influisce sul log storico.

Parameter	Description
None	None

Result Date

Example:

```
Public Sub OnResetAlarm(bRet As Boolean)
    DateTimeOFF = CDate(Date)
    Debug.Print DateTimeOFF
End Sub
```

DateTimeOFFMs, AlarmThresholdCmdTarget Property

Syntax DateTimeOFFMs = _Integer

Description Questa proprietà imposta o restituisce i millisecondi del secondo in cui l'allarme è andato OFF. Per avere il quadro completo utilizzare abbinata a DateTimeOFF facendo attenzione al tipo di variabili differente (formato data e intero per questa proprietà). Questa proprietà modifica solo il valore visualizzato nella finestra allarmi, non influisce sul log storico.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub OnResetAlarm(bRet As Boolean)
    Debug.Print "OFF = " & Format(DateTimeOff,"yyyy/mm/dd hh.nn.ss") & ", " &
    DateTimeOFFMs
End Sub
```

DateTimeON, AlarmThresholdCmdTarget Property

Syntax DateTimeON = _Date

Description Questa proprietà imposta o restituisce la data e l'ora in cui l'allarme e' andato ON. Accetta un parametro di tipo data. Questa proprietà modifica solo il valore visualizzato nella finestra allarmi, non influisce sul log storico.

Parameter	Description
None	None

Result Date

Example:

```
Public Sub OnAckAlarm(bRet As Boolean)
    DateTimeOn = CDate(Date)
    Debug.Print DateTimeOn
End Sub
```

DateTimeOnMs, AlarmThresholdCmdTarget Property

Syntax DateTimeOnMs = _Integer

Description Questa proprietà imposta o restituisce i millisecondi del secondo in cui l'allarme e' andato ON. Per avere il quadro completo utilizzare abbinata a DateTimeOn facendo attenzione al tipo di variabili differente (formato data e intero per questa proprietà). Questa proprietà modifica solo il valore visualizzato nella finestra allarmi, non influisce sul log storico.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub OnResetAlarm(bRet As Boolean)
    Debug.Print "ON = " & Format(DateTimeOn,"yyyy/mm/dd hh.nn.ss") & "," &
    DateTimeOnMs ' ON = 2001/03/16 11.27.17,10
    Debug.Print "ACK = " & Format(DateTimeAck,"yyyy/mm/dd hh.nn.ss") & "," &
    DateTimeAckMs ' ACK = 2001/03/16 11.27.24,210
End Sub
```

DateTimeRESET, AlarmThresholdCmdTarget Property

Syntax DateTimeRESET = _Date

Description Questa proprietà imposta o restituisce la data e l'ora in cui è stato effettuato il reset dell'allarme. Accetta un parametro di tipo data. Questa proprietà modifica solo il valore visualizzato nella finestra allarmi, non influisce sul log storico.

Parameter	Description
-----------	-------------

None	None
------	------

Result Date

Example:

```
Public Sub OnResetAlarm(bRet As Boolean)
    Debug.Print DateTimeRESET
End Sub
```

DateTimeRESETMs, AlarmThresholdCmdTarget Property

Syntax DateTimeRESETMs = _Integer

Description Questa proprietà imposta o restituisce i millisecondi del secondo in cui è stato effettuato il reset dell'allarme. Per avere il quadro completo utilizzare abbinata a DateTimeAck facendo attenzione al tipo di variabili differente (formato data e intero per questa proprietà). Questa proprietà modifica solo il valore visualizzato nella finestra allarmi, non influisce sul log storico.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub OnResetAlarm(bRet As Boolean)
    Debug.Print "RESET = " & Format(DateTimeRESET,"yyyy/mm/dd hh.nn.ss") & ","
    & DateTimeRESETMs
End Sub
```

DurationFormat, AlarmThresholdCmdTarget Property

Syntax DurationFormat = _String

Description Questa proprietà si può inserire il messaggio che verrà archiviato nella tabella "Alarms" del Log Storico per la colonna "CommCol". La registrazione del messaggio verrà fatta soltanto sull'evento di "Allarme Off". Il messaggio potrà includere soltanto i seguenti codici speciali:

- **%D** = giorni di durata dell'allarme
- **%H** = Ora di durata dell'allarme
- **%M** = Minuti di durata dell'allarme
- **%S** = Secondi di durata dell'allarme

Se il campo di introduzione viene lasciato vuoto, Movicon inserirà automaticamente la durata complessiva dell'allarme con la seguente stringa:

Durata totale 0,00:00:00

dove 0,00:00:00 indica il tempo di durata dell'allarme in giorni, ore, minuti e secondi.

Parameter	Description
None	None

Result String

Example:

```
Public Sub OnResetAlarm(bRet As Boolean)
    Debug.Print DurationFormat
End Sub
```

Help, AlarmThresholdCmdTarget Property

Syntax Help = lpar

Description Questa proprietà imposta o restituisce l'ID della stringa selezionata per l'help dell'allarme. Accetta un parametro di tipo stringa.

Parameter	Description
None	None

Result String

Example:

```
Public Sub AlarmLoading()
    Help = "STR00001"
    Debug.Print Help 'Return-> STR00001
End Sub
```

LastComment, AlarmThresholdCmdTarget Property

Syntax LastComment = _String

Description Questa proprietà consente di leggere o scrivere un commento per l'allarme in questione. Tale commento sarà lo stesso che può anche essere inserito o letto utilizzando il "Pulsante Commento" della Finestra Allarmi.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim objAlarm As AlarmCmdTarget
    Dim objAlarmThreshold As AlarmThresholdCmdTarget
    Set objAlarm = GetAlarm("Alarm01")
```

```

Set objAlarmThreshold = objAlarm.GetAlarmThreshold("Threshold01")
MsgBox " LastComment = " & CStr(objAlarmThreshold.LastComment
),vbInformation,GetProjectTitle
Set objAlarm = Nothing
Set objAlarmThreshold = Nothing
End Sub

```

LastTotalTimeOn, AlarmThresholdCmdTarget Property

Syntax LastTotalTimeOn

Description Questa proprietà restituisce un valore di tipo data che indica l'orario in cui l'allarme è andato ON per l'ultima volta. Quando l'allarme va OFF questo valore viene azzerato.

Parameter	Description
None	None

Result Date

Example:

```

Public Sub Click()
Dim objAlarm As AlarmCmdTarget
Dim objAlarmThreshold As AlarmThresholdCmdTarget
Set objAlarm = GetAlarm("Alarm01")
Set objAlarmThreshold = objAlarm.GetAlarmThreshold("Threshold01")
MsgBox " LastTotalTimeOn= " &
CStr(objAlarmThreshold.LastTotalTimeOn),vbInformation,GetProjectTitle
Set objAlarm = Nothing
Set objAlarmThreshold = Nothing
End Sub

```

Log, AlarmThresholdCmdTarget Property

Syntax Log = _Boolean

Description Questa proprietà permette di determinare se attivare o meno la funzione di registrazione nel Log Storico dell'allarme o messaggio al momento degli eventi relativi (ON, OFF, ACK, RESET). Tali registrazioni saranno contenute nella tabella "Alarms" del Database di Log Storico creato da Movicon nel formato specificato o nel file Alarms.dat a seconda che si utilizzi l'ODBC o l'IMDB.



Utilizzando la connessione ODBC, il file di Log Storico viene creato con le impostazioni di default da Movicon, ma all'occorrenza è possibile personalizzare il file di Log Storico, ovvero è possibile creare un collegamento ODBC personalizzato e definire un diverso nome per la tabella. Queste funzionalità possono essere eseguite dalle "Impostazioni Log Storico" del Porgetto.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print Log  
End Sub
```

Name, AlarmThresholdCmdTarget Property

Syntax Name

Description Questa proprietà restituisce il nome della soglia. Questa proprietà è di sola lettura.

Parameter	Description
None	None

Result String

Example:

```
Public Sub AlarmLoading()  
    Debug.Print Name  
End Sub
```

PlaysoundContinuously, AlarmThresholdCmdTarget Property

Syntax PlaysoundContinuously = _Boolean

Description Questa proprietà permette di definire il comportamento del file di suono eventualmente associato all'allarme. Attivando la proprietà il file sarà eseguito continuamente fino alla tacitazione dell'allarme. In caso contrario, il file sarà eseguito solo una volta alla comparsa dell'allarme.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print PlaysoundContinuously  
End Sub
```


Print, AlarmThresholdCmdTarget Property

Syntax Print = _Boolean

Description Questa proprietà permette di determinare se attivare o meno la funzione di stampa dell'allarme o messaggio al momento degli eventi relativi (ON, OFF, ACK, RESET). La funzionalità di stampa dovrà essere stata attivata e configurata nelle impostazioni di "Stampa Log Storico".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print Print  
End Sub
```

ReadAccessLevel, AlarmThresholdCmdTarget Property

Syntax ReadAccessLevel = _Long

Description Questa proprietà imposta o restituisce la maschera di Livelli di Accesso necessaria per poter visualizzare l'allarme nella "Finestra Allarmi". Se la maschera di Livelli di Accesso dell'utente loggato in quel momento non ha nessuna corrispondenza con quella impostata sul controllo, l'utente non potrà vedere gli allarmi. I livelli "0000" e "FFFF" renderanno l'oggetto accessibile in lettura a qualsiasi utente. Naturalmente le operazioni di storicizzazione degli allarmi verranno eseguite comunque indipendentemente dai diritti di accesso dell'utente loggato al momento. Per ulteriori chiarimenti sui "Livelli di Accesso" far riferimento al paragrafo "Livelli Utente e Livelli di Accesso".

Parameter	Description
None	None

Result Long

Example:

```
Public Sub AlarmLoading()  
    Debug.Print ReadAccessLevel  
End Sub
```

Recipient, AlarmThresholdCmdTarget Property

Syntax Recipient = _String

Description Tramite questa proprietà è possibile reperire o impostare l'utente o il gruppo di utenti destinatari verso i quali deve essere inviato il messaggio, l'SMS, l'E-mail, ecc.

Parameter	Description
None	None

Result String

Example:

```
Public Sub AlarmLoading()  
    Debug.Print Recipient  
End Sub
```

RepeatSpeechEverySec, AlarmThresholdCmdTarget Property

Syntax RepeatSpeechEverySec = _Long

Description Questa proprietà consente di impostare o visualizzare il tempo espresso in secondi dopo il quale lo Speech dell'allarme dovrà essere ripetuto. Impostando questa proprietà al valore "0" lo speech dell'allarme viene ripetuto una sola volta. Il tempo da impostare deve tenere conto anche del tempo necessario per eseguire lo speech dell'allarme.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub AlarmLoading()  
    Debug.Print RepeatSpeechEverySec  
End Sub
```

SecDelay, AlarmThresholdCmdTarget Property

Syntax SecDelay = _Long

Description Questa proprietà imposta o restituisce il tempo di ritardo dell'allarme. E' possibile definire un valore numerico compreso tra 0 e 65535. Tale valore, espresso in secondi, stabilisce il tempo di ritardo all'intervento dell'allarme, creando così un filtro sulla soglia.
Il valore di default è zero (nessun ritardo).

Parameter	Description
None	None

Result Long

Example:

```
Public Sub AlarmLoading()  
    Debug.Print SecDelay  
End Sub
```

SendFaxEnabledACK, AlarmThresholdCmdTarget Property

Syntax SendFaxEnabledACK = _Boolean

Description Questa proprietà attiva o disattiva la gestione di invio di FAX. Questa gestione richiede che siano abilitate e configurate le apposite funzioni per l'invio di FAX e che sia configurato l'eventuale modem.
Il messaggio verrà inviato nel momento in cui sarà eseguito il riconoscimento (ACK) dell'allarme.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print SendFaxEnabledACK  
End Sub
```

SendFaxEnabledOFF, AlarmThresholdCmdTarget Property

Syntax SendFaxEnabledOFF = _Boolean

Description Questa proprietà attiva o disattiva la gestione di invio di FAX. Questa gestione richiede che siano abilitate e configurate le apposite funzioni per l'invio di FAX e che sia configurato l'eventuale modem.
Il messaggio verrà inviato al momento della scomparsa dell'allarme, indipendentemente dallo stato di riconoscimento o di reset.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print SendFaxEnabledOFF  
End Sub
```

SendFaxEnabledON, AlarmThresholdCmdTarget Property

Syntax SendFAXEnabledON= _Boolean

Description Questa proprietà attiva o disattiva la gestione di invio di FAX. Questa gestione richiede che siano abilitate e configurate le apposite funzioni per l'invio di FAX e che sia configurato l'eventuale modem.
Il messaggio verrà inviato al momento dell'intervento (ON) dell'allarme.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print SendFAXEnabledON  
End Sub
```

SendFaxEnabledRESET, AlarmThresholdCmdTarget Property

Syntax SendFaxEnabledRESET = _Boolean

Description Questa proprietà attiva o disattiva la gestione di invio di FAX. Questa gestione richiede che siano abilitate e configurate le apposite funzioni per l'invio di FAX e che sia configurato l'eventuale modem.
Il messaggio verrà inviato nel momento in cui sarà eseguito il reset (RESET) dell'allarme.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print SendFaxEnabledRESET  
End Sub
```

SendMailEnabledACK, AlarmThresholdCmdTarget Property

Syntax SendMailEnabledACK = _Boolean

Description Questa proprietà attiva o disattiva la gestione di invio di messaggi E-mail. Questa gestione richiede che sia installato lo strumento gestore della posta E-mail sia Client che Server, oltre al necessario modem e collegamento Internet.

Il messaggio verrà inviato nel momento in cui sarà eseguito il riconoscimento (ACK) dell'allarme.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print SendMailEnabledACK  
End Sub
```

SendMailEnabledOFF, AlarmThresholdCmdTarget Property

Syntax SendMailEnabledOFF = _Boolean

Description Questa proprietà attiva o disattiva la gestione di invio di messaggi E-mail. Questa gestione richiede che sia installato lo strumento gestore della posta E-mail sia Client che Server, oltre al necessario modem e collegamento Internet.
Il messaggio verrà inviato al momento della scomparsa dell'allarme, indipendentemente dallo stato di riconoscimento o di reset.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print SendMailEnabledOFF  
End Sub
```

SendMailEnabledON, AlarmThresholdCmdTarget Property

Syntax SendMailEnabledON = _Boolean

Description Questa proprietà attiva o disattiva la gestione di invio di messaggi E-mail. Questa gestione richiede che sia installato lo strumento gestore della posta E-mail sia Client che Server, oltre al necessario modem e collegamento Internet.
Il messaggio verrà inviato al momento dell'intervento (ON) dell'allarme.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print SendMailEnabledON  
End Sub
```

SendMailEnabledRESET, AlarmThresholdCmdTarget Property

Syntax SendMailEnabledRESET = _Boolean

Description Questa proprietà attiva o disattiva la gestione di invio di messaggi E-mail. Questa gestione richiede che sia installato lo strumento gestore della posta E-mail sia Client che Server, oltre al necessario modem e collegamento Internet. Il messaggio verrà inviato nel momento in cui sarà eseguito il reset (RESET) dell'allarme.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print SendMailEnabledRESET  
End Sub
```

SendSMSEnabledACK, AlarmThresholdCmdTarget Property

Syntax SendSMSEnabledACK = _Boolean

Description Questa proprietà attiva o disattiva la gestione di invio di SMS. Questa gestione richiede che siano abilitate e configurate le apposite funzioni per l'invio degli SMS e che sia configurato l'eventuale modem. Il messaggio verrà inviato nel momento in cui sarà eseguito il riconoscimento (ACK) dell'allarme.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print SendSMSEnabledACK  
End Sub
```

SendSMSEnabledOFF, AlarmThresholdCmdTarget Property

Syntax SendSMSEnabledOFF = _Boolean

Description Questa proprietà attiva o disattiva la gestione di invio di SMS. Questa gestione richiede che siano abilitate e configurate le apposite funzioni per l'invio degli SMS e che sia configurato l'eventuale modem.
Il messaggio verrà inviato al momento della scomparsa dell'allarme, indipendentemente dallo stato di riconoscimento o di reset.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print SendSMSEnabledOFF  
End Sub
```

SendSMSEnabledON, AlarmThresholdCmdTarget Property

Syntax SendSMSEnabledON = _Boolean

Description Questa proprietà attiva o disattiva la gestione di invio di SMS. Questa gestione richiede che siano abilitate e configurate le apposite funzioni per l'invio degli SMS e che sia configurato l'eventuale modem.
Il messaggio verrà inviato al momento dell'intervento (ON) dell'allarme.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print SendSMSEnabledON  
End Sub
```

SendSMSEnabledRESET, AlarmThresholdCmdTarget Property

Syntax SendVoiceEnabledRESET = _Boolean

Description Questa proprietà attiva o disattiva la gestione di invio di SMS. Questa gestione richiede che siano abilitate e configurate le apposite funzioni per l'invio degli SMS e che sia configurato l'eventuale modem.

Il messaggio verrà inviato nel momento in cui sarà eseguito il reset (RESET) dell'allarme.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print SendVoiceEnabledRESET  
End Sub
```

SendVoiceEnabledACK, AlarmThresholdCmdTarget Property

Syntax SendVoiceEnabledACK = _Boolean

Description Questa proprietà attiva o disattiva la gestione di invio di messaggi Vocali. Questa gestione richiede che siano abilitate e configurate le apposite funzioni per l'invio dei messaggi vocali (TAPI), che siano installati i fonemi desiderati e che sia configurato l'eventuale modem.
Il messaggio verrà inviato nel momento in cui sarà eseguito il riconoscimento (ACK) dell'allarme.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print SendVoiceEnabledACK  
End Sub
```

SendVoiceEnabledOFF, AlarmThresholdCmdTarget Property

Syntax SendVoiceEnabledOFF = _Boolean

Description Questa proprietà attiva o disattiva la gestione di invio di messaggi Vocali. Questa gestione richiede che siano abilitate e configurate le apposite funzioni per l'invio dei messaggi vocali (TAPI), che siano installati i fonemi desiderati e che sia configurato l'eventuale modem con funzioni vocali.
Il messaggio verrà inviato al momento della scomparsa dell'allarme, indipendentemente dallo stato di riconoscimento o di reset.

Parameter	Description
-----------	-------------

None	None
------	------

Result Boolean

Example:

```
Public Sub AlarmLoading()
    Debug.Print SendVoiceEnabledOFF
End Sub
```

SendVoiceEnabledON, AlarmThresholdCmdTarget Property

Syntax SendVoiceEnabledON = _Boolean

Description Questa proprietà attiva o disattiva la gestione di invio di messaggi Vocali. Questa gestione richiede che siano abilitate e configurate le apposite funzioni per l'invio dei messaggi vocali (TAPI), che siano installati i fonemi desiderati e che sia configurato l'eventuale modem.
Il messaggio verrà inviato al momento dell'intervento (ON) dell'allarme.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()
    Debug.Print SendVoiceEnabledON
End Sub
```

SendVoiceEnabledRESET, AlarmThresholdCmdTarget Property

Syntax SendSMSEnabledRESET = _Boolean

Description Questa proprietà attiva o disattiva la gestione di invio di messaggi Vocali. Questa gestione richiede che siano abilitate e configurate le apposite funzioni per l'invio dei messaggi vocali (TAPI), che siano installati i fonemi desiderati e che sia configurato l'eventuale modem.
Il messaggio verrà inviato nel momento in cui sarà eseguito il reset (RESET) dell'allarme.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()
```

```
Debug.Print SendSMSEnabledRESET  
End Sub
```

Severity, AlarmThresholdCmdTarget Property

Syntax Severity = _Long

Description Questa proprietà imposta o restituisce la priorità desiderata per l'allarme. La priorità verrà visualizzata e registrata nell'apposito campo riservato all'allarme. E' possibile assegnare un numero compreso tra 0 e 65535. La finestra allarmi ed il Log Storico registreranno il numero di priorità assegnato.

Parameter	Description
None	None

Result Long

Example:
Public Sub AlarmLoading()
Debug.Print **Severity**
End Sub

SpeechEnabled, AlarmThresholdCmdTarget Property

Syntax SpeechEnabled = _Boolean

Description Questa proprietà permette di abilitare le funzioni di speeching per la sintesi vocale della stringa di testo che costituisce l'allarme. Il testo dell'allarme sarà quindi "pronunciato" da Movicon alla sua comparsa.



Affinchè la funzionalità sia attivata, occorre in precedenza procedere alla installazione del motore speeching API di Microsoft (o di un altro produttore), che normalmente non viene installato automaticamente in Windows. La cartella "Speech" del DVD di Movicon contiene l'autoinstallatore del motore di speeching di Microsoft. Oltre alla installazione del motore di speech, occorre installare i fonemi per la pronuncia del testo. I fonemi permettono di "pronunciare" il testo interpretandolo secondo la pronuncia della lingua desiderata. I fonemi dipendono dalla lingua utilizzata e normalmente sono forniti a parte. Sarà cura dell'utente procurarsi ed installare i fonemi desiderati.

Parameter	Description
None	None

Result Boolean

Example:
Public Sub AlarmLoading()
Debug.Print **SpeechEnabled**
End Sub

SpeechEnableVariable, AlarmThresholdCmdTarget Property

Syntax SpeechEnableVariable = String

Description Questa proprietà imposta o restituisce il nome della variabile che consente di gestire l'abilitazione dello Speech (proprietà di stile "Var. Abilitazione Speech" delle soglie di allarme).

Parameter	Description
None	None

Result String

Example:

```
Public Sub AlarmLoading()  
    Debug.Print SpeechEnableVariable  
End Sub
```

Status, AlarmThresholdCmdTarget Property

Syntax Status = _Integer

Description Questa proprietà imposta o restituisce lo stato dell'allarme.
I valori possibili sono:

0 = Allarme non presente
1 = Allarme non attivo e non riconosciuto
2 = Allarme non attivo ma riconosciuto
3 = Allarme attivo e non riconosciuto
4 = Allarme attivo e riconosciuto

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub OnResetAlarm(bRet As Boolean)  
    Debug.Print Status  
End Sub
```

StatusVariable, AlarmThresholdCmdTarget Property

Syntax StatusVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile di stato-comando della soglia di allarme. Impostando una variabile di tipo bit sarà possibile soltanto comandare un riconoscimento dell'allarme. Impostando una variabile di tipo byte (o con un numero di bit superiore) sarà possibile anche avere lo stato dell'allarme o comandare il reset.

Parameter	Description
None	None

Result String

Example:

```
Public Sub AlarmLoading()  
    MsgBox "Status Variable Name is: " & cstr(StatusVariable), vbInformation,  
    GetProjectTitle  
End Sub
```

SupportAcknowledge, AlarmThresholdCmdTarget Property

Syntax SupportAcknowledge = _Boolean

Description Tramite questa proprietà è possibile determinare se attivare o meno la funzione di tacitazione (ACK) dell'allarme.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print Attachment  
End Sub
```

SupportReset, AlarmThresholdCmdTarget Property

Syntax SupportResetWithConditionOn = _Boolean

Description Tramite questa proprietà è possibile determinare se attivare o meno la funzione di reset (RESET) dell'allarme con allarme On.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()
    Debug.Print SupportResetWithConditionOn
End Sub
```

TextColor, AlarmThresholdCmdTarget Property

Syntax TextColor = _Long

Description Questa proprietà imposta o restituisce il colore del testo relativo all'allarme o messaggio visualizzato nella apposita finestra.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub AlarmLoading()
    Debug.Print TextColor
End Sub
```

Text, AlarmThresholdCmdTarget Property

Syntax Text = _String

Description Questa proprietà imposta o restituisce l'ID della stringa o il testo per la soglia di allarme referenziato.

Parameter	Description
None	None

Result String

Example:

```
Public Sub AlarmLoading()
    Text = "STR00001"
    Debug.print Text 'STR00001
End Sub
```

SupportResetWithConditionOn, AlarmThresholdCmdTarget Property

Syntax SupportReset = _Boolean

Description Tramite questa proprietà è possibile determinare se attivare o meno la funzione di reset (RESET) dell'allarme.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()  
    Debug.Print SupportReset  
End Sub
```

Threshold, AlarmThresholdCmdTarget Property

Syntax Threshold = _Double

Description Questa proprietà imposta o restituisce il valore di soglia da raggiungere affinché l'allarme relativo sia attivato effettivamente.

Parameter	Description
None	None

Result Double

Example:

```
Public Sub AlarmLoading()  
    Debug.Print "Threshold is " & Threshold  
End Sub
```

ThresholdLow, AlarmThresholdCmdTarget Property

Syntax Threshold = _Double

Description Questa proprietà imposta o restituisce il valore di soglia minima per l'attivazione dell'allarme. Questo valore viene considerato soltanto se come "Condizione di Attivazione" si è selezionato "Compreso tra (Between)". Negli altri casi il valore non avrà nessun effetto.

Parameter	Description
-----------	-------------

None	None
------	------

Result Double

Example:

```
Public Sub AlarmLoading()
    MsgBox "Threshold is: " & cstr(ThresholdLow), vbInformation, GetProjectTitle
End Sub
```

TotalTimeOn, AlarmThresholdCmdTarget Property

Syntax TotalTimeOn

Description Questa proprietà restituisce un valore di tipo double che indica il tempo totale espresso in secondi in cui l'allarme è rimasto attivo nello stato di ON.

Parameter	Description
None	None

Result Double

Example:

```
Public Sub Click()
    Dim objAlarm As AlarmCmdTarget
    Dim objAlarmThreshold As AlarmThresholdCmdTarget
    Set objAlarm = GetAlarm("Alarm01")
    Set objAlarmThreshold = objAlarm.GetAlarmThreshold("Threshold01")
    MsgBox " TotalTimeOn = " &
    CStr(objAlarmThreshold.TotalTimeOn), vbInformation, GetProjectTitle
    Set objAlarm = Nothing
    Set objAlarmThreshold = Nothing
End Sub
```

VariableSeverity, AlarmThresholdCmdTarget Property

Syntax VariableSeverity = _String

Description Questa proprietà restituisce o imposta il nome della Variabile che definisce il valore della Priorità di una soglia dall'allarme.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim objAlarm As AlarmCmdTarget
    Dim objAlarmThreshold As AlarmThresholdCmdTarget

    Set objAlarm = GetAlarm("AlarmNoReset")
    Set objAlarmThreshold = objAlarm.GetAlarmThreshold("NoReset")

    MsgBox "          VariableSeverity          =          "          &
    CStr(objAlarmThreshold.VariableSeverity ),vbInformation,GetProjectTitle

    Set objAlarm = Nothing
    Set objAlarmThreshold = Nothing
End Sub

```

VariableThreshold, AlarmThresholdCmdTarget Property

Syntax VariableThreshold = _String

Description Questa proprietà imposta o restituisce il nome della variabile il cui valore viene utilizzato al posto della proprietà Threshold ('Valore'). In questo modo è possibile rendere la soglia di tipo dinamico. Inserendo una stringa nulla, Movicon prende in considerazione solo la soglia fissa.

Parameter	Description
None	None

Result String

Example:

```

Public Sub AlarmLoading()
    MsgBox "Variable Threshold Name is: " & cstr(VariableThreshold),
    vbInformation, GetProjectTitle
End Sub

```

VariableThresholdLow, AlarmThresholdCmdTarget Property

Syntax VariableThresholdLow = _String

Description Questa proprietà imposta o restituisce il nome della variabile il cui valore viene utilizzato al posto della proprietà ThresholdLow ('Valore Minimo di Attivazione'). In questo modo è possibile rendere la soglia di tipo dinamico. Inserendo una stringa nulla, Movicon prende in considerazione solo la soglia fissa.

Parameter	Description
None	None

Result Double

Example:

```
Public Sub AlarmLoading()
    MsgBox "Variable Threshold Low Name is: " & cstr(VariableThresholdLow),
vbInformation, GetProjectTitle
End Sub
```

WriteAccessLevel, AlarmThresholdCmdTarget Property

Syntax WriteAccessLevel = _Long

Description Questa proprietà imposta o restituisce la maschera di Livelli di Accesso necessaria per poter eseguire ad esempio il riconoscimento e il reset dell'allarme. Se la maschera di Livelli di Accesso dell'utente loggato in quel momento non ha nessuna corrispondenza con quella impostata sul controllo, l'utente non potrà eseguire le operazioni di comando associate all'allarme. Il livello "FFFF" renderanno l'oggetto accessibile in scrittura a qualsiasi utente. Per ulteriori chiarimenti sui "Livelli di Accesso" far riferimento al paragrafo "Livelli Utente e Livelli di Accesso".

Parameter	Description
None	None

Result Long

Example:

```
Public Sub AlarmLoading()
    Debug.Print WriteAccessLevel
End Sub
```

1.14.3. AlarmWndCmdTarget

Events

OnAckAll, AlarmWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di eseguire il riconoscimento di tutti gli allarmi presenti nella finestra di visualizzazione.



Questa funzione non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
bRet As Boolean	Abilitazione al riconoscimento degli allarmi. Se impostato a False l'evento non eseguirà il riconoscimento allarmi.

OnAckSel, AlarmWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di eseguire il riconoscimento dell'allarme selezionato nella finestra di visualizzazione.



Questa funzione non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
bRet As Boolean	Abilitazione al riconoscimento dell'allarme selezionato. Se impostato a False l'evento non eseguirà il riconoscimento dell'allarme.

OnCommentSel, AlarmWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di inserire un commento per l'allarme selezionato nella finestra di visualizzazione.



Questa funzione non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
bRet As Boolean	Abilitazione all'inserimento del commento. Se impostato a False l'evento annullerà la richiesta di commento.

OnGetHistory, AlarmWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di eseguire il caricamento dello storico di un allarme tramite il pulsante "Visualizza Storico".

Parameter	Description
bRet As Boolean	Abilitazione al recupero dei dati. Se impostato a False l'evento non restituirà la cronologia dell'allarme.

OnHelp, AlarmWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di visualizzare il file di help associato all'allarme selezionato.

Parameter	Description
bRet As Boolean	Abilitazione all'apertura del file di help. Se impostato a False l'evento non aprirà il file di help.

OnInsertOrUpdateAlarm, AlarmWndCmdTarget Event

Description Evento notificato ogni volta che viene inserito un nuovo allarme o viene modificato lo stato di un allarme presente nella finestra di visualizzazione.

Parameter	Description
AlarmObject As Object	Oggetto di tipo 'AlarmThresholdCmdTarget' dell'Allarme appena inserito nella Finestra Allarmi o del quale è stato modificato lo stato
bRet As Boolean	Abilitazione all'inserimento dell'allarme nella finestra. Se impostato a False l'evento non notificherà alla finestra l'evento di inserimento di un nuovo allarme o l'evento di cambiamento di stato di un allarme già presente.

OnOPCAEvent, AlarmWndCmdTarget Event

Description Evento notificato ogni volta che viene notificato un evento allarme via OPC.



Questo evento non è supportato in Windows CE.

Parameter	Description
zSource As String	Sorgente dell'evento
dDate As Date	Data e ora
szEvent As String	Tipo di evento
dwSeverity As Long	Livello di gravità
bRet As Boolean	Abilitazione alla notifica dell'evento OPC nella finestra. Se impostato a False l'evento OPC non verrà notificato alla finestra allarmi.

OnResetAll, AlarmWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di eseguire il reset di tutti gli allarmi presenti nella finestra di visualizzazione.



Questa funzione non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
bRet As Boolean	Abilitazione al reset degli allarmi. Se impostato a False l'evento non eseguirà il reset allarmi.

OnResetSelSel, AlarmWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di eseguire il reset dell'allarme selezionato nella finestra di visualizzazione.



Questa funzione non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
bRet As Boolean	Abilitazione al reset dell'allarme selezionato. Se impostato a False l'evento non eseguirà il reset dell'allarme.

OnToggleSound, AlarmWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di tacitare il suono associato agli allarmi attivi.

Parameter	Description
bRet As Boolean	Abilitazione ad eseguire il comando di toggle del suono. Se impostato a False l'evento non eseguirà il comando

Functions

AckSelectedAlarms, AlarmWndCmdTarget Function

Syntax AckSelectedAlarms()

Description Questa funzione permette il riconoscimento degli allarmi selezionati.



Questa funzione non è gestita dall'oggetto "Banner Allarmi". Restituisce sempre False.

Parameter	Description
None	None

Result Boolean

Example:
Sub Click()
 AckSelectedAlarms
End Sub

EditLayout, AlarmWndCmdTarget Function

Syntax EditLayout()

Description Questa funzione apre la finestra di configurazione dei campi da visualizzare nella finestra di visualizzazione degli allarmi.



La funzione verrà eseguita soltanto se nell'oggetto Finestra è stata abilitata la proprietà "Mostra Finestra di Controllo". In caso contrario la "Finestra Scelta Campo" non verrà aperta e la funzione ritornerà il valore "False"



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:
Sub Click()
 EditLayout
End Sub

EditCopy, AlarmWndCmdTarget Function

Syntax EditCopy()

Description Questa proprietà esegue una copia negli appunti del contenuto dell'allarme selezionato.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:
Dim objAlarmWnd As AlarmWndCmdTarget
Public Sub Click()
 Debug.Print objAlarmWnd.**EditCopy**
End Sub
Public Sub SymbolLoading()
 Set objAlarmWnd =
 GetSynopticObject.GetSubObject("ALRWindow").GetObjectInterface
End Sub

GetNumTotalAlarms, AlarmWndCmdTarget Function

Syntax GetNumTotalAlarms()

Description Questa funzione restituisce il numero degli allarmi presenti nella finestra di visualizzazione.



Questa funzione non è gestita dall'oggetto "Banner Allarmi". Restituisce sempre 0.

Parameter	Description
None	None

Result Long

Example:

```
Sub Click()  
    Dim IResult As Long  
    IResult = GetNumTotalAlarms  
    Debug.Print IResult  
End Sub
```

GetSelectedAlarm, AlarmWndCmdTarget Function

Syntax GetSelectedAlarm()

Description Questa funzione restituisce l'allarme selezionato.



Questa funzione non è gestita dall'oggetto "Banner Allarmi". Restituisce sempre un'oggetto nothing.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo AlarmThresholdCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Sub Click()  
    Dim objAlarm As AlarmThresholdCmdTarget  
    Set objAlarm = GetSelectedAlarm  
    Debug.Print objAlarm.Condition  
    Set objAlarm = Nothing  
End Sub
```

GetSelHistory, AlarmWndCmdTarget Function

Syntax GetSelHistory()

Description Questo metodo esegue lo stesso comando del relativo pulsante "Visualizza Storico", ovvero recupera le informazioni storiche per l'allarme selezionato. Se il valore di ritorno della funzione è False significa che si è verificato un errore nell'operazione.



Questa funzione non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim objWnd As AlarmWndCmdTarget
    Set objWnd = GetSynopticObject.GetSubObject("AlarmWnd").GetObjectInterface
    If objWnd Is Nothing Then Exit Sub
    objWnd.GetSelHistory
    Set objWnd = Nothing
End Sub
```

LoadExtSettings, AlarmWndCmdTarget Function

Syntax LoadExtSettings

Description Questa funzione permette di caricare la configurazione dell'oggetto dal file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".XML".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim objSymbol As AlarmWndCmdTarget
    Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
    If objSymbol Is Nothing Then Exit Sub
    objSymbol.ExtSettingsFile = "test.xml"
    objSymbol.LoadExtSettings
    Set objSymbol = Nothing
End Sub
```

RecalcLayout, AlarmWndCmdTarget Function

Syntax RecalcLayout()

Description Questa funzione esegue un'aggiornamento grafico dell'oggetto. E' necessario eseguire tale funzione dopo che e' stata modificata una proprietà il cui impatto grafico sia particolarmente consistente, come ad esempio la modifica della dimensione di una colonna.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()  
    Dim bResult As Boolean  
    bResult = RecalcLayout  
    Debug.Print bResult  
End Sub
```

ResetSelectedAlarms, AlarmWndCmdTarget Function

Syntax ResetSelectedAlarms()

Description Questa funzione permette il reset degli allarmi selezionati.



Questa funzione non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()  
    ResetSelectedAlarms  
End Sub
```

Refresh, AlarmWndCmdTarget Function

Syntax Refresh()

Description Questa funzione esegue un'aggiornamento dei dati nell'oggetto.

Parameter	Description
None	None

Result Boolean

Example:
Sub Click()
 Refresh
End Sub

SelectAll, AlarmWndCmdTarget Function

Syntax SelectAll()

Description Questa funzione permette di selezionare tutti gli allarmi visualizzati nella finestra di visualizzazione degli allarmi. Restituisce il numero di allarmi selezionati.



Questa funzione non è gestita dall'oggetto "Banner Allarmi". Restituisce sempre 0.

Parameter	Description
None	None

Result Long

Example:
Sub Click()
 SelectAll
End Sub

SaveExtSettings, AlarmWndCmdTarget Function

Syntax SaveExtSettings

Description Questa funzione permette di salvare la configurazione dell'oggetto nel file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Boolean

Example:
Public Sub Click()
 Dim objSymbol As AlarmWndCmdTarget
 Set objSymbol =
 GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
 If objSymbol Is Nothing Then Exit Sub
 objSymbol.ExtSettingsFile = "test.sxml"
 objSymbol.SaveExtSettings
 Set objSymbol = Nothing
End Sub

Properties

AckAllBtnText, AlarmWndCmdTarget Property

Syntax AckAllBtnText = _String

Description Questa proprietà restituisce il testo che dovrà apparire sul "Pulsante Riconoscimento Tutti". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Sub Click()  
    Debug.Print AckAllBtnText  
End Sub
```

AckSelBtnText, AlarmWndCmdTarget Property

Syntax AckSelBtnText = _String

Description Questa proprietà restituisce il testo che dovrà apparire sul "Pulsante Riconosci Sel". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Sub Click()  
    Debug.Print AckSelBtnText  
End Sub
```

AlarmFilter, AlarmWndCmdTarget Property

Syntax AlarmFilter = _String

Description Questa proprietà permette di inserire un filtro per la visualizzazione degli allarmi in base al loro testo. Il filtro viene applicato alla colonna "Descrizione Allarme" e può contenere uno o più di un carattere speciale "*" (Es. *Turbina*).

Parameter	Description
None	None

Result String

Example:
Sub Click()
 Debug.Print **AlarmFilter**
End Sub

AlarmFilterMask, AlarmWndCmdTarget Property

Syntax AlarmFilterMask = _Integer

Description Questa proprietà permette di inserire un filtro in base allo stato degli allarmi da visualizzare nella finestra.

Il filtro può assumere i seguenti valori:

1 = Allarme ON
2 = Allarme ACK
4 = Allarme OFF ACK
8 = Allarme OFF
16 = Nessun Allarme

Per maggiori informazione vedere Inserimento degli Allarmi.

Parameter	Description
None	None

Result Integer

Example:
Sub Click()
 Debug.Print **AlarmFilter**
End Sub

AlarmFilterSeverity, AlarmWndCmdTarget Property

Syntax AlarmFilterSeverity = _Long

Description Questa proprietà imposta o restituisce il valore di priorità per il quale verrà eseguito il filtro sulla Finestra Allarmi.

Parameter	Description
-----------	-------------

None	None
------	------

Result Long

Example:

```
Public Sub Click()
    Dim objWnd As AlarmWndCmdTarget

    Set objWnd = GetSynopticObject.GetSubObject("AlarmWnd").GetObjectInterface
    If objWnd Is Nothing Then Exit Sub
    MsgBox " AlarmFilterSeverity = " &
    cstr(objWnd.AlarmFilterSeverity),vbInformation,GetProjectTitle
    Set objWnd = Nothing
End Sub
```

AlarmFilterSeverityCondition, AlarmWndCmdTarget Property

Syntax AlarmFilterSeverityCondition = _Integer

Description Questa proprietà imposta o restituisce il tipo di condizione di confronto per eseguire il filtro sulla priorità nella Finestra Allarmi. I valori possibili sono:

0 = Uguale
1 = Minore-uguale
2 = Maggiore-uguale

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    Dim objWnd As AlarmWndCmdTarget
    Set objWnd = GetSynopticObject.GetSubObject("AlarmWnd").GetObjectInterface
    If objWnd Is Nothing Then Exit Sub
    MsgBox " AlarmFilterSeverityCondition = " &
    cstr(objWnd.AlarmFilterSeverityCondition),vbInformation,GetProjectTitle
    Set objWnd = Nothing
End Sub
```

AreaFilter, AlarmWndCmdTarget Property

Syntax AreaFilter = _String

Description Questa proprietà permette di inserire un filtro per visualizzare solo gli allarmi che appartengono ad una certa area. Il filtro può contenere uno o più di un carattere speciale "*" (Es. *Area*).

Parameter	Description
None	None

Result String

Example:

```
Sub Click()  
    Debug.Print AreaFilter  
End Sub
```

AutoLayout, AlarmWndCmdTarget Property

Syntax AutoLayout = _Boolean

Description Abilitando questa proprietà il layout verrà impostato in modo automatico. Questo significa che le colonne verranno automaticamente ridimensionate in modo che risultino visibili tutte quante nell'area della Finestra Allarmi. Disabilitando invece questa proprietà all'apertura della finestra le colonne avranno la dimensione definita in programmazione, con la possibilità che le ultime sulla destra non siano visibili se non attivando la barra di scorrimento orizzontale.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()  
    Debug.Print AutoLayout  
End Sub
```

Autoscroll, AlarmWndCmdTarget Property

Syntax Autoscroll = _Boolean

Description Abilitando questa proprietà lo scroll degli allarmi attivi avverrà in modo automatico. Impostando la proprietà al valore "True" lo scroll automatico verrà abilitato, impostandola a "False" verrà disabilitato.



Questa proprietà non è gestita dall'oggetto "Finestra Allarmi". Restituisce sempre False.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objAlarmBanner As AlarmWndCmdTarget  
    Set objAlarmBanner =   
    GetSynopticObject.GetSubObject("objAlarmBanner").GetObjectInterface
```

```

objAlarmBanner.Autoscroll = Not objAlarmBanner.Autoscroll
Set objAlarmBanner = Nothing
End Sub

```

BlinkTime, AlarmWndCmdTarget Property

Syntax BlinkTime = _Long

Description Questa proprietà rappresenta il tempo di lampeggio per gli allarmi non ancora riconosciuti. Il valore è espresso in millisecondi.

Parameter	Description
None	None

Result Long

Example:
Sub Click()
 Debug.Print **BlinkTime**
End Sub

ButtonPos, AlarmWndCmdTarget Property

Syntax ButtonPos

Description Questa impostazione restituisce la posizione in cui dovranno apparire i pulsanti nella Finestra Allarmi.

Le possibilità sono:
0 = sinistra
1 = sopra
2 = destra
3 = sotto



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:
Sub Click()
 ButtonPos = 2
 Debug.Print **ButtonPos**
End Sub

ButtonSize, AlarmWndCmdTarget Property

Syntax ButtonSize

Description Questa impostazione restituisce la dimensione dei pulsanti che verranno poi visualizzati nella Finestra Allarmi.

Le possibilità sono:

0 = piccolo

1 = medio

2 = grande



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    ButtonSize = 2  
    Debug.Print ButtonSize  
End Sub
```

Clickable, AlarmWndCmdTarget Property

Syntax Clickable = _Boolean

Description Questa proprietà permette di definire se l'operatore potrà interagire con la finestra di visualizzazione degli allarmi. Disabilitando questa proprietà il controllo non potrà infatti essere gestito nè con il mouse nè con la tastiera.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()  
    Debug.Print Clickable  
End Sub
```

ExtSettingsFile, AlarmWndCmdTarget Property

Syntax ExtSettingsFile = _String

Description Questa proprietà imposta o restituisce il file di configurazione esterna per l'oggetto referenziato. Il file può essere specificato anche in fase di design nella proprietà "File di Configurazione" dell'oggetto. L'estensione prevista per il file è ".XML".

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim objSymbol As AlarmWndCmdTarget  
    Set objSymbol =  
        GetSynopticObject.GetSubObject("TestObject").GetObjectInterface  
    If objSymbol Is Nothing Then Exit Sub  
    objSymbol.ExtSettingsFile = "test.xml"  
    objSymbol.SaveExtSettings  
    Set objSymbol= Nothing  
End Sub
```

FormatDateTime, AlarmWndCmdTarget Property

Syntax FormatDateTime = _String

Description Questa proprietà permette di inserire il formato della data e dell'ora con cui verranno visualizzati gli orari nelle colonne "Tempo On", "Tempo Ack", "Tempo Off" e "Tempo Reset". Tutti i codici di formattazione utilizzabili in questa proprietà sono elencati nella sezione Proprietà Stile dei Disegni e Controlli.

Parameter	Description
None	None

Result String

Example:

```
Sub Click()  
    Debug.Print FormatDateTime  
End Sub
```

FormatDuration, AlarmWndCmdTarget Property

Syntax FormatDuration = _String

Description Questa proprietà permette di inserire il formato della durata riportato nella colonna "Durata" della Finestra Allarmi. Tutti i codici di formattazione utilizzabili in questa proprietà sono elencati nella sezione Proprietà Stile dei Disegni e Controlli. Il valore di durata verrà aggiornato soltanto sull'evento di "Allarme Off".



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Sub Click()
    Debug.Print FormatDuration
End Sub
```

GetHistoryBtnTex, AlarmWndCmdTarget Property

Syntax GetHistoryBtnTex = _String

Description Questa proprietà imposta o restituisce il testo per il pulsante "Visualizza Storico". Impostando una stringa nulla verrà visualizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim objWnd As AlarmWndCmdTarget
    Set objWnd = GetSynopticObject.GetSubObject("AlarmWnd").GetObjectInterface
    If objWnd Is Nothing Then Exit Sub
    MsgBox " GetHistoryBtnTex = " & cstr(objWnd.GetHistoryBtnTex
    ),vbInformation,GetProjectTitle
    Set objWnd = Nothing
End Sub
```

GraphicButtons, AlarmWndCmdTarget Property

Syntax GraphicButtons = _Boolean

Description Abilitando questa proprietà i pulsanti della Finestra Allarmi verranno disegnati usando un'icona invece del testo. Il testo opportuno verrà invece visualizzato come tooltip posizionando il mouse sopra ad un pulsante.



Il tooltip non viene gestito nella versione Windows CE.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()
    GraphicButtons = True
    RecalcLayout
End Sub
```

HasSpin, AlarmWndCmdTarget Property

Syntax HasSpin = _Boolean

Description Abilitando questa proprietà verrà visualizzato il pulsante di spin tramite il quale sarà possibile gestire lo scroll degli allarmi attivi nella finestra. Impostando la proprietà al valore "True" il pulsante spin sarà visibile, impostandola a "False" non sarà più visibile.



Questa proprietà non è gestita dall'oggetto "Finestra Allarmi". Restituisce sempre False

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim objAlarmBanner As AlarmWndCmdTarget
    Set objAlarmBanner =
    GetSynopticObject.GetSubObject("objAlarmBanner").GetObjectInterface
    objAlarmBanner.HasSpin = Not objAlarmBanner.HasSpin
    Set objAlarmBanner = Nothing
End Sub
```

HelpBtnText, AlarmWndCmdTarget Property

Syntax HelpBtnText = _String

Description Questa proprietà restituisce il testo che dovrà apparire sul "Pulsante Help". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
-----------	-------------

None	None
------	------

Result String

Example:

```
Sub Click()  
    Debug.Print HelpBtnText  
End Sub
```

HisLogBackColor, AlarmWndCmdTarget Property

Syntax HisLogBackColor = _Long

Description Questa proprietà imposta o restituisce il colore di sfondo dell'area contenente i dati storici dell'allarme.



Questa funzione non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim objWnd As AlarmWndCmdTarget  
    Set objWnd = GetSynopticObject.GetSubObject("AlarmWnd").GetObjectInterface  
    If objWnd Is Nothing Then Exit Sub  
    MsgBox " HisLogBackColor = " & cstr(objWnd.HisLogBackColor  
    ),vbInformation,GetProjectTitle  
    Set objWnd = Nothing  
End Sub
```

HisLogTextColor, AlarmWndCmdTarget Property

Syntax HisLogTextColor = _Long

Description Questa proprietà imposta o restituisce il colore del testo dei dati storici dell'allarme.



Questa funzione non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim objWnd As AlarmWndCmdTarget  
    Set objWnd = GetSynopticObject.GetSubObject("AlarmWnd").GetObjectInterface  
    If objWnd Is Nothing Then Exit Sub  
    MsgBox " HisLogTextColor = " &  
        cstr(objWnd.HisLogTextColor),vbInformation,GetProjectTitle  
    Set objWnd = Nothing  
End Sub
```

HorizontalSpin, AlarmWndCmdTarget Property

Syntax HorizontalSpin = _Boolean

Description Tramite questa proprietà è possibile definire se il pulsante di spin, tramite il quale sarà possibile gestire lo scroll degli allarmi attivi nella finestra, dovrà essere visualizzato in modo orizzontale o verticale. Impostando la proprietà al valore "True" il pulsante spin sarà visualizzato in modo verticale, impostandola a "False" sarà visualizzato in modo orizzontale.



Questa proprietà non è gestita dall'oggetto "Finestra Allarmi". Restituisce sempre False

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objAlarmBanner As AlarmWndCmdTarget  
    Set objAlarmBanner = GetSynopticObject.GetSubObject("objAlarmBanner").GetObjectInterface  
    objAlarmBanner.HorizontalSpin = Not objAlarmBanner.HorizontalSpin  
    Set objAlarmBanner = Nothing  
End Sub
```

IncludeMilliseconds, AlarmWndCmdTarget Property

Syntax IncludeMilliseconds = _Boolean

Description Abilitando questa proprietà verranno inclusi anche i millisecondi al formato "Ora" nelle colonne della finestra che supportano questo tipo di dato.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()  
    IncludeMilliseconds = True  
    Debug.Print IncludeMilliseconds  
End Sub
```

MaxOPCAEEEvents, AlarmWndCmdTarget Property

Syntax MaxOPCAEEEvents = _Long

Description Questa proprietà restituisce il numero massimo di eventi da visualizzare nella finestra allarmi.



Questa funzione non è supportata su Windows CE.(se utilizzata restituisce sempre il valore 0)



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim objAlarmWnd As AlarmWndCmdTarget  
    Set objAlarmWnd = GetSynopticObject.GetSubObject("AlarmWnd").GetObjectInterface  
    If Not objAlarmWnd Is Nothing Then  
        MsgBox "objAlarmWnd's MaxOPCAEEEvents are" & objAlarmWnd.MaxOPCAEEEvents, vbInformation, GetProjectTitle  
    Else  
        MsgBox "objAlarmWnd is nothing", vbInformation, GetProjectTitle  
    End If  
End Sub
```

NetworkBackupServerName, AlarmWndCmdTarget Property

Syntax NetworkBackupServerName = _String

Description Questa proprietà imposta o restituisce il nome dell'eventuale Server di Networking di Backup usato per prelevare gli allarmi da visualizzare sulla finestra quando il Server Primario, ovvero quello impostato nella proprietà "NetowrkServer", è in timeout.

Parameter	Description
None	None

Result String

Example:

```
Dim objAlarmWnd As AlarmWndCmdTarget
Public Sub Click()
    Debug.Print objAlarmWnd.NetworkBackupServerName
End Sub
Public Sub SymbolLoading()
    Set objAlarmWnd
    GetSynopticObject.GetSubObject("AlarmWindow").GetObjectInterface =
End Sub
```

NetworkServer, AlarmWndCmdTarget Property

Syntax NetworkServer = _String

Description Questa proprietà restituisce il nome dell'eventuale Server di Networking dal quale dovranno essere prelevati gli allarmi da visualizzare sulla finestra.

Parameter	Description
None	None

Result String

Example:

```
Sub Click()
    Debug.Print NetworkServer
End Sub
```

OPCAEServer, AlarmWndCmdTarget Property

Syntax OPCAEServer = _String

Description Questa proprietà restituisce il nome dell'eventuale Server OPC AE dal quale dovranno arrivare le notifiche di allarme da visualizzare sulla finestra.

Parameter	Description
None	None

Result String

Example:

```
Sub Click()
    Debug.Print OPCAEServer
End Sub
```

RefreshTimePoll, AlarmWndCmdTarget Property

Syntax RefreshTimePoll = _Long

Description Questa proprietà restituisce il tempo di interrogazione della rete. Il valore è espresso in millisecondi.

Parameter	Description
None	None

Result Long

Example:
Sub Click()
 Debug.Print **RefreshTimePoll**
End Sub

ResetAllSelBtnText, AlarmWndCmdTarget Property

Syntax ResetAllSelBtnText = _String

Description Questa proprietà restituisce il testo che dovrà apparire sul "Pulsante Reset Tutti". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:
Sub Click()
 Debug.Print **ResetAllSelBtnText**
End Sub

ResetSelBtnText, AlarmWndCmdTarget Property

Syntax ResetSelBtnText = _String

Description Questa proprietà restituisce il testo che dovrà apparire sul "Pulsante Reset Sel". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Sub Click()
    Debug.Print ResetSelBtnText
End Sub
```

ScrollTime, AlarmWndCmdTarget Property

Syntax ScrollTime = _Long

Description Tramite questa proprietà è possibile definire il tempo di scroll degli allarmi, espresso in millisecondi, che verrà utilizzato quando la proprietà "Autoscroll" è abilitata.



Questa proprietà non è gestita dall'oggetto "Finestra Allarmi". Restituisce sempre 0.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
    Dim objAlarmBanner As AlarmWndCmdTarget
    Set objAlarmBanner = objAlarmBanner =
    GetSynopticObject.GetSubObject("objAlarmBanner").GetObjectInterface
    objAlarmBanner.ScrollTime = 1000
    Set objAlarmBanner = Nothing
End Sub
```

ShowAckAlIBtn, AlarmWndCmdTarget Property

Syntax ShowAckAlIBtn = _Boolean

Description Abilitando questa proprietà Movicon rende disponibile il pulsante di comando per l'operazione di riconoscimento di tutti gli allarmi presenti nella Finestra Allarmi, senza obbligo di selezione.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()  
    ShowAckAllBtn = True  
    Debug.Print ShowAckAllBtn  
End Sub
```

ShowAckSelBtn, AlarmWndCmdTarget Property

Syntax ShowAckSelBtn = _Boolean

Description Abilitando questa proprietà Movicon rende disponibile il pulsante di comando per l'operazione di riconoscimento dell'allarme selezionato nella Finestra Allarmi. Per eseguire l'azione in Runtime occorre selezionare uno o più allarmi.



Questa funzione non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()  
    ShowAckSelBtn = True  
    Debug.Print ShowAckSelBtn  
End Sub
```

ShowDateTime, AlarmWndCmdTarget Property

Syntax ShowDateTime = _Boolean

Description Tramite questa proprietà è possibile definire se nel testo dell'allarme visualizzato dovrà comparire anche la data e ora di attivazione dell'evento. Impostando la proprietà al valore "True" oltre al testo dell'allarme verrà anche visualizzata la data e ora di intervento dell'evento, impostandola a "False" sarà visualizzato solo il testo dell'allarme.



Questa proprietà non è gestita dall'oggetto "Finestra Allarmi". Restituisce sempre False

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objAlarmBanner As AlarmWndCmdTarget  
    Set objAlarmBanner = objAlarmBanner =  
    GetSynopticObject.GetSubObject("objAlarmBanner").GetObjectInterface  
    objAlarmBanner.ShowDateTime = Not objAlarmBanner.ShowDateTime  
    Set objAlarmBanner = Nothing  
End Sub
```

ShowGetHistoryBtn, AlarmWndCmdTarget Property

Syntax ShowGetHistoryBtn = _Boolean

Description Questa proprietà mostra o nasconde il pulsante "Visualizza Storico".



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objWnd As AlarmWndCmdTarget  
    Set objWnd = GetSynopticObject.GetSubObject("AlarmWnd").GetObjectInterface  
    If objWnd Is Nothing Then Exit Sub  
    objWnd.ShowGetHistoryBtn = Not objWnd.ShowGetHistoryBtn  
    Set objWnd = Nothing  
End Sub
```

ShowHelpBtn, AlarmWndCmdTarget Property

Syntax ShowHelpBtn = _Boolean

Description Abilitando questa proprietà Movicon rende disponibile il pulsante di comando per l'apertura di un file di help per l'allarme selezionato.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```

Sub Click()
    Dim objAlarmWnd As AlarmWndCmdTarget
    Set objAlarmWnd = GetSynopticObject.GetSubObject("AlarmWnd").GetObjectInterface
    If Not objAlarmWnd Is Nothing Then
        objAlarmWnd.ShowResetAllBtn = Not objAlarmWnd.ShowResetAllBtn
        objAlarmWnd.RecalcLayout
        MsgBox "objAlarmWnd's ShowResetAllBtn is " & objAlarmWnd.ShowResetAllBtn ,vbInformation,GetProjectTitle
    Else
        MsgBox "objAlarmWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub

```

ShowHigherSeverity, AlarmWndCmdTarget Property

Syntax ShowHigherSeverity = _Boolean

Description Tramite questa proprietà è possibile definire se la sequenza con cui gli allarmi verranno visualizzati nel Banner dovrà tener conto della loro priorità. Impostando la proprietà al valore "True" nello scroll degli allarmi verranno visualizzati prima gli allarmi con priorità più alta. In questo caso la sequenza di scroll verrà fatta in base alla priorità e per gli allarmi con la stessa priorità in base all'orario di attivazione, dal più recente al più vecchio. Impostando la proprietà al valore "False" la sequenza di scroll verrà fatta solo in base all'orario di attivazione, dal più recente al più vecchio.



Questa proprietà non è gestita dall'oggetto "Finestra Allarmi". Restituisce sempre False

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub Click()
    Dim objAlarmBanner As AlarmWndCmdTarget
    Set objAlarmBanner = GetSynopticObject.GetSubObject("objAlarmBanner").GetObjectInterface
    objAlarmBanner.ShowHigherSeverity = Not objAlarmBanner.ShowHigherSeverity
    Set objAlarmBanner = Nothing
End Sub

```

ShowResetAllBtn, AlarmWndCmdTarget Property

Syntax ShowResetAllBtn = _Boolean

Description Abilitando questa proprietà Movicon rende disponibile il pulsante di comando per l'operazione di reset di tutti gli allarmi presenti nella Finestra Allarmi, senza obbligo di selezione, purché questi siano stati tacitati con il comando "Riconosci Sel" o "Riconosci Tutti".



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()
    ShowResetAllBtn = True
    Debug.Print ShowResetAllBtn
End Sub
```

ShowResetSelBtn, AlarmWndCmdTarget Property

Syntax ShowResetSelBtn = _Boolean

Description Abilitando questa proprietà Movicon rende disponibile il pulsante di comando per l'operazione di reset dell'allarme selezionato nella Finestra Allarmi. Per eseguire l'azione in Runtime occorre selezionare uno o più allarmi, purchè questi siano stati tacitati con il comando "Riconosci Sel" o "Riconosci Tutti".



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()
    ShowResetSelBtn = True
    Debug.Print ShowResetSelBtn
End Sub
```

ShowSoundOnBtn, AlarmWndCmdTarget Property

Syntax ShowSoundOnBtn = _Boolean

Description Abilitando questa proprietà Movicon rende disponibile il pulsante di comando per l'attivazione o la disattivazione delle segnalazioni acustiche associate alle priorità degli allarmi non riconosciuti.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()
    ShowSoundOnBtn = True
    Debug.Print ShowSoundOnBtn
End Sub
```

SoundOnBtnText, AlarmWndCmdTarget Property

Syntax SoundOnBtnText = _String

Description Questa proprietà restituisce il testo che dovrà apparire sul "Pulsante ON/OFF Suono". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Sub Click()
    Debug.Print SoundOnBtnText
End Sub
```

SpinSize, AlarmWndCmdTarget Property

Syntax SpinSize = _Byte

Description Tramite questa proprietà è possibile definire la dimensione del pulsante di spin dell'oggetto Banner Allarmi.

Le possibilità sono:

0 = piccolo

1 = medio

2 = grande



Questa proprietà non è gestita dall'oggetto "Finestra Allarmi". Restituisce sempre 0

Parameter	Description
-----------	-------------

None	None
------	------

Result Byte

Example:

```
Public Sub Click()
Dim objAlarmBanner As AlarmWndCmdTarget
Set objAlarmBanner = GetSynopticObject.GetSubObject("objAlarmBanner").GetObjectInterface
objAlarmBanner.SpinSize = 2
Set objAlarmBanner = Nothing
End Sub
```

SubItemAck, AlarmWndCmdTarget Property

Syntax SubItemAck = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Tempo Ack". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Sub Click()
Debug.Print SubItemAck
End Sub
```

SubItemAckPos, AlarmWndCmdTarget Property

Syntax SubItemAckPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Tempo ACK" all'interno della Finestra Allarmi. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:
Sub Click()
 Debug.Print **SubItemAckPos**
End Sub

SubItemAckWidth, AlarmWndCmdTarget Property

Syntax SubItemDurationWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione degli allarmi. Se la colonna non è visualizzata, restituisce il valore -1.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:
Sub Click()
 Debug.Print **SubItemDurationWidth**
End Sub

SubItemCondition, AlarmWndCmdTarget Property

Syntax SubItemCondition = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Condizione". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:
Sub Click()
 Debug.Print **SubItemCondition**
End Sub

SubItemConditionPos, AlarmWndCmdTarget Property

Syntax SubItemConditionPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Condizione" all'interno della Finestra Allarmi. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemConditionPos  
End Sub
```

SubItemConditionWidth, AlarmWndCmdTarget Property

Syntax SubItemConditionWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna "Condizione" all'interno della finestra di visualizzazione degli allarmi. Se la colonna non è visualizzata, restituisce il valore -1.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemConditionWidth  
End Sub
```


SubItemDuration, AlarmWndCmdTarget Property

Syntax SubItemDuration = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Durata". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Sub Click()  
    Debug.Print SubItemDuration  
End Sub
```

SubItemDurationPos, AlarmWndCmdTarget Property

Syntax SubItemDurationPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Durata" all'interno della Finestra Allarmi. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemDurationPos  
End Sub
```

SubItemDurationWidth, AlarmWndCmdTarget Property

Syntax SubItemDurationWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione degli allarmi. Se la colonna non è visualizzata, restituisce il valore -1.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemDurationWidth  
End Sub
```

SubItemImage, AlarmWndCmdTarget Property

Syntax SubItemImage = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Immagine". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Sub Click()  
    Debug.Print SubItemImage  
End Sub
```

SubItemImagePos, AlarmWndCmdTarget Property

Syntax SubItemImagePos = _Integer

Description

Questa proprietà imposta o restituisce la posizione della colonna "Immagine" all'interno della Finestra Allarmi. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result

Integer

Example:

```
Sub Click()  
    Debug.Print SubItemImagePos  
End Sub
```

SubItemImageWidth, AlarmWndCmdTarget Property

Syntax

SubItemImageWidth = _Integer

Description

Questa proprietà indica la dimensione in pixel della colonna "Immagine" all'interno della finestra di visualizzazione degli allarmi. Se la colonna non è visualizzata, restituisce il valore -1.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result

Integer

Example:

```
Sub Click()  
    Debug.Print SubItemImageWidth  
End Sub
```

SubItemOff, AlarmWndCmdTarget Property

Syntax

SubItemOff = _String

Description

Permette di definire il testo che dovrà apparire come nome della colonna "Tempo Off". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Sub Click()  
    Debug.Print SubItemOff  
End Sub
```

SubItemOffPos, AlarmWndCmdTarget Property

Syntax SubItemOffPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Tempo OFF" all'interno della Finestra Allarmi. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemOffPos  
End Sub
```

SubItemOffWidth, AlarmWndCmdTarget Property

Syntax SubItemOffWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione degli allarmi. Se la colonna non è visualizzata, restituisce il valore -1.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
-----------	-------------

None	None
------	------

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemOffWidth  
End Sub
```

SubItemOn, AlarmWndCmdTarget Property

Syntax SubItemOn = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Tempo On". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Sub Click()  
    Debug.Print SubItemOn  
End Sub
```

SubItemOnPos, AlarmWndCmdTarget Property

Syntax SubItemOnPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Tempo ON" all'interno della Finestra Allarmi. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:
 Sub Click()
 Debug.Print **SubItemOnPos**
 End Sub

SubItemOnWidth, AlarmWndCmdTarget Property

Syntax SubItemOnWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione degli allarmi. Se la colonna non è visualizzata, restituisce il valore -1.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:
 Sub Click()
 Debug.Print **SubItemOnWidth**
 End Sub

SubItemReset, AlarmWndCmdTarget Property

Syntax SubItemReset = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Tempo Reset". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:
 Sub Click()
 Debug.Print **SubItemReset**
 End Sub

SubItemResetPos, AlarmWndCmdTarget Property

Syntax SubItemResetPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Tempo RESET" all'interno della Finestra Allarmi. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:
Sub Click()
 Debug.Print **SubItemResetPos**
End Sub

SubItemResetWidth, AlarmWndCmdTarget Property

Syntax SubItemResetWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione degli allarmi. Se la colonna non è visualizzata, restituisce il valore -1.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:
Sub Click()
 Debug.Print **SubItemResetWidth**
End Sub

SubItemSeverity, AlarmWndCmdTarget Property

Syntax SubItemSeverity = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Priorità". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Sub Click()  
    Debug.Print SubItemSeverity  
End Sub
```

SubItemSeverityPos, AlarmWndCmdTarget Property

Syntax SubItemSeverityPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Priorità" all'interno della Finestra Allarmi. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemSeverityPos  
End Sub
```


SubItemSeverityWidth, AlarmWndCmdTarget Property

Syntax SubItemSeverityWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione degli allarmi. Se la colonna non è visualizzata, restituisce il valore -1.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:
Sub Click()
 Debug.Print **SubItemSeverityWidth**
End Sub

SubItemStatus, AlarmWndCmdTarget Property

Syntax SubItemStatus = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Stato". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:
Sub Click()
 Debug.Print **SubItemStatus**
End Sub

SubItemStatusPos, AlarmWndCmdTarget Property

Syntax SubItemStatusPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Stato" all'interno della Finestra Allarmi. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemStatusPos  
End Sub
```

SubItemStatusWidth, AlarmWndCmdTarget Property

Syntax SubItemStatusWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione degli allarmi. Se la colonna non è visualizzata, restituisce il valore -1.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemStatusWidth  
End Sub
```

SubItemText, AlarmWndCmdTarget Property

Syntax SubItemText = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Descrizione Allarme". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:
Sub Click()
 Debug.Print **SubItemText**
End Sub

SubItemTextPos, AlarmWndCmdTarget Property

Syntax SubItemTextPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Descrizione Allarme" all'interno della Finestra Allarmi. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:
Sub Click()
 Debug.Print **SubItemTextPos**
End Sub

SubItemTextWidth, AlarmWndCmdTarget Property

Syntax SubItemTextWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione degli allarmi. Se la colonna non è visualizzata, restituisce il valore -1.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
-----------	-------------

None	None
------	------

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemTextWidth  
End Sub
```

SubItemTotalNumAck, AlarmWndCmdTarget Property

Syntax SubItemTotalNumAck = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Numero Totale ACK". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Sub Click()  
    Debug.Print SubItemTotalNumAck  
End Sub
```

SubItemTotalNumAckPos, AlarmWndCmdTarget Property

Syntax SubItemTotalNumAckPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Numero Totale ACK" all'interno della Finestra Allarmi. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemTotalNumAckPos  
End Sub
```

SubItemTotalNumAckWidth, AlarmWndCmdTarget Property

Syntax SubItemTotalNumAckWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna "Numero Totale ACK" all'interno della finestra di visualizzazione degli allarmi. Se la colonna non è visualizzata, restituisce il valore -1.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemTotalNumAckWidth  
End Sub
```

SubItemTotalNumOn, AlarmWndCmdTarget Property

Syntax SubItemTotalNumOn = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Numero Totale ON". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Sub Click()  
    Debug.Print SubItemTotalNumOn
```

End Sub

SubItemTotalNumOnPos, AlarmWndCmdTarget Property

Syntax SubItemTotalNumOnPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Numero Totale ON" all'interno della Finestra Allarmi. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemTotalNumOnPos  
End Sub
```

SubItemTotalNumOnWidth, AlarmWndCmdTarget Property

Syntax SubItemTotalNumOnWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna "Numero Totale ON" all'interno della finestra di visualizzazione degli allarmi. Se la colonna non è visualizzata, restituisce il valore -1.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemTotalNumOnWidth  
End Sub
```

SubItemTotalNumReset, AlarmWndCmdTarget Property

Syntax SubItemTotalNumReset = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Numero Totale RESET". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Sub Click()  
    Debug.Print SubItemTotalNumReset  
End Sub
```

SubItemTotalNumResetPos, AlarmWndCmdTarget Property

Syntax SubItemTotalNumResetPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Numero Totale RESET" all'interno della Finestra Allarmi. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemTotalNumResetPos  
End Sub
```

SubItemTotalNumResetWidth, AlarmWndCmdTarget Property

Syntax SubItemTotalNumResetWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna "Numero Totale RESET" all'interno della finestra di visualizzazione degli allarmi. Se la colonna non è visualizzata, restituisce il valore -1.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemTotalNumResetWidth  
End Sub
```

SubItemTotalTimeOn, AlarmWndCmdTarget Property

Syntax SubItemTotalTimeOn = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Tempo Totale ON". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result String

Example:

```
Sub Click()  
    Debug.Print SubItemTotalTimeOn  
End Sub
```

SubItemTotalTimeOnPos, AlarmWndCmdTarget Property

Syntax SubItemTotalTimeOnPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Tempo Totale ON" all'interno della Finestra Allarmi. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemTotalTimeOnPos  
End Sub
```

SubItemTotalTimeOnWidth, AlarmWndCmdTarget Property

Syntax SubItemTotalTimeOnWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna "Tempo Totale ON" all'interno della finestra di visualizzazione degli allarmi. Se la colonna non è visualizzata, restituisce il valore -1.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Integer

Example:

```
Sub Click()  
    Debug.Print SubItemTotalTimeOnWidth  
End Sub
```

1.14.4. ButtonCmdTarget

Functions

GetCommandsInterfaceOnPressed, ButtonCmdTarget Function

Syntax GetCommandsInterfaceOnPressed()

Description Questa funzione consente di caricare l'interfaccia CommandsListCmdTarget relativa alla lista comandi del pulsante referenziato. Tramite questa interfaccia è possibile modificare la lista "Comandi su Pulsante Premuto" dell'oggetto referenziato.

Parameter	Description
-----------	-------------

None	None
------	------

Result Object: restituisce un oggetto di tipo CommandsListCmdTarget.

Example1:

```
Public Sub Click()
    Dim objButton As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objButton =
    GetSynopticObject.GetSubObject("objButton").GetObjectInterface
    Set objCommandList = objButton.GetCommandsInterfaceOnPressed

    Set objCommandList = Nothing
    Set objButton = Nothing
End Sub
```

GetCommandsInterfaceOnRelease, ButtonCmdTarget Function

Syntax GetCommandsInterfaceOnRelease ()

Description Questa funzione consente di caricare l'interfaccia CommandsListCmdTarget relativa alla lista comandi del pulsante referenziato. Tramite questa interfaccia è possibile modificare la lista "Comandi su Rilascio" dell'oggetto referenziato.

Parameter	Description
None	None

Result Object: restituisce un oggetto di tipo CommandsListCmdTarget.

Example1:

```
Public Sub Click()
    Dim objButton As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objButton =
    GetSynopticObject.GetSubObject("objButton").GetObjectInterface
    Set objCommandList = objButton.GetCommandsInterfaceOnRelease

    Set objCommandList = Nothing
    Set objButton = Nothing
End Sub
```

GetCommandsInterfaceWhileDown, ButtonCmdTarget Function

Syntax GetCommandsInterfaceWhileDown()

Description Questa funzione consente di caricare l'interfaccia CommandsListCmdTarget relativa alla lista comandi del pulsante referenziato. Tramite questa interfaccia è possibile modificare la lista "Comandi su Mantiene Premuto" dell'oggetto referenziato.

Parameter	Description
None	None

Result Object: restituisce un oggetto di tipo CommandsListCmdTarget.

Example1:

```
Public Sub Click()
    Dim objButton As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objButton =
    GetSynopticObject.GetSubObject("objButton").GetObjectInterface
    Set objCommandList = objButton.GetCommandsInterfaceWhileDown

    Set objCommandList = Nothing
    Set objButton = Nothing
End Sub
```

GetShorcutText, ButtonCmdTarget Function

Syntax GetShorcutText()

Description Questa funzione restituisce il testo dell'acceleratore impostato nel pulsante referenziato.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()
    Dim objButton As ButtonCmdTarget

    Set objButton =
    GetSynopticObject.GetSubObject("objButton").GetObjectInterface
    MsgBox "Shorcut Text = " &
    objButton.GetShorcutText,vbInformation,GetProjectTitle

    Set objButton = Nothing
End Sub
```

Properties

AsciiKeyShortcut, ButtonCmdTarget Property

Syntax AsciiKeyShortcut = _Byte

Description Questa proprietà imposta o restituisce il codice ASCII del tasto da utilizzare come acceleratore per il pulsante referenziato.

Parameter	Description
None	None

Result Byte

Example1:

```
Public Sub Click()
    Dim objButton As ButtonCmdTarget
    Dim bCode As String

    Set objButton =
    GetSynopticObject.GetSubObject("objButton").GetObjectInterface
    bCode = InputBox("Insert Shortcut Key:", "Button
    Prop", Chr(objButton.AsciiKeyShortcut))
    objButton.AsciiKeyShortcut = Asc(bCode)

    Set objButton = Nothing
End Sub
```

Border, ButtonCmdTarget Property

Syntax Border = eBorderType

Description Questa proprietà imposta o restituisce il tipo di bordo per il pulsante referenziato. Il tipo di bordo si può specificare usando l'enumeratore eBorderType oppure inserendo il corrispondente valore numerico:

```
enum_bt_none (valore 0, Semplice)
enum_bt_EDGE_BUMP (valore 1, A Sbalzo)
enum_bt_EDGE_ETCHED (valore 2, Inciso)
enum_bt_EDGE_RAISED (valore 3, Rialzato)
enum_bt_EDGE_SUNKEN (valore 4, Infossato)
```

Parameter	Description
None	None

Result eBorderType

Example1:

```
Public Sub Click()
    Dim objButton As ButtonCmdTarget

    Set objButton =
    GetSynopticObject.GetSubObject("objButton").GetObjectInterface
    objButton.Border = enum_bt_EDGE_RAISED

    Set objButton = Nothing
End Sub
```

ButtonStyle, ButtonCmdTarget Property

Syntax ButtonStyle = eButtonStyle

Description Questa proprietà imposta o restituisce il tipo di stile per il pulsante referenziato. Il tipo di stile si può specificare usando l'enumeratore eButtonStyle oppure inserendo il corrispondente valore numerico:

enum_windowsstylebutton (valore 0, Normale)
enum_style3D (valore 0, 3D)
enum_yellowlight (valore 0, luce gialla)
enum_bluelight (valore 0, luce blu)
enum_greenlight (valore 0, luce verde)
enum_redlight (valore 0, luce rossa)
enum_yellowbutton (valore 0, pulsante con luce gialla)
enum_bluebutton (valore 0, pulsante con luce blu)
enum_greenbutton (valore 0, pulsante con luce verde)
enum_redbutton (valore 0, pulsante con luce rossa)
enum_emergencya (valore 0, Emergenza A)
enum_emergencyb (valore 0, Emergenza B)
enum_squareb (valore 0, quadrato blu)
enum_squarer (valore 0, quadrato rosso)
enum_squarey (valore 0, quadrato giallo)
enum_squareg (valore 0, quadrato verde)
enum_squaren (valore 0, quadrato nero)
enum_switcha (valore 0, interruttore A)
enum_switchb (valore 0, interruttore B)
enum_switchc (valore 0, interruttore C)
enum_selectora (valore 0, selettore A)
enum_selectorb (valore 0, selettore B)
enum_selectorc (valore 0, selettore C)
enum_switch3sthor (valore 0, interruttore 3 stati orizzontale)
enum_switch3stver (valore 0, interruttore 3 stati verticale)
enum_selector3sta (valore 0, selettore 3 stati A)
enum_selector3stb (valore 0, selettore 3 stati B)
enum_selector3stc (valore 0, selettore 3 stati C)
enum_selector3std (valore 0, selettore 3 stati D)
enum_selector3ste (valore 0, selettore 3 stati E)
enum_selector3stf (valore 0, selettore 3 stati F)
enum_selector3stg (valore 0, selettore 3 stati G)
enum_selector3sth (valore 0, selettore 3 stati zero centrale)

Parameter	Description
None	None

Result eButtonStyle

Example1:

```
Public Sub Click()  
    Dim objButton As ButtonCmdTarget  
  
    Set objButton =  
        GetSynopticObject.GetSubObject("objButton").GetObjectInterface  
        objButton.ButtonStyle = enum_bluebutton  
  
    Set objButton = Nothing  
End Sub
```

Clickable, ButtonCmdTarget Property

Syntax Clickable = _Boolean

Description Questa proprietà imposta o restituisce la proprietà "Cliccabile" del pulsante referenziato. Impostando la proprietà a False il pulsante non sarà più cliccabile.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()
    Dim objButton As ButtonCmdTarget

    Set objButton =
        GetSynopticObject.GetSubObject("objButton").GetObjectInterface
    objButton.Clickable = False

    Set objButton = Nothing
End Sub
```

CommandStateVariable, ButtonCmdTarget Property

Syntax CommandStateVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile inserita nella proprietà "Tag Comando/Stato" del pulsante referenziato. Questo è il nome della variabile su cui agisce il comando "ON-OFF" del pulsante.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()
    Dim objButton As ButtonCmdTarget
    Dim sVarName As String

    GetVariableNameFromList(sVarName)

    Set objButton =
        GetSynopticObject.GetSubObject("objButton").GetObjectInterface
    objButton.CommandStateVariable = sVarName

    Set objButton = Nothing
End Sub
```

CommandType, ButtonCmdTarget Property

Syntax CommandType = eMechanicStyle

Description Questa proprietà imposta o restituisce il tipo di comando per il pulsante referenziato. Il tipo di comando si può specificare usando l'enumeratore eMechanicStyle oppure inserendo il corrispondente valore numerico:

enum_ms_Command (valore 0, Esegui Comandi)

enum_ms_OnOff (valore 1, ON-OFF)
enum_ms_Impulsive (valore 2, Impulsivo)
enum_ms_TristateHor (valore 3, Tre-Stati Orizzontale)
enum_ms_TristateVer (valore 4, Tre-Stati Verticale)

Parameter	Description
None	None

Result eMechanicStyle

Example1:

```
Public Sub Click()
    Dim objButton As ButtonCmdTarget

    Set objButton =
    GetSynopticObject.GetSubObject("objButton").GetObjectInterface
    objButton.CommandType = enum_ms_OnOff

    Set objButton = Nothing
End Sub
```

DisableCommandsOnCheckedState, ButtonCmdTarget Property

Syntax DisableCommandsOnCheckedState = _Boolean

Description Questa proprietà imposta o restituisce il valore della proprietà "Comandi Condizionati" che consente di eseguire la lista comandi di un pulsante soltanto se la variabile inserita nella proprietà "Tag Comando/Stato" è a zero.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()
    Dim objButton As ButtonCmdTarget
    Set objButton = GetSynopticObject.GetSubObject("Button1").GetObjectInterface
    objButton.DisableCommandsOnCheckedState = True
    Set objButton = Nothing
End Sub
```

EnableShortcut, ButtonCmdTarget Property

Syntax EnableShortcut = _Boolean

Description Questa proprietà imposta o restituisce la proprietà "Abilita Acceleratore" del pulsante referenziato. Impostando la proprietà a False l'eventuale acceleratore del pulsante non sarà più abilitato.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()
    Dim objButton As ButtonCmdTarget

    Set objButton =
        GetSynopticObject.GetSubObject("objButton").GetObjectInterface
        objButton.EnableShortcut= True

    Set objButton = Nothing
End Sub
```

ExecuteCommandsOnMouseMove, ButtonCmdTarget Property

Syntax ExecuteCommandsOnMouseMove = _Boolean

Description Questa proprietà imposta o restituisce il valore della proprietà "Comando su Mouse Move" che consente di eseguire la lista "Comandi su Rilascio" dell'oggetto pulsante sul rilascio del tasto del mouse anche nel caso in cui, dopo la pressione del pulsante, la posizione del cursore sia stata spostata al di fuori dell'area del pulsante.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()
    Dim objButton As ButtonCmdTarget
    Set objButton = GetSynopticObject.GetSubObject("Button1").GetObjectInterface
    objButton.ExecuteCommandsOnMouseMove = True
    Set objButton = Nothing
End Sub
```

ImageBtnChecked, ButtonCmdTarget Property

Syntax ImageBtnChecked = _String

Description Questa proprietà imposta o restituisce il nome dell'immagine associata allo stato di "pulsante selezionato" del pulsante referenziato.

Parameter	Description
-----------	-------------

None	None
------	------

Result String

Example1:

```
Public Sub Click()
    Dim objButton As ButtonCmdTarget
    Dim sVarName As String

    GetVariableNameFromList(sVarName)

    Set objButton =
    GetSynopticObject.GetSubObject("objButton").GetObjectInterface
    objButton.ImageBtnChecked = "ImageChecked.bmp"

    Set objButton = Nothing
End Sub
```

ImageBtnDisabled, ButtonCmdTarget Property

Syntax ImageBtnDisabled = _String

Description Questa proprietà imposta o restituisce il nome dell'immagine associata allo stato di "pulsante disabilitato" del pulsante referenziato.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()
    Dim objButton As ButtonCmdTarget
    Dim sVarName As String

    GetVariableNameFromList(sVarName)

    Set objButton =
    GetSynopticObject.GetSubObject("objButton").GetObjectInterface
    objButton.ImageBtnDisabled = "ImageDisabled.bmp"

    Set objButton = Nothing
End Sub
```

ImageBtnPressed, ButtonCmdTarget Property

Syntax ImageBtnPressed = _String

Description Questa proprietà imposta o restituisce il nome dell'immagine associata allo stato di "pulsante premuto" del pulsante referenziato.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()
    Dim objButton As ButtonCmdTarget
    Dim sVarName As String

    GetVariableNameFromList(sVarName)

    Set objButton =
    GetSynopticObject.GetSubObject("objButton").GetObjectInterface
    objButton.ImageBtnPressed = "ImagePressed.bmp"

    Set objButton = Nothing
End Sub
```

ImageBtnReleased, ButtonCmdTarget Property

Syntax ImageBtnReleased = _String

Description Questa proprietà imposta o restituisce il nome dell'immagine associata allo stato di "pulsante rilasciato" del pulsante referenziato.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()
    Dim objButton As ButtonCmdTarget
    Dim sVarName As String

    GetVariableNameFromList(sVarName)

    Set objButton =
    GetSynopticObject.GetSubObject("objButton").GetObjectInterface
    objButton.ImageBtnReleased = "ImageReleased.bmp"

    Set objButton = Nothing
End Sub
```

ImpulsiveTime, ButtonCmdTarget Property

Syntax ImpulsiveTime = _Long

Description Questa proprietà imposta o restituisce il tempo di impulso per il pulsante referenziato. Questo valore verrà considerato solo per alcuni tipi di comando del pulsante.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()  
    Dim objButton As ButtonCmdTarget  
  
    Set objButton =  
        GetSynopticObject.GetSubObject("objButton").GetObjectInterface  
        objButton.ImpulsiveTime = 1000  
  
    Set objButton = Nothing  
End Sub
```

OverlapImageText, ButtonCmdTarget Property

Syntax OverlapImageText = _Boolean

Description Questa proprietà imposta o restituisce la proprietà "Sovrapponi Immagine" del pulsante referenziato. Impostando la proprietà a True l'eventuale immagine associata al pulsante verrà sovrapposta all'eventuale testo del titolo, in caso contrario l'immagine e il titolo verranno affiancati.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()  
    Dim objButton As ButtonCmdTarget  
  
    Set objButton =  
        GetSynopticObject.GetSubObject("objButton").GetObjectInterface  
        objButton.OverlapImageText= True  
  
    Set objButton = Nothing  
End Sub
```

RadioBtnNumOptions, ButtonCmdTarget Property

Syntax RadioBtnNumOptions = _Byte

Description Questa proprietà imposta o restituisce il numero di opzioni per l'oggetto "Pulsante di Opzione". Questa proprietà non verrà considerata se il pulsante non è di tipo "Pulsante di Opzione".

Parameter	Description
None	None

Result Byte

Example1:

```
Public Sub Click()  
    Dim objButton As ButtonCmdTarget  
    Dim bCode As String  
  
    Set objButton =  
        GetSynopticObject.GetSubObject("objButton").GetObjectInterface  
        objButton.RadioBtnNumOptions = 5  
  
    Set objButton = Nothing  
End Sub
```

RadioCheckBtnSize, ButtonCmdTarget Property

Syntax RadioCheckBtnSize = eRadioCheckSize

Description Questa proprietà imposta o restituisce la dimensione per l'oggetto "Pulsante di Opzione" o "Pulsante di Selezione". Questa proprietà non verrà considerata se il pulsante non è di tipo "Pulsante di Opzione" o "Pulsante di Selezione". La dimensione si può specificare usando l'enumeratore eRadioCheckSize oppure inserendo il corrispondente valore numerico:

enum_rcsz_small (valore 0, piccolo)
enum_rcsz_medium (valore 1, medio)
enum_rcsz_large (valore 2, grande)

Parameter	Description
None	None

Result eButtonSize

Example1:

```
Public Sub Click()  
    Dim objButton As ButtonCmdTarget  
    Dim bCode As String  
  
    Set objButton =  
        GetSynopticObject.GetSubObject("objButton").GetObjectInterface  
        objButton.RadioCheckBtnSize = enum_rcsz_medium  
  
    Set objButton = Nothing  
End Sub
```

Round3DStyle, ButtonCmdTarget Property

Syntax Round3DStyle = eBtnRoundLevel

Description Questa proprietà imposta o restituisce il tipo di smussatura per il pulsante 3D referenziato. Il tipo di stile si può specificare usando l'enumeratore eBtnRoundLevel oppure inserendo il corrispondente valore numerico:

enum_br1_none (valore 0)
enum_br1_small (valore 1)
enum_br1_medium (valore 2)
enum_br1_large (valore 3)

Parameter	Description
None	None

Result eBtnRoundLevel

Example1:

```
Public Sub Click()  
    Dim objButton As ButtonCmdTarget  
  
    Set objButton =  
        GetSynopticObject.GetSubObject("objButton").GetObjectInterface  
        objButton.Round3DStyle = enum_br1_large  
  
    Set objButton = Nothing  
End Sub
```

ShowShortcut, ButtonCmdTarget Property

Syntax ShowShortcut = _Boolean

Description Questa proprietà imposta o restituisce la proprietà "Mostra Acceleratore" del pulsante referenziato. Impostando la proprietà a True l'eventuale testo dell'acceleratore associato al pulsante verrà visualizzato a fianco del testo del titolo dell'oggetto.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()  
    Dim objButton As ButtonCmdTarget  
  
    Set objButton =  
        GetSynopticObject.GetSubObject("objButton").GetObjectInterface  
        objButton.ShowShortcut= True  
  
    Set objButton = Nothing  
End Sub
```

TriStateCentralZero, ButtonCmdTarget Property

Syntax TriStateCentralZero = _Boolean

Description Questa proprietà imposta o restituisce la proprietà "Zero Centrale" del pulsante referenziato. Impostando la proprietà a True la posizione di zero del selettore verrà visualizzata come posizione centrale. Questa proprietà verrà considerata soltanto se l'oggetto è un selettore a tre posizioni (Selettore Tre Stati).

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()  
    Dim objButton As ButtonCmdTarget  
  
    Set objButton =  
        GetSynopticObject.GetSubObject("objButton").GetObjectInterface  
        objButton.TriStateCentralZero= True  
  
    Set objButton = Nothing  
End Sub
```

VirtualKeyShortcut, ButtonCmdTarget Property

Syntax VirtualKeyShortcut = eVirtualKey

Description Questa proprietà imposta o restituisce l'eventuale tasto funzione da utilizzare in combinazione al tasto acceleratore definito per il pulsante referenziato. Il tipo di tasto funzione si può specificare usando l'enumeratore eVirtualKey oppure inserendo il corrispondente valore numerico:

enum_VK_NONE (valore 0, Nessun Tasto)
enum_VK_CTRL (valore 1, Tasto CTRL)
enum_VK_SHFT (valore 2, Tasto SHIFT)
enum_VK_ALT (valore 4, Tasto ALT)

Parameter	Description
None	None

Result eVirtualKey

Example1:

```
Public Sub Click()  
    Dim objButton As ButtonCmdTarget  
  
    Set objButton =  
        GetSynopticObject.GetSubObject("objButton").GetObjectInterface  
        objButton.VirtualKeyShortcut = enum_VK_SHFT
```

```
        Set objButton = Nothing
    End Sub
```

1.14.5. ChartWndCmdTarget

Events

OnErrorRecordset, ChartWndCmdTarget Event

Description Evento notificato in seguito a un errore durante l'acquisizione dei valori che costituiscono il grafico.

Parameter	Description
RecordsetError As String	Descrizione dettagliata dell'errore

OnRecordsetMoveNext, ChartWndCmdTarget Event

Description Evento notificato durante lo scorrimento del set di valori che soddisfano la query di selezione, ogni volta che il sistema acquisisce un nuovo valore.

Parameter	Description
NumRecord As Long	Indice del record attualmente puntato
bRet As Boolean	Abilitazione al proseguimento dello scorrimento

OnRecordsetQueryEnd, ChartWndCmdTarget Event

Description Evento notificato al termine dell'acquisizione dei valori che soddisfano la query di selezione.

Parameter	Description
None	None

OnRecordsetQueryStart, ChartWndCmdTarget Event

Description Evento notificato all'inizio dell'acquisizione dei valori che soddisfano la query di selezione.

Parameter	Description
None	None

Functions

GetChartInterface, ChartWndCmdTarget Function

Syntax GetChartInterface()

Description Questa funzione restituisce l'interfaccia "VtChart" relativa all'oggetto Grafico inserito in un sinottico.



Per qualsiasi ulteriore informazione sui grafici e sulle loro configurazioni, consultare la relativa guida in linea "First Impression 5.0 On-Line documentation" (VCFI5.HLP) contenuta nella cartella di installazione di Movicon.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo VtChart se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Option Explicit
Public Sub Click()
    Dim objChart As ChartWndCmdTarget
    Set objChart = GetSynopticObject.GetSubObject("Chart1").GetObjectInterface
    Dim obj As VtChart
    Set obj = objChart.GetChartInterface
    Debug.Print obj.Picture
End Sub
```

LoadExtSettings, ChartWndCmdTarget Function

Syntax LoadExtSettings

Description Questa funzione permette di caricare la configurazione dell'oggetto dal file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".XML".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
Dim objSymbol As ChartWndCmdTarget
Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
If objSymbol Is Nothing Then Exit Sub
objSymbol.ExtSettingsFile = "test.sxml"
objSymbol.LoadExtSettings
Set objSymbol = Nothing
End Sub
```

RecalcLayout, ChartWndCmdTarget Function

Syntax RecalcLayout()

Description Questa funzione esegue un'aggiornamento grafico dell'oggetto. E' necessario eseguire tale funzione dopo che e' stata modificata una proprietà il cui impatto grafico sia particolarmente consistente, come ad esempio la modifica della ElevationVariable.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
Dim ChartWnd As ChartWndCmdTarget
Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
If Not ChartWnd Is Nothing Then
ChartWnd.ElevationVariable = "VAR00003"
ChartWnd.RecalcLayout
End If
Set ChartWnd = Nothing
End Sub
```

SaveExtSettings, ChartWndCmdTarget Function

Syntax SaveExtSettings

Description Questa funzione permette di salvare la configurazione dell'oggetto nel file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
Dim objSymbol As ChartWndCmdTarget
```

```

Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
If objSymbol Is Nothing Then Exit Sub
objSymbol.ExtSettingsFile = "test.xml"
objSymbol.SaveExtSettings
Set objSymbol = Nothing
End Sub

```

Properties

AddStackVariable, ChartWndCmdTarget Property

Syntax AddStackVariable = _String

Description Questa proprietà imposta o restituisce il nome della 'variabile di Inserimento'. Il Grafico attualmente visualizza tutti i valori delle variabili di tipo array assegnate. In previsione sarà possibile utilizzare una variabile di "Inserimento" per aggiornare i valori del grafico sul cambiamento di stato di tale variabile.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        Debug.Print ChartWnd.AddStackVariable
    End If
    Set ChartWnd = Nothing
End Sub

```

ArrayType, ChartWndCmdTarget Property

Syntax ArrayType = _Integer

Description Questa proprietà imposta o restituisce il tipo di array di dati da visualizzare sul grafico.

I tipi possibili son:
enum_Byte
enum_Double
enum_Dword
enum_Float
enum_Word

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        If ChartWnd.ArrayType=enum_Float Then
            ChartWnd.ArrayType=enum_Dword
        Else
            ChartWnd.ArrayType=enum_Float
        End If
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub

```

BackupLink, ChartWndCmdTarget Property

Syntax BackupLink = _String

Description Questa proprietà permette di impostare la connessione ODBC associata al grafico. Questa proprietà è utile nel caso in cui si vogliano visualizzare dati archiviati in altri file.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim sConnectionString As String

    sConnectionString = "MyProject__BackupLink" 'DSN name

    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.BackupLink = sConnectionString
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub

```

Border, ChartWndCmdTarget Property

Syntax Border = _Integer

Description Questa proprietà imposta o restituisce il tipo di bordo grafico.

I tipi possibili son:
enum_bt_EDGE_BUMP
enum_bt_EDGE_ETCHED
enum_bt_EDGE_RAISED
enum_bt_EDGE_SUNKEN
enum_bt_none

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        If ChartWnd.Border = enum_bt_EDGE_BUMP Then
            ChartWnd.Border = enum_bt_EDGE_RAISED
        Else
            ChartWnd.Border = enum_bt_EDGE_BUMP
        End If
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub
```

Clickable, ChartWndCmdTarget Property

Syntax Clickable = _Boolean

Description Questa proprietà permette di definire se l'operatore potrà interagire con il grafico. Disabilitando questa proprietà il controllo non potrà infatti essere gestito né con il mouse né con la tastiera.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Clickable = Not ChartWnd.Clickable
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub
```

DataDefaultQuery, ChartWndCmdTarget Property

Syntax DataDefaultQuery = _String

Description Questa proprietà imposta o restituisce la query SQL di aggiornamento per i dati del database (Data Logger) associato all'oggetto grafico. Accetta un valore String.

La funzione di "RecalcLayout" verifica se questa proprietà è impostata con un valore. In tal caso la stringa viene inviata all'ODBC ed utilizzata per recuperare il recordset dei dati. I valori delle altre due proprietà "DataFilterBy" e "DataSortBy" vengono utilizzati solo nel caso in cui la proprietà "DataDefaultQuery" non è stata impostata. Questa proprietà può consentire di eseguire delle query di SELECT personalizzate. Ad esempio la seguente query recupera un recordset di massimo 10.000 valori dove ogni valore rappresenta la media dei valori registrati in un'ora:

```
DataDefaultQuery = "SELECT TOP 10000 0 As MSecCol, MIN(LocalCol) As LocalCol,
AVG(Cosine) As Cosine, AVG(Ramp) As Ramp FROM Data Logger GROUP BY
DatePart (dayofyear, LocalCol ), DatePart (Hour, LocalCol ) ORDER BY LocalCol
DESC"
```

Parameter	Description
None	None

Result String

Example:

Option Explicit

Public Sub Click()

```
Dim objChart As ChartWndCmdTarget
Set objChart = GetSynopticObject.GetSubObject("Chart1").GetObjectInterface
Begin Dialog UserDialog 370,154,"ChartWndCmdTarget" ' %GRID:10,7,1,1
GroupBox 20,7,340,84,"DataDefaultQuery",.GroupBox1
TextBox 100,28,250,56,.Query,1
Text 30,28,60,21,"Query",.Text1
OKButton 20,105,160,42
CancelButton 190,105,160,42
End Dialog
Dim dlg As UserDialog
dlg.Query = "DELETE FROM Log1sec"
If Dialog(dlg) <> -1 Then Exit Sub
objChart.DataDefaultQuery = dlg.Query
objChart.DataFilterBy = ""
objChart.DataSortBy = ""
objChart.RecalcLayout
```

End Sub

DataFilterBy, ChartWndCmdTarget Property

Syntax DataFilterBy = _String

Description Questa proprietà imposta o restituisce il campo "Filtro" per l'estrazione dei dati dal database associato all'oggetto grafico mediante il Data Logger. Il campo "Filtro" rispetta la sintassi SQL e corrisponde alla clausola "WHERE". Tale clausola viene utilizzata solo al momento dell'accesso al database cioè al lancio in run del progetto. Accetta un valore String.

Parameter	Description
None	None

Result String

Example:

Option Explicit

Public Sub Click()

```
Dim objChart As ChartWndCmdTarget
```

```

Set objChart = GetSynopticObject.GetSubObject("Chart1").GetObjectInterface
Begin Dialog UserDialog 370,154,"ChartWndCmdTarget" ' %GRID:10,7,1,1
GroupBox 20,7,340,84,"DataFilterBy",.GroupBox1
TextBox 100,28,250,21,.FromDate
TextBox 100,56,250,21,.ToDate
Text 30,28,60,21,"From",.Text1
Text 30,56,60,21,"To",.Text12
OKButton 20,105,160,42
CancelButton 190,105,160,42
End Dialog
Dim dlg As UserDialog
dlg.FromDate = Format(Now, c)
dlg.ToDate = Format(Now, c)
If Dialog(dlg) <> -1 Then Exit Sub
Dim dFromDate As Date
Dim dToDate As Date
Dim sDataFilterBy As String
dFromDate = CDate(dlg.FromDate)
dToDate = CDate(dlg.ToDate)
sDataFilterBy = "LocalCol >= { ts "" & Format(dFromDate, "yyyy\-\mm\-\dd
hh\:nn:ss") & "" } AND LocalCol <= { ts "" & Format(dToDate, "yyyy\-\mm\-\dd
hh\:nn:ss") & "" }"
objChart.DataDefaultQuery = ""
objChart.DataFilterBy = sDataFilterBy
objChart.DataSortBy = "LocalCol DESC"
objChart.RecalcLayout
End Sub

```

DataSortBy, ChartWndCmdTarget Property

Syntax DataSortBy = _String

Description Questa proprietà imposta o restituisce il campo "Ordina" per l'estrazione dei dati dal database associato all'oggetto grafico mediante il Data Logger. Il campo "Ordina" rispetta la sintassi SQL e corrisponde alla clausola "ORDER BY". Tale clausola viene utilizzata solo al momento dell'accesso al database cioè al lancio in run del progetto. Una volta caricato il sinottico occorre tenere presente che il primo record dei valori del grafico è il più recente in ordine di tempo. Accetta un valore String.

Parameter	Description
None	None

Result String

Example:

Option Explicit

Public Sub Click()

```

Dim objChart As ChartWndCmdTarget
Set objChart = GetSynopticObject.GetSubObject("Chart1").GetObjectInterface
Begin Dialog UserDialog 370,154,"ChartWndCmdTarget" ' %GRID:10,7,1,1
GroupBox 20,7,340,91,"DataSortBy",.GroupBox1
OptionGroup .Group1
OptionButton 60,28,280,28,"LocalCol ASC",.OptionButton1
OptionButton 60,56,280,28,"LocalCol DESC",.OptionButton12
OKButton 10,105,160,42
CancelButton 190,105,160,42
End Dialog
Dim dlg As UserDialog
If Dialog(dlg) <> -1 Then Exit Sub
objChart.DataDefaultQuery = ""
objChart.DataFilterBy = ""
Select Case dlg.Group1
Case 0

```

```

objChart.DataSortBy = "LocalCol ASC"
Case 1
objChart.DataSortBy = "LocalCol DESC"
End Select
objChart.RecalcLayout
End Sub

```

ExtSettingsFile, ChartWndCmdTarget Property

Syntax ExtSettingsFile = _String

Description Questa proprietà imposta o restituisce il file di configurazione esterna per l'oggetto referenziato. Il file può essere specificato anche in fase di design nella proprietà "File di Configurazione" dell'oggetto. L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
Dim objSymbol As ChartWndCmdTarget
Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
If objSymbol Is Nothing Then Exit Sub
objSymbol.ExtSettingsFile = "test.sxml"
objSymbol.SaveExtSettings
Set objSymbol= Nothing
End Sub

```

ElevationVariable, ChartWndCmdTarget Property

Syntax ElevationVariable = _String

Description Questa proprietà permette di definire il nome della variabile che determina la rotazione "verticale" del grafico tridimensionale. L'angolo di rotazione visuale del grafico 3D viene in tal caso determinato in runtime dal valore della variabile associata.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Public Sub Click()
Dim ChartWnd As ChartWndCmdTarget
Dim tmpString As String
Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
If Not ChartWnd Is Nothing Then

```

```

        tmpString = ChartWnd.RotationVariable
        ChartWnd.RotationVariable = ChartWnd.ElevationVariable
        ChartWnd.ElevationVariable = tmpString
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub

```

LinkedDataLogger, ChartWndCmdTarget Property

Syntax LinkedDataLogger = _String

Description Questa proprietà imposta o restituisce il nome del Data Logger associato al grafico.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.LinkedDataLogger = "DLR5sec"
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub

```

NetworkBackupServerName, ChartWndCmdTarget Property

Syntax NetworkBackupServerName = _String

Description Questa proprietà imposta o restituisce il nome dell'eventuale Server di Networking di Backup usato per prelevare i dati da visualizzare nel Grafico quando il Server Primario, ovvero quello impostato nella proprietà "NetowrkServerName", è in timeout.

Parameter	Description
None	None

Result String

Example:


```

Dim objChart As ChartWndCmdTarget
Public Sub Click()
    Debug.Print objChart.NetworkBackupServerName
End Sub
Public Sub SymbolLoading()
    Set objChart =
    GetSynopticObject.GetSubObject("ChartWindow").GetObjectInterface
End Sub

```

NetworkServerName, ChartWndCmdTarget Property

Syntax NetworkServerName = _String

Description Questa proprietà restituisce il nome dell'eventuale Server di Networking dal quale dovranno essere prelevati i dati da visualizzare nel Grafico.

Parameter	Description
None	None

Result String

Example:

```

Dim objChart As ChartWndCmdTarget
Public Sub Click()
    Debug.Print objChart.NetworkServerName
End Sub
Public Sub SymbolLoading()
    Set objChart =
    GetSynopticObject.GetSubObject("ChartWnd").GetObjectInterface
End Sub

```

NumSamples, ChartWndCmdTarget Property

Syntax NumSamples = _Long

Description Questa proprietà imposta o restituisce il numero di valori (campionamenti) che devono essere rappresentati nel grafico. Il valore di default (20) significa che il grafico visualizzerà 20 valori in funzione del tipo di dato specificato, a prescindere dalla dimensione dell'array che è espressa in byte ed è riferita alla variabile.

Parameter	Description
None	None

Result Long

Example:

```

Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        Debug.Print ChartWnd.NumSamples
    End If

```

```
        Set ChartWnd = Nothing
End Sub
```

RotationVariable, ChartWndCmdTarget Property

Syntax RotationVariable = _String

Description Questa proprietà permette di definire il nome della variabile che determina la rotazione "orizzontale" del grafico tridimensionale. L'angolo di rotazione visuale del grafico 3D viene in tal caso determinato in runtime dal valore della variabile associata.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        tmpString = ChartWnd.RotationVariable
        ChartWnd.RotationVariable = ChartWnd.ElevationVariable
        ChartWnd.ElevationVariable = tmpString
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub
```

Title1, ChartWndCmdTarget Property

Syntax Title1 = _String

Description Questa proprietà permette di associare un nome alla curva numero 1 rappresentata nel grafico.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Title1 = "Curve 1"
    End If
End Sub
```

```

        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub

```

Title2, ChartWndCmdTarget Property

Syntax Title2 = _String

Description Questa proprietà permette di associare un nome alla curva numero 2 rappresentata nel grafico.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Title2 = "Curve 2"
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub

```

Title3, ChartWndCmdTarget Property

Syntax Title3 = _String

Description Questa proprietà permette di associare un nome alla curva numero 3 rappresentata nel grafico.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Title3 = "Curve 3"
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub

```

Title4, ChartWndCmdTarget Property

Syntax Title4 = _String

Description Questa proprietà permette di associare un nome alla curva numero 4 rappresentata nel grafico.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Title4 = "Curve 4"
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub
```

Title5, ChartWndCmdTarget Property

Syntax Title5 = _String

Description Questa proprietà permette di associare un nome alla curva numero 5 rappresentata nel grafico.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Title5 = "Curve 5"
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub
```

Title6, ChartWndCmdTarget Property

Syntax Title6 = _String

Description Questa proprietà permette di associare un nome alla curva numero 6 rappresentata nel grafico.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Title6 = "Curve 6"
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub
```

Title7, ChartWndCmdTarget Property

Syntax Title7 = _String

Description Questa proprietà permette di associare un nome alla curva numero 7 rappresentata nel grafico.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Title7 = "Curve 7"
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub
```

Title8, ChartWndCmdTarget Property

Syntax Title8 = _String

Description Questa proprietà permette di associare un nome alla curva numero 8 rappresentata nel grafico.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Title8 = "Curve 8"
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub
```

Variable1, ChartWndCmdTarget Property

Syntax Variable1 = _String

Description Questa proprietà permette di impostare il nome della variabile associata alla curva numero 1 del grafico.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Variable1 = "VAR00001"
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub
```

Variable2, ChartWndCmdTarget Property

Syntax Variable2 = _String

Description Questa proprietà permette di impostare il nome della variabile associata alla curva numero 2 del grafico.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Variable2 = "VAR00002"
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub
```

Variable3, ChartWndCmdTarget Property

Syntax Variable3 = _String

Description Questa proprietà permette di impostare il nome della variabile associata alla curva numero 3 del grafico.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Variable3 = "VAR00003"
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub
```

Variable4, ChartWndCmdTarget Property

Syntax Variable4 = _String

Description Questa proprietà permette di impostare il nome della variabile associata alla curva numero 4 del grafico.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Variable4 = "VAR00004"
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub
```

Variable5, ChartWndCmdTarget Property

Syntax Variable5 = _String

Description Questa proprietà permette di impostare il nome della variabile associata alla curva numero 5 del grafico.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Variable5 = "VAR00005"
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub
```


Variable6, ChartWndCmdTarget Property

Syntax Variable6 = _String

Description Questa proprietà permette di impostare il nome della variabile associata alla curva numero 6 del grafico.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Variable6 = "VAR00006"
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub
```

Variable7, ChartWndCmdTarget Property

Syntax Variable7 = _String

Description Questa proprietà permette di impostare il nome della variabile associata alla curva numero 7 del grafico.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Variable7 = "VAR00007"
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub
```

Variable8, ChartWndCmdTarget Property

Syntax Variable8 = _String

Description Questa proprietà permette di impostare il nome della variabile associata alla curva numero 8 del grafico.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim tmpString As String
    Set ChartWnd = GetSynopticObject.GetSubObject("Chart").GetObjectInterface
    If Not ChartWnd Is Nothing Then
        ChartWnd.Variable8 = "VAR00008"
        ChartWnd.RecalcLayout
    End If
    Set ChartWnd = Nothing
End Sub
```

1.14.6. ClientRulesInterface

Properties

ClientTimeout, ClientRulesInterface Property

Syntax ClientTimeout = _Long

Description Questa funzione imposta o restituisce il tempo di attesa (espresso in ms) sulla risposta del Server dopo l'invio di un messaggio da parte del Client. Allo scadere del Timeout verrà generato un messaggio di errore.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
    Dim NetwObj As NetworkClientCmd
    Dim ClientRulesObj As ClientRulesInterface
    Set NetwObj = GetNetworkClient
    If Not NetwObj Is Nothing Then
        Set ClientRulesObj = NetwObj.GetClientRules("")
        If Not ClientRulesObj Is Nothing Then
            Debug.Print ClientRulesObj.ClientTimeout
            Set ClientRulesObj = Nothing
        End If
        Set NetwObj = Nothing
    End Sub
```

End If
End Sub

DefaultClientUser, ClientRulesInterface Property

Syntax DefaultClientUser = _String

Description Questa funzione imposta o restituisce il nome dell'utente con il quale il Client si presenta al Server. Questa impostazione ha significato soltanto se il progetto Server ha la "Gestione Password" abilitata. In questo caso l'utente deve essere presente anche sul progetto Server in modo da essere riconosciuto. Così facendo il Client acquisisce i diritti associati all'utente in questione, e può avere l'accesso alle variabili in base a tali diritti.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim NetwObj As NetworkClientCmd  
    Dim ClientRulesObj As ClientRulesInterface  
    Set NetwObj = GetNetworkClient  
    If Not NetwObj Is Nothing Then  
        Set ClientRulesObj = NetwObj.GetClientRules("")  
        If Not ClientRulesObj Is Nothing Then  
            Debug.Print ClientRulesObj.DefaultClientUser  
            Set ClientRulesObj = Nothing  
        End If  
        Set NetwObj = Nothing  
    End If  
End Sub
```

Name, ClientRulesInterface Property

Syntax Name = _String

Description Questa funzione restituisce il server di connessione al quale la regola si riferisce. Se si inserisce una stringa (ad es. 'RegolaServer1'), va definito l'indirizzo IP, relativo a tale nome, nella proprietà 'Alias dei Server' presente nelle impostazioni Network Client, in caso contrario è possibile inserire in questo campo direttamente l'indirizzo IP del server a cui connettersi.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim NetwObj As NetworkClientCmd  
    Dim ClientRulesObj As ClientRulesInterface  
    Set NetwObj = GetNetworkClient
```

```

        If Not NetwObj Is Nothing Then
            Set ClientRulesObj = NetwObj.GetClientRules("")
            If Not ClientRulesObj Is Nothing Then
                Debug.Print ClientRulesObj.Name
                Set ClientRulesObj = Nothing
            End If
            Set NetwObj = Nothing
        End If
    End Sub

```

PingTime, ClientRulesInterface Property

Syntax PingTime = _Long

Description Questa funzione imposta o restituisce il tempo di ping da utilizzare durante la connessione al Server (il valore "0" evita l'utilizzo del tempo di ping).

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
    Dim NetwObj As NetworkClientCmd
    Dim ClientRulesObj As ClientRulesInterface
    Set NetwObj = GetNetworkClient
    If Not NetwObj Is Nothing Then
        Set ClientRulesObj = NetwObj.GetClientRules("")
        If Not ClientRulesObj Is Nothing Then
            Debug.Print ClientRulesObj.PingTime
            Set ClientRulesObj = Nothing
        End If
        Set NetwObj = Nothing
    End If
End Sub

```

Priority, ClientRulesInterface Property

Syntax Priority = _Integer

Description Questa funzione imposta o restituisce il livello di priorità definito per la connessione in questione. I valori ammessi vanno da 0 a 100. Un numero più alto corrisponde ad una priorità più alta. Ne risulta che la massima priorità è quindi 100.

Parameter	Description
None	None

Result Integer

Example:

```

Public Sub Click()
    Dim NetwObj As NetworkClientCmd
    Dim ClientRulesObj As ClientRulesInterface
    Set NetwObj = GetNetworkClient

```

```

        If Not NetwObj Is Nothing Then
            Set ClientRulesObj = NetwObj.GetClientRules("")
            If Not ClientRulesObj Is Nothing Then
                Debug.Print ClientRulesObj.Priority
                Set ClientRulesObj = Nothing
            End If
            Set NetwObj = Nothing
        End If
    End Sub

```

Protocol, ClientRulesInterface Property

Syntax Protocol = _Integer

Description Questa funzione imposta o restituisce il tipo di Protocollo che il Client dovrà utilizzare per comunicare con il Server.

Parameter	Description
None	None

Result Integer

Example:

```

Public Sub Click()
    Dim NetwObj As NetworkClientCmd
    Dim ClientRulesObj As ClientRulesInterface
    Set NetwObj = GetNetworkClient
    If Not NetwObj Is Nothing Then
        Set ClientRulesObj = NetwObj.GetClientRules("")
        If Not ClientRulesObj Is Nothing Then
            Debug.Print ClientRulesObj.Protocol
            Set ClientRulesObj = Nothing
        End If
        Set NetwObj = Nothing
    End If
End Sub

```

RasStation, ClientRulesInterface Property

Syntax RasStation = _String

Description Questa funzione imposta o restituisce il nome della connessione RAS (che deve essere stata creata in precedenza) per eseguire la connessione al Server sfruttando una linea telefonica.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim NetwObj As NetworkClientCmd
    Dim ClientRulesObj As ClientRulesInterface
    Set NetwObj = GetNetworkClient

```

```

        If Not NetwObj Is Nothing Then
            Set ClientRulesObj = NetwObj.GetClientRules("")
            If Not ClientRulesObj Is Nothing Then
                Debug.Print ClientRulesObj.RasStation
                Set ClientRulesObj = Nothing
            End If
            Set NetwObj = Nothing
        End If
    End Sub

```

UseRASStation, ClientRulesInterface Property

Syntax UseRASStation = _Boolean

Description Questa proprietà consente di abilitare l'utilizzo di una stazione RAS per eseguire la connessione al Server sfruttando una linea telefonica.

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub Click()
    Dim NetwObj As NetworkClientCmd
    Dim ClientRulesObj As ClientRulesInterface
    Set NetwObj = GetNetworkClient
    If Not NetwObj Is Nothing Then
        Set ClientRulesObj = NetwObj.GetClientRules("")
        If Not ClientRulesObj Is Nothing Then
            ClientRulesObj.RasStation = "StRAS0001"
            ClientRulesObj.UseRASStation = True
            Set ClientRulesObj = Nothing
        End If
        Set NetwObj = Nothing
    End If
End Sub

```

1.14.7. CommandAlarmCmdTarget

Properties

Action, CommandAlarmCmdTarget Property

Syntax Action= eAlarmCommand

Description Questa proprietà imposta o restituisce l'azione che dovrà eseguire il Comando Allarme referenziato. Il tipo di azione si può specificare usando l'enumeratore eAlarmCommand oppure inserendo il corrispondente valore numerico:

```

enum_ac_ackall (valore 0, Riconosci Tutti)
enum_ac_rstall (valore 1, Reset Tutti)
enum_ac_togglesound (valore 2, Abilita Suono)
enum_ac_viewreport (valore 3, Visualizza Report)
enum_ac_printreport (valore 4, Stampa Report)

```

enum_ac_exportreport (valore 5, Esporta Report)
enum_ac_ViewTextReport (valore 6, Visualizza Report Testuale)
enum_ac_PrintTextReport (valore 7, Stampa Report Testuale)
enum_ac_SaveTextReport (valore 8, Salva Report Testuale)
enum_ac_AppendTextReport (valore 9, Appendi Report Testuale)
enum_ac_ExportAndSendMail (valore 10, Esporta e Invia Email)
enum_ac_EmbeddedRptView (valore 11, Visualizza Report Interno)
enum_ac_EmbeddedRptPrint (valore 12, Stampa Report Interno)
enum_ac_EmbeddedRptSave (valore 13, Salva Report Interno)
enum_ac_EmbeddedRptMail (valore 14, Invia Report Interno)
enum_ac_rststatisticdata (valore 15, Reset Statistiche)



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result eAlarmCommand

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.Action = enum_ac_ackall
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.Action = enum_ac_ackall
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
```

```

Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

AreaFilter, CommandAlarmCmdTarget Property

Syntax AreaFilter= _String

Description Questa proprietà imposta o restituisce il nome dell'Area Allarmi impostata sul comando. In questo modo i comandi di "Riconosci Tutti" e "Reset Tutti" verranno eseguiti solo sugli allarmi dell'area specificata.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String: nome dell'Area Allarmi

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.AreaFilter = "AREA00001"
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

```



```

objCommandAlarm.AreaFilter = "AREA00001"
objCommandList.SaveChanges

Set objCommandAlarm = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

PrintSettingsLandscape, CommandAlarmCmdTarget Property

Syntax PrintSettingsLandscape = _Boolean

Description Tramite questa proprietà è possibile impostare la pagina Report con un orientamento verticale o orizzontale. Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Testuale", "Visualizza Report Interno", "Salva Report Interno", "Stampa Report Interno" o "Invia Report Interno". Impostando la proprietà a "False" la pagina avrà un orientamento verticale, impostandola a "True" avrà un orientamento orizzontale.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodiché al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Boolean

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.PrintSettingsLandscape = True
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandAlarm.PrintSettingsLandscape = True  
    objCommandList.SaveChanges  
  
    Set objCommandAlarm = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

PrintSettingsPageHeight, CommandAlarmCmdTarget Property

Syntax PrintSettingsPageHeight = _Long

Description Attraverso questo comando è possibile impostare l'altezza del foglio di stampa. Il valore viene impostato in millimetri e il valore -1 (valore di default) consente di utilizzare la dimensione di stampa della stampante. Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Interno".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objButtonRelease =  
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)
```

```

objCommandAlarm.PrintSettingsPageHeight = 100
objCommandList.SaveChanges

Set objCommandAlarm = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.PrintSettingsPageHeight = 100
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

PrintSettingsPageWidth, CommandAlarmCmdTarget Property

Syntax PrintSettingsPageWidth = _Long

Description Attraverso questo comando è possibile impostare la larghezza del foglio di stampa. Il valore viene impostato in millimetri e il valore -1 (valore di default) consente di utilizzare la dimensione di stampa della stampante. Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Interno".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget

```

```

Dim objCommandList As CommandsListCmdTarget
Dim objCommandAlarm As CommandAlarmCmdTarget

Set objButtonRelease =
GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

objCommandAlarm.PrintSettingsPageWidth = 70
objCommandList.SaveChanges

Set objCommandAlarm = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
Dim objRect As DrawCmdTarget
Dim objCommandList As CommandsListCmdTarget
Dim objCommandAlarm As CommandAlarmCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

objCommandAlarm.PrintSettingsPageWidth = 70
objCommandList.SaveChanges

Set objCommandAlarm = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

PrintSettingsPortSettings, CommandAlarmCmdTarget Property

Syntax PrintSettingsPortSettings = _String

Description Questa proprietà imposta o restituisce la stringa di configurazione della porta di stampa per il Comando Allarme referenziato. Questa impostazione viene utilizzata solo nel caso in cui nella proprietà "PrinterPort" è stata fatta una selezione tra "File", "Stampante di Rete", "Bluetooth Broadcom" o "Bluetooth Microsoft":

File: va impostato il percorso e il nome del file che il driver della stampante deve usare per salvare l'output di stampa (es. "\\FlashDrv\Output.prn")

Stampante di Rete: va impostato il percorso di rete della stampante (es. "\\ServerName\PrinterName")

Bluetooth Broadcom: vanno impostati tre valori delimitati dal carattere pipe ('|'). Il primo valore rappresenta l'indirizzo della scheda bluetooth (es. 00:0A:D9:EB:66:C7), il secondo valore rappresenta il nome del servizio da utilizzare e il terzo valore rappresenta il numero del canale

Bluetooth Microsoft: va impostato l'indirizzo della scheda bluetooth (es. 00:0A:D9:EB:66:C7)



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodiché al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di

configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objButtonRelease =  
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandAlarm.PrintSettingsPortSettings = "00:0A:D9:EB:66:C7"  
    objCommandList.SaveChanges  
  
    Set objCommandAlarm = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandAlarm.PrintSettingsPortSettings = "00:0A:D9:EB:66:C7"  
    objCommandList.SaveChanges  
  
    Set objCommandAlarm = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

PrintSettingsPrinterName, CommandAlarmCmdTarget Property

Syntax PrintSettingsPrinterName = _String

Description Tramite questo campo è possibile scegliere la stampante alla quale inviare il report. La stampante potrà essere selezionata tra le stampanti locali del PC. Se non è specificata nessuna stampante nel parametro verrà usata quella impostata come stampante di default di Windows. L'opzione "Scelta Stampante" avrà comunque la priorità su questa impostazione.
Nel caso in cui il progetto è impostato per la piattaforma Windows CE, l'elenco delle stampanti è fisso e riporta quelle supportate dal tool "PrintCE.dll", ovvero le seguenti stampanti:

- HP PCL 3
- Epson ESC/P 2
- Epson Stylus COLOR
- PocketJet II
- PocketJet 200
- Canon BJ (300 dpi)
- Canon BJ (360 dpi)
- Amtech
- Epson LX (9-pin)
- Adobe PDF file
- MTE W40
- Canon IP90
- Partner M1POS
- SP-T8
- Canon IP100
- Zebra
- MP-300
- O'Neil 4 inch
- O'Neil 3 inch
- HP PCL 5e

Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Testuale" o "Stampa Report Interno".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.PrintSettingsPrinterName = "Movicon PDF Writer"
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
```

End Sub

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandAlarm.PrintSettingsPrinterName = "Movicon PDF Writer"  
    objCommandList.SaveChanges  
  
    Set objCommandAlarm = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

PrintSettingsPrinterPort, CommandAlarmCmdTarget Property

Syntax PrintSettingsPrinterPort = ePrinterPorts

Description Questa proprietà imposta o restituisce la porta di stampa per il Comando Allarme referenziato. Il tipo di azione si può specificare usando l'enumeratore ePrinterPorts oppure inserendo il corrispondente valore numerico:

```
enum_port_Undefined (valore -1)  
enum_port_Infrared (valore 0)  
enum_port_COM1 (valore 1)  
enum_port_COM2 (valore 2)  
enum_port_COM3 (valore 3)  
enum_port_COM4 (valore 4)  
enum_port_COM5 (valore 5)  
enum_port_COM6 (valore 6)  
enum_port_COM7 (valore 7)  
enum_port_COM8 (valore 8)  
enum_port_File (valore 9)  
enum_port_NetworkPrinter (valore 10)  
enum_port_COM9 (valore 11)  
enum_port_COM10 (valore 12)  
enum_port_COM11 (valore 13)  
enum_port_COM12 (valore 14)  
enum_port_BluetoothBroadcom (valore 15)  
enum_port_BluetoothMicrosoft (valore 16)  
enum_port_LPT1 (valore 17)  
enum_port_USB (valore 18)
```



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodiché al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result ePrinterPorts

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.PrintSettingsPrinterPort = enum_port_LPT1
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.PrintSettingsPrinterPort = enum_port_LPT1
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

PrintSettingsShowPrintDialog, CommandAlarmCmdTarget Property

Syntax PrintSettingsShowPrintDialog = _Boolean

Description Abilitando questa opzione, prima della stampa del report, verrà aperta la finestra di dialogo per la selezione della stampante da utilizzare. Sarà quindi possibile scegliere quale stampante utilizzare tra quelle installate nel PC. Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Testuale" o "Stampa Report Interno".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto. Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla

memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.PrintSettingsShowPrintDialog = True
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.PrintSettingsShowPrintDialog = True
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

Recipient, CommandAlarmCmdTarget Property

Syntax Recipient = _String

Description In questo campo va inserito il nome dell'utente o del gruppo utenti a cui inviare la mail con allegato il file di report.
Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Esporta e Invia Email" o "Invia Report Interno".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objButtonRelease =  
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandAlarm.Recipient = "Progea"  
    objCommandList.SaveChanges  
  
    Set objCommandAlarm = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandAlarm.Recipient = "Progea"  
    objCommandList.SaveChanges  
  
    Set objCommandAlarm = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

StatisticRptFile, CommandAlarmCmdTarget Property

Syntax StatisticRptFile = _String

Description Questa proprietà imposta o restituisce il nome del file di report da utilizzare. Le scelte possibili sono:

- OrderByDate
- OrderByDuration
- GroupByFrequency
- GroupByThreshold



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String: nome del report da visualizzare/stampare

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objButtonRelease =  
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandAlarm.StatisticRptFile = "OrderByDate"  
    objCommandList.SaveChanges  
  
    Set objCommandAlarm = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)
```

```

objCommandAlarm.StatisticRptFile = "OrderByDate"
objCommandList.SaveChanges

Set objCommandAlarm = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

StatisticRptReferenceDate, CommandAlarmCmdTarget Property

Syntax StatisticRptReferenceDate = _String

Description Questa proprietà imposta o restituisce il periodo di riferimento che dovrà essere utilizzato per l'estrazione dei dati da visualizzare/stampare nel report. Il valore da inserire, di tipo stringa, dovrà riportare la data di inizio e di fine del periodo, utilizzando il formato: "dd/mm/yyyy hh:mm:ss dd/mm/yyyy hh:mm:ss".

Questa proprietà verrà presa in considerazione soltanto se la proprietà "StatisticRptReferenceDate" è impostata sul valore "None" o è vuota.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String: periodo di riferimento da visualizzare

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.StatisticRptReferenceDate = "01/01/2010 00:00:00
    02/01/2010 23:59:59"
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.StatisticRptReferenceDate = "01/01/2010 00:00:00
02/01/2010 23:59:59"
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

StatisticRptReferenceDuration, CommandAlarmCmdTarget Property

Syntax StatisticRptReferenceDuration= _String

Description Questa proprietà imposta o restituisce un filtro sulla durata di ogni allarme. Il valore di default è "00:00:00" ma è possibile impostare un filtro per cui vengano recuperati dal database solo gli allarmi che sono durati più di un certo tempo impostabile in "hh:mm:ss".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String: durata minima per gli allarmi da visualizzare/stampare

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.StatisticRptReferenceDuration = "01:30:00"
    objCommandList.SaveChanges

```

```

        Set objCommandAlarm = Nothing
        Set objCommandList = Nothing
        Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.StatisticRptReferenceDuration = "01:30:00"
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

StatisticRptReferencePeriod, CommandAlarmCmdTarget Property

Syntax StatisticRptReferencePeriod = _String

Description Questa proprietà imposta o restituisce il periodo di riferimento che dovrà essere utilizzato per l'estrazione dei dati da visualizzare/stampare nel report. I valori permessi, di tipo stringa sono i seguenti:

- None
- Today
- Yesterday or today
- Current week
- Current month
- Current year
- Last 7 days
- Last 30 days
- Last 60 days
- Last 90 days
- Last 1 years
- Last 2 years
- Last 5 years
- Last 10 years

E' anche possibile utilizzare la funzione "ConvertPeriodNumToString()" per convertire un valore di tipo eReportPeriod nella stringa richiesta dalla proprietà StatisticRptReferenceDate.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di

configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String: periodo di riferimento da visualizzare

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objButtonRelease =  
        GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandAlarm.StatisticRptReferencePeriod =  
        objCommandAlarm.ConvertPeriodNumToString(1)  
    objCommandList.SaveChanges  
  
    Set objCommandAlarm = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandAlarm.StatisticRptReferencePeriod =  
        objCommandAlarm.ConvertPeriodNumToString(1)  
    objCommandList.SaveChanges  
  
    Set objCommandAlarm = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

StatisticRptShowToolbar, CommandAlarmCmdTarget Property

Syntax StatisticRptShowToolbar = _Boolean

Description Questa proprietà permette di visualizzare o nascondere la barra degli strumenti nella finestra di anteprima del report. La proprietà verrà gestita soltanto se il report è stato creato con Crystal Report.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges

dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objButtonRelease =  
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandAlarm.StatisticRptShowToolbar = True  
    objCommandList.SaveChanges  
  
    Set objCommandAlarm = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandAlarm.StatisticRptShowToolbar = True  
    objCommandList.SaveChanges  
  
    Set objCommandAlarm = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

StatisticRptShowTree, CommandAlarmCmdTarget Property

Syntax `StatisticRptShowTree = _Boolean`

Description Questa proprietà permette di visualizzare o nascondere la struttura dei gruppi nella finestra di anteprima del report. La proprietà verrà gestita soltanto se il report è stato creato con Crystal Report.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.StatisticRptShowTree = True
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.StatisticRptShowTree = True
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

TextualRptBottomMargin, CommandAlarmCmdTarget Property

Syntax TextualRptBottomMargin = _Long

Description Questa proprietà permette impostare o restituire il margine di stampa basso. Il margine viene impostato in millimetri e il valore -1 (valore di default) consente di utilizzare l'eventuale margine di stampa di default recuperato attraverso il driver della stampante utilizzata.
Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Testuale".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodiché al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objButtonRelease =  
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandAlarm.TextualRptBottomMargin = 10  
    objCommandList.SaveChanges  
  
    Set objCommandAlarm = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandAlarm.TextualRptBottomMargin = 10
```

```

objCommandList.SaveChanges

Set objCommandAlarm = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

TextualRptLeftMargin, CommandAlarmCmdTarget Property

Syntax TextualRptLeftMargin = _Long

Description Questa proprietà permette impostare o restituire il margine di stampa sinistro. Il margine viene impostato in millimetri e il valore -1 (valore di default) consente di utilizzare l'eventuale margine di stampa di default recuperato attraverso il driver della stampante utilizzata.
Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Testuale".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.
Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.TextualRptLeftMargin = 10
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget

```

```

Dim objCommandList As CommandsListCmdTarget
Dim objCommandAlarm As CommandAlarmCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

objCommandAlarm.TextualRptLeftMargin = 10
objCommandList.SaveChanges

Set objCommandAlarm = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

TextualRptMaxPages, CommandAlarmCmdTarget Property

Syntax TextualRptMaxPages = _Long

Description Questa proprietà permette impostare o restituire il numero massimo di pagine stampate con un singolo comando "Visualizza Report Testuale", "Stampa Report Testuale", "Salva Report Testuale" e "Appendi Report Testuale". Impostando il valore su "0", il numero di pagine stampabili è illimitato (si consiglia di non impostare "0" per evitare di utilizzare troppa memoria o impegnare troppo la stampante in caso di errori nella formulazione della query di estrazione dei dati).



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.TextualRptMaxPages = 10
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing

```

```

        Set objCommandList = Nothing
        Set objButtonRelease = Nothing
    End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.TextualRptMaxPages = 10
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

TextualRptOutputFile, CommandAlarmCmdTarget Property

Syntax TextualRptOutputFile = _String

Description Questa proprietà permette impostare o restituire il nome del file con cui salvare il report testuale. Il parametro è richiesto solo dai comandi "Salva Report testuale" e "Appendi Report Testuale". Se non viene specificato il percorso il file verrà creato nella cartella "DLOGGERS" del progetto.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease

```

```

Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

objCommandAlarm.TextualRptOutputFile = "TestReport.rtf"
objCommandList.SaveChanges

Set objCommandAlarm = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.TextualRptOutputFile = "TestReport.rtf"
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

TextualRptRightMargin, CommandAlarmCmdTarget Property

Syntax TextualRptRightMargin = _Long

Description Questa proprietà permette impostare o restituire il margine di stampa destro. Il margine viene impostato in millimetri e il valore -1 (valore di default) consente di utilizzare l'eventuale margine di stampa di default recuperato attraverso il driver della stampante utilizzata.
Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Testuale".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.
Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.TextualRptRightMargin = 10
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.TextualRptRightMargin = 10
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

TextualRptSQLQuery, CommandAlarmCmdTarget Property

Syntax TextualRptSQLQuery = _String

Description Questa proprietà permette impostare o restituire la query da utilizzare per estrarre i dati di un "Report Testuale" o di un "Report Interno". "Nei Report Testuali" per default, i dati verranno estratti dalla tabella "Alarms" del Log Storico. Tramite la query però è possibile specificare una diversa tabella da cui estrarre i dati (Drivers o SysMsgs).
In questo campo va inserito il testo della query che dovrà essere coerente con il database utilizzato e non è possibile inserire il nome di una variabile per rendere dinamica la query.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodiché al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.TextualRptSQLQuery = "Select * From Alarms"
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.TextualRptSQLQuery = "Select * From Alarms"
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

TextualRptTemplateFile, CommandAlarmCmdTarget Property

Syntax TextualRptTemplateFile = _String

Description Questa proprietà permette impostare o restituire il nome del file di Layout del report testuale che, oltre alla formattazione, contiene anche i campi speciali da utilizzare. Se non viene specificato il percorso il file verrà cercato nella cartella Risorse del progetto.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere

persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()

    Dim objButtonRelease As ButtonCmdTarget

    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.TextualRptTemplateFile = "TemplateRpt.rtf"
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTaDim objCommandAlarm As
    CommandAlarmCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.TextualRptTemplateFile = "TemplateRpt.rtf"
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

TextualRptTopMargin, CommandAlarmCmdTarget Property

Syntax TextualRptTopMargin = _Long

Description Questa proprietà permette impostare o restituire il margine di stampa superiore. Il margine viene impostato in millimetri e il valore -1 (valore di default) consente di utilizzare l'eventuale margine di stampa di default recuperato attraverso il driver della stampante utilizzata.

Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Testuale".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.TextualRptTopMargin = 10
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandAlarm.TextualRptTopMargin = 10
    objCommandList.SaveChanges

    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

Functions

ConvertPeriodNumToString, CommandAlarmCmdTarget Function

Syntax ConvertPeriodNumToString(_IPeriod)

Description Questa funzione consente di convertire un valore numerico dell'enumeratore eReportPeriod nella stringa richiesta dalla funzione "StatisticRptReferencePeriod". Il tipo di periodo si può specificare usando l'enumeratore eReportPeriod oppure inserendo il corrispondente valore numerico:

enum_rp_None (valore 0, Nessuno)
enum_rp_Today (valore 1, Oggi)
enum_rp_YesterdayorToday (valore 2, Ieri e Oggi)
enum_rp_CurrentWeek (valore 3, Settimana Corrente)
enum_rp_CurrentMonth (valore 4, Mese Corrente)
enum_rp_CurrentYear (valore 5, Anno Corrente)
enum_rp_Last7days (valore 6, Ultimi 7 Giorni)
enum_rp_Last30days (valore 7, Ultimi 30 Giorni)
enum_rp_Last60Days (valore 8, Ultimi 60 Giorni)
enum_rp_Last90days (valore 9, Ultimi 90 Giorni)
enum_rp_Last1year (valore 10, Ultimo Anno)
enum_rp_Last2years (valore 11, Ultimi 2 Anni)
enum_rp_Last5years (valore 12, Ultimi 5 Anni)
enum_rp_Last10years (valore 13, Ultimi 10 Anni)

Parameter	Description
_IPeriod as eReportPeriod	Valore di periodo da convertire in stringa.

Result String

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objButtonRelease =  
        GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandAlarm.ConvertPeriodNumToString(enum_rp_Today)  
  
    Set objCommandAlarm = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandAlarm As CommandAlarmCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandAlarm.ConvertPeriodNumToString(enum_rp_Today)
```

```

        Set objCommandAlarm = Nothing
        Set objCommandList = Nothing
        Set objRect = Nothing
    End Sub

```

GetCommandBaseInterface, CommandAlarmCmdTarget Function

Syntax GetCommandBaseInterface()

Description Questa funzione consente di caricare l'interfaccia CommandBaseCmdTarget relativa al tipo di comando referenziato.

Parameter	Description
None	None

Result Object: restituisce un oggetto di tipo CommandBaseCmdTarget.

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    Set objCommandBase = objCommandAlarm.GetCommandBaseInterface

    Set objCommandBase = Nothing
    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = NothingEnd Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandAlarm As CommandAlarmCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandAlarm = objCommandList.GetCommandInterfaceAtPos(0)

    Set objCommandBase = objCommandAlarm.GetCommandBaseInterface

    Set objCommandBase = Nothing
    Set objCommandAlarm = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

1.14.8. CommandBaseCmdTarget

Properties

Type, CommandBaseCmdTarget Property

Syntax Type = eCommandTypes

Description Questa proprietà è di sola lettura e restituisce un valore che identifica il tipo di comando referenziato. Il valore di ritorno è l'enumeratore eCommandTypes:

enum_ct_synoptic (valore 0, Comando Sinottico)
enum_ct_script (valore 1, Comando Script)
enum_ct_variable (valore 2, Comando Variabile)
enum_ct_changelanguage (valore 3, Comando Lingua)
enum_ct_menu (valore 4, Comando Menù)
enum_ct_report (valore 5, Comando Report)
enum_ct_system (valore 6, Comando Sistema)
enum_ct_users (valore 7, Comando Utente)
enum_ct_help (valore 8, Comando Help)
enum_ct_alarm (valore 9, Comando Allarme)
enum_ct_event (valore 10, Comando Evento)

Parameter	Description
None	None

Result eCommandTypes

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandBase As CommandBaseCmdTarget  
    Dim objObject As Object  
  
    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objObject = objCommandList.GetCommandInterfaceAtPos(0)  
    Set objCommandBase = objObject.GetCommandBaseInterface  
  
    MsgBox "Command " & objCommandBase.Type & " = " & objCommandBase.Type & " &  
    CStr(objCommandBase.Type), vbInformation, GetProjectTitle  
  
    Set objObject = Nothing  
    Set objCommandBase = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As CommandBaseCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandBase As CommandBaseCmdTarget  
    Dim objObject As Object  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objObject = objCommandList.GetCommandInterfaceAtPos(0)
```

```

Set objCommandBase = objObject.GetCommandBaseInterface

MsgBox "Command Type = " &
CStr(objCommandBase.Type),vbInformation,GetProjectTitle

Set objObject = Nothing
Set objCommandBase = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

XmlSettings, CommandBaseCmdTarget Property

Syntax XmlSettings = _string

Description Questa proprietà è di sola lettura e restituisce il codice xml del comando referenziato.

Parameter	Description
None	None

Result String

Example1:

```

Public Sub Click()
Dim objButtonRelease As ButtonCmdTarget
Dim objCommandList As CommandsListCmdTarget
Dim objCommandBase As CommandBaseCmdTarget
Dim objObject As Object

Set objButtonRelease =
GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
Set objObject = objCommandList.GetCommandInterfaceAtPos(0)
Set objCommandBase = objObject.GetCommandBaseInterface

MsgBox "Command XmlSettings = " &
CStr(objCommandBase.XmlSettings),vbInformation,GetProjectTitle

Set objObject = Nothing
Set objCommandBase = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
Dim objRect As CommandBaseCmdTarget
Dim objCommandList As CommandsListCmdTarget
Dim objCommandBase As CommandBaseCmdTarget
Dim objObject As Object

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objObject = objCommandList.GetCommandInterfaceAtPos(0)
Set objCommandBase = objObject.GetCommandBaseInterface

```

```

MsgBox "Command XmlSettings = " &
CStr(objCommandBase.XmlSettings),vbInformation,GetProjectTitle

Set objObject = Nothing
Set objCommandBase = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

1.14.9. CommandEventCmdTarget

Property

Event, CommandEventCmdTarget Property

Syntax Event = _String

Description Questa proprietà imposta o restituisce il nome dell'oggetto Evento che dovrà essere eseguito dal comando referenziato.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandEvent As CommandEventCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandEvent = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandEvent.Event = "EventTest"
    objCommandList.SaveChanges

    Set objCommandEvent = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandEvent As CommandEventCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandEvent = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandEvent.Event = "EventTest"
    objCommandList.SaveChanges

    Set objCommandEvent = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

Functions

GetCommandBaseInterface, CommandEventCmdTarget Function

Syntax GetCommandBaseInterface()

Description Questa funzione consente di caricare l'interfaccia CommandBaseCmdTarget relativa al tipo di comando referenziato.

Parameter	Description
None	None

Result Object: restituisce un oggetto di tipo CommandBaseCmdTarget.

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandEvent As CommandEventCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandEvent = objCommandList.GetCommandInterfaceAtPos(0)

    Set objCommandBase = objCommandEvent.GetCommandBaseInterface

    Set objCommandBase = Nothing
    Set objCommandEvent = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandEvent As CommandEventCmdTarget

```



```

Dim objCommandBase As CommandBaseCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandEvent = objCommandList.GetCommandInterfaceAtPos(0)

Set objCommandBase = objCommandEvent.GetCommandBaseInterface

Set objCommandBase = Nothing
Set objCommandEvent = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

1.14.10. CommandHelpCmdTarget

Properties

Action, CommandHelpCmdTarget Property

Syntax Action = eHelpMode

Description Questa proprietà imposta o restituisce l'azione che dovrà eseguire il Comando Help referenziato. Il tipo di azione si può specificare usando l'enumeratore eHelpMode oppure inserendo il corrispondente valore numerico:

enum_hm_topic (valore 0, Topic)
enum_hm_tooltip (valore 1, ToolTip)



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result eHelpMode

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandHelp As CommandHelpCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandHelp = objCommandList.GetCommandInterfaceAtPos(0)

```

```

objCommandHelp.Action = enum_hm_tooltip
objCommandList.SaveChanges

Set objCommandHelp = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandHelp As CommandHelpCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandHelp = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandHelp.Action = enum_hm_tooltip
    objCommandList.SaveChanges

    Set objCommandHelp = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

Topic, CommandHelpCmdTarget Property

Syntax Topic= _String

Description Questa proprietà imposta o restituisce il nome del Topic o il testo del ToolTip da visualizzare, a seconda dell'impostazione del campo "Azione".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandHelp As CommandHelpCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface

```

```

Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
Set objCommandHelp = objCommandList.GetCommandInterfaceAtPos(0)

objCommandHelp.Topic = "Hello!"
objCommandHelp.SaveChanges

Set objCommandHelp = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandHelp As CommandHelpCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandHelp = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandHelp.Topic = "Hello!"
    objCommandHelp.SaveChanges

    Set objCommandHelp = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

Functions

GetCommandBaseInterface, CommandHelpCmdTarget Function

Syntax GetCommandBaseInterface()

Description Questa funzione consente di caricare l'interfaccia CommandBaseCmdTarget relativa al tipo di comando referenziato.

Parameter	Description
None	None

Result Object: restituisce un oggetto di tipo CommandBaseCmdTarget.

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandHelp As CommandHelpCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandHelp = objCommandList.GetCommandInterfaceAtPos(0)

    Set objCommandBase = objCommandHelp.GetCommandBaseInterface

```

```

        Set objCommandBase = Nothing
        Set objCommandHelp = Nothing
        Set objCommandList = Nothing
        Set objButtonRelease = Nothing
    End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandHelp As CommandHelpCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandHelp = objCommandList.GetCommandInterfaceAtPos(0)

    Set objCommandBase = objCommandHelp.GetCommandBaseInterface

    Set objCommandBase = Nothing
    Set objCommandHelp = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

1.14.11. CommandLanguageCmdTarget

Properties

Language, CommandLanguageCmdTarget Property

Syntax Language= _String

Description Questa proprietà imposta o restituisce la lingua che dovrà attivare il Comando Lingua referenziato. Il testo rappresenta il nome della colonna della Tabella Stringhe.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandLanguage As CommandLanguageCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandLanguage = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandLanguage.Language = "Italian"
    objCommandList.SaveChanges

    Set objCommandLanguage = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandLanguage As CommandLanguageCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandLanguage = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandLanguage.Language = "Italian"
    objCommandList.SaveChanges

    Set objCommandLanguage = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

Functions

GetCommandBaseInterface, CommandLanguageCmdTarget Function

Syntax GetCommandBaseInterface()

Description Questa funzione consente di caricare l'interfaccia CommandBaseCmdTarget relativa al tipo di comando referenziato.

Parameter	Description
None	None

Result Object: restituisce un oggetto di tipo CommandBaseCmdTarget.

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandLanguage As CommandLanguageCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease

```

```

Set objCommandLanguage = objCommandList.GetCommandInterfaceAtPos(0)

Set objCommandBase = objCommandLanguage.GetCommandBaseInterface

Set objCommandBase = Nothing
Set objCommandLanguage = Nothing
Set objCommandList = Nothing
Set objButtonRelease = NothingEnd Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandLanguage As CommandLanguageCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandLanguage = objCommandList.GetCommandInterfaceAtPos(0)

    Set objCommandBase = objCommandLanguage.GetCommandBaseInterface

    Set objCommandBase = Nothing
    Set objCommandLanguage = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

1.14.12. CommandsListCmdTarget

Functions

AddToTail, CommandsListCmdTarget Function

Syntax AddToTail(_nCommandType)

Description Questo metodo consente di aggiungere un nuovo comando in fondo alla Lista Comandi referenziata. Il parametro "nCommandType" definisce il tipo di comando che si vuole aggiungere. Il tipo di comando si può specificare usando l'enumeratore eCommandTypes oppure inserendo il corrispondente valore numerico:

```

enum_ct_synoptic (valore 0, Comando Sinottico)
enum_ct_script (valore 1, Comando Script)
num_ct_variable (valore 2, Comando Variabile)
enum_ct_changelanguage (valore 3, Comando Lingua)
enum_ct_menu (valore 4, Comando Menù)
num_ct_report (valore 5, Comando Report)
enum_ct_system (valore 6, Comando Sistema)
enum_ct_users (valore 7, Comando Utente)
enum_ct_help (valore 8, Comando Help)
enum_ct_alarm (valore 9, Comando Allarme)
enum_ct_event (valore 10, Comando Evento)

```

Parameter		Description
nCommandType eCommandTypes	as	Tipo di comando da aggiungere.

Result Object

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
  
    Set objButtonRelease =  
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
  
    objCommandList.AddToTail(enum_ct_variable)  
  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
  
    objCommandList.AddToTail(enum_ct_variable)  
  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

AddToHead, CommandsListCmdTarget Function

Syntax AddToHead(_nCommandType)

Description Questo metodo consente di aggiungere un nuovo comando in cima alla Lista Comandi referenziata. Il parametro "nCommandType" definisce il tipo di comando che si vuole aggiungere. Il tipo di comando si può specificare usando l'enumeratore eCommandTypes oppure inserendo il corrispondente valore numerico:

enum_ct_synoptic (valore 0, Comando Sinottico)
enum_ct_script (valore 1, Comando Script)
enum_ct_variable (valore 2, Comando Variabile)
enum_ct_changelanguage (valore 3, Comando Lingua)
enum_ct_menu (valore 4, Comando Menù)
enum_ct_report (valore 5, Comando Report)
enum_ct_system (valore 6, Comando Sistema)
enum_ct_users (valore 7, Comando Utente)
enum_ct_help (valore 8, Comando Help)
enum_ct_alarm (valore 9, Comando Allarme)
enum_ct_event (valore 10, Comando Evento)

Parameter	Description
nCommandType eCommandTypes	Tipo di comando da aggiungere.

Result Object

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget
```

```

Set objButtonRelease =
GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease

objCommandList.AddToHead(enum_ct_variable)

Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease

    objCommandList.AddToHead(enum_ct_variable)

    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

DiscardChanges, CommandsListCmdTarget Function

Syntax DiscardChanges()

Description Questo metodo consente di annullare (scaricare) le modifiche apportate alla lista comandi. Le modifiche che verranno scaricate saranno soltanto quelle apportate ma non ancora salvate tramite il comando "SaveChanges".

Parameter	Description
None	None

Result None

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease

    objCommandList.DiscardChanges

    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease

    objCommandList.DiscardChanges

```



```

        Set objCommandList = Nothing
        Set objRect = Nothing
    End Sub

```

GetCommandInterfaceAtPos, CommandsListCmdTarget Function

Syntax GetCommandTypeAtPos(_nIndex)

Description Questo metodo consente di sapere il tipo di comando presente ad una determinata posizione della Lista Comandi referenziata. Il parametro indice rappresenta la posizione del comando nella lista. Il valore numerico restituito corrisponde all'enumeratore eCommandTypes:

```

enum_ct_synoptic (valore 0, Comando Sinottico)
enum_ct_script (valore 1, Comando Script)
enum_ct_variable (valore 2, Comando Variabile)
enum_ct_changelanguage (valore 3, Comando Lingua)
enum_ct_menu (valore 4, Comando Menù)
enum_ct_report (valore 5, Comando Report)
enum_ct_system (valore 6, Comando Sistema)
enum_ct_users (valore 7, Comando Utente)
enum_ct_help (valore 8, Comando Help)
enum_ct_alarm (valore 9, Comando Allarme)
enum_ct_event (valore 10, Comando Evento)

```

Parameter	Description
nIndex as Long	Indice del comando di cui caricare il tipo. E' la posizione nella lista del comando. L'indice parte dal valore zero.

Result eCommandTypes

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Set objButtonRelease =
        GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease

    MsgBox "Command Type = " &
        objCommandList.GetCommandTypeAtPos(0), vbInformation, GetProjectTitle

    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease

    MsgBox "Command Type = " &
        objCommandList.GetCommandTypeAtPos(0), vbInformation, GetProjectTitle

    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

GetCommandTypeAtPos, CommandsListCmdTarget Function

Syntax GetCommandInterfaceAtPos(_nIndex)

Description Questo metodo consente di caricare un oggetto comando dalla lista comandi referenziata. Il parametro indice rappresenta la posizione del comando nella lista.

Parameter	Description
nIndex as Long	Indice del comando da caricare. E' la posizione nella lista del comando. L'indice parte dal valore zero.

Result Object

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objObject As Object  
  
    Set objButtonRelease =  
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
  
    Set objObject = objCommandList.GetCommandInterfaceAtPos(0)  
  
    Set objObject = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objObject As Object  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
  
    Set objObject = objCommandList.GetCommandInterfaceAtPos(0)  
  
    Set objObject = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

GetTotNumCommands, CommandsListCmdTarget Function

Syntax GetTotNumCommands

Description Questo metodo consente di sapere quanti comandi sono presenti nella Lista Comandi referenziata.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease

    MsgBox "Number of commands: " &
    objCommandList.GetTotNumCommands,vbInformation,GetProjectTitle

    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease

    MsgBox "Number of commands: " &
    objCommandList.GetTotNumCommands,vbInformation,GetProjectTitle

    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

InsertAfter, CommandsListCmdTarget Function

Syntax InsertAfter(_nIndex, _nCommandType)

Description Questo metodo consente di inserire un nuovo comando nella Lista Comandi referenziata. Il comando verrà inserito nella posizione successiva a quella specificata dal parametro nIndex e del tipo specificato nel parametro nCommandType. Il tipo di comando si può specificare usando l'enumeratore eCommandTypes oppure inserendo il corrispondente valore numerico:

```
enum_ct_synoptic (valore 0, Comando Sinottico)
enum_ct_script (valore 1, Comando Script)
enum_ct_variable (valore 2, Comando Variabile)
enum_ct_changelanguage (valore 3, Comando Lingua)
enum_ct_menu (valore 4, Comando Menù)
enum_ct_report (valore 5, Comando Report)
enum_ct_system (valore 6, Comando Sistema)
enum_ct_users (valore 7, Comando Utente)
enum_ct_help (valore 8, Comando Help)
enum_ct_alarm (valore 9, Comando Allarme)
enum_ct_event (valore 10, Comando Evento)
```

Parameter	Description
nCommandType as eCommandTypes	Tipo di comando da aggiungere.
nIndex as Long	Indice del comando dopo il quale inserire quello nuovo. E' la posizione nella lista del comando. L'indice parte dal valore zero.

Result Object

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease

    objCommandList.InsertAfter(0, enum_ct_variable)

    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease

    objCommandList.InsertAfter(0, enum_ct_variable)

    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

InsertBefore, CommandsListCmdTarget Function

Syntax InsertBefore (_nIndex, _nCommandType)

Description Questo metodo consente di inserire un nuovo comando nella Lista Comandi referenziata. Il comando verrà inserito nella posizione precedente a quella specificata dal parametro nIndex e del tipo specificato nel parametro nCommandType. Il tipo di comando si può specificare usando l'enumeratore eCommandTypes oppure inserendo il corrispondente valore numerico:

```
enum_ct_synoptic (valore 0, Comando Sinottico)
enum_ct_script (valore 1, Comando Script)
enum_ct_variable (valore 2, Comando Variabile)
enum_ct_changelanguage (valore 3, Comando Lingua)
enum_ct_menu (valore 4, Comando Menù)
enum_ct_report (valore 5, Comando Report)
enum_ct_system (valore 6, Comando Sistema)
enum_ct_users (valore 7, Comando Utente)
enum_ct_help (valore 8, Comando Help)
enum_ct_alarm (valore 9, Comando Allarme)
enum_ct_event (valore 10, Comando Evento)
```

Parameter	Description
nCommandType as eCommandTypes	Indice del comando prima del quale inserire quello nuovo. E' la posizione nella lista del comando. L'indice parte dal valore zero.
nIndex as Long	Indice del comando prima del quale inserire quello nuovo. E' la posizione nella lista del comando. L'indice parte dal valore zero.

Result Long

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease

    objCommandList.InsertBefore(0, enum_ct_variable)

    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease

    objCommandList.InsertBefore(0, enum_ct_variable)

    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

MoveToHead, CommandsListCmdTarget Function

Syntax MoveToHead(_nIndex)

Description Questo metodo consente di spostare il comando specificato dal parametro nIndex nella prima posizione della Lista Comandi referenziata.

Parameter	Description
nIndex as Long	Indice del comando che dovrà essere spostato all'inizio lista. E' la posizione nella lista del comando. L'indice parte dal valore zero.

Result Boolean

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
```

```

Set objButtonRelease =
GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease

objCommandList.MoveToHead(2)

Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
Dim objRect As DrawCmdTarget
Dim objCommandList As CommandsListCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease

objCommandList.MoveToHead(2)

Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

MoveToTail, CommandsListCmdTarget Function

Syntax MoveToTail(_nIndex)

Description Questo metodo consente di spostare il comando specificato dal parametro nIndex nell'ultima posizione della Lista Comandi referenziata.

Parameter	Description
nIndex as Long	Indice del comando che dovrà essere spostato alla fine della lista. E' la posizione nella lista del comando. L'indice parte dal valore zero.

Result Boolean

Example1:

```

Public Sub Click()
Dim objButtonRelease As ButtonCmdTarget
Dim objCommandList As CommandsListCmdTarget

Set objButtonRelease =
GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease

objCommandList.MoveToTail(1)

Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
Dim objRect As DrawCmdTarget
Dim objCommandList As CommandsListCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease

```

```

objCommandList.MoveToTail(1)

Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

RemoveAll, CommandsListCmdTarget Function

Syntax RemoveAll

Description Questo metodo consente di eliminare tutti i comandi della Lista Comandi referenziata.

Parameter	Description
None	None

Result None

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease

    objCommandList.RemoveAll()

    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease

    objCommandList.RemoveAll()

    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

RemoveAtPos, CommandsListCmdTarget Function

Syntax RemoveAtPos(_nIndex)

Description Questo metodo consente di eliminare il comando specificato dal parametro nIndex dalla Lista Comandi referenziata.

Parameter	Description
nIndex as Long	Indice del comando che dovrà essere eliminato. E' la posizione nella lista del comando. L'indice parte dal valore zero.

Result Boolean

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease

    objCommandList.RemoveAtPos(1)

    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease

    objCommandList.RemoveAtPos(1)

    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

RemoveFromHead, CommandsListCmdTarget Function

Syntax RemoveFromHead()

Description Questo metodo consente di eliminare il primo comando dalla Lista Comandi referenziata.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
```



```

        Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease

        objCommandList.RemoveFromHead()

        Set objCommandList = Nothing
        Set objButtonRelease = Nothing
    End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease

    objCommandList.RemoveFromHead()

    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

RemoveFromTail, CommandsListCmdTarget Function

Syntax RemoveFromTail()

Description Questo metodo consente di eliminare l'ultimo comando dalla Lista Comandi referenziata.

Parameter	Description
None	None

Result Boolean

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease

    objCommandList.RemoveFromTail()

    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease

    objCommandList.RemoveFromTail()

```

```

        Set objCommandList = Nothing
        Set objRect = Nothing
    End Sub

```

SaveChanges, CommandsListCmdTarget Function

Syntax SaveChanges()

Description Questo metodo consente di salvare la Lista Comandi referenziata per l'oggetto in questione. Le modifiche apportate alla lista comandi infatti saranno effettive ed eseguibili dall'oggetto soltanto dopo che il metodo SaveChanges è stato invocato.

Attenzione: le modifiche alla Lista Comandi degli oggetti, ad esclusione dei Pulsanti, verranno acquisite soltanto se all'inizializzazione dell'oggetto la sua Lista Comandi conteneva già almeno un comando. Nel caso questa sia inizialmente vuota l'inserimento di eventuali comandi non avrà effetto sull'oggetto. Nel caso in cui all'oggetto in questione sia stato associato un file di configurazione, il comando di salvataggio del file di configurazione salverà anche l'eventuale nuova Lista Comandi e in questo caso al successivo caricamento della pagina, e conseguente inizializzazione del controllo, la nuova Lista Comandi verrà acquisita.

Parameter	Description
None	None

Result Boolean

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease

    objCommandList.SaveChanges()

    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease

    objCommandList.SaveChanges()

    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

SetAtPos, CommandsListCmdTarget Function

Syntax SetAtPos(_nIndex, _nCommandType)

Description Questo metodo consente di modificare il tipo di comando identificato dal parametro nIndex nella Lista Comandi referenziata. Il comando esistente verrà reimpostato con il nuovo tipo specificato nel parametro nCommandType. Questa funzione non è in grado di aggiungere comandi nuovi ma può solo reimpostare un comando esistente. Il tipo di comando si può specificare usando l'enumeratore eCommandTypes oppure inserendo il corrispondente valore numerico:

enum_ct_synoptic (valore 0, Comando Sinottico)
enum_ct_script (valore 1, Comando Script)
enum_ct_variable (valore 2, Comando Variabile)
enum_ct_changelanguage (valore 3, Comando Lingua)
enum_ct_menu (valore 4, Comando Menù)
enum_ct_report (valore 5, Comando Report)
enum_ct_system (valore 6, Comando Sistema)
enum_ct_users (valore 7, Comando Utente)
enum_ct_help (valore 8, Comando Help)
enum_ct_alarm (valore 9, Comando Allarme)
enum_ct_event (valore 10, Comando Evento)

Parameter	Description
nCommandType eCommandTypes	as Tipo di comando da impostare.
nIndex as Long	Indice del comando che dovrà essere impostato con il nuovo tipo. E' la posizione nella lista del comando. L'indice parte dal valore zero.

Result Object

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
  
    Set objButtonRelease =  
        GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
  
    objCommandList.SetAtPos(0, enum_ct_variable)  
  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
  
    objCommandList.SetAtPos(0, enum_ct_variable)  
  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

SwapCommands, CommandsListCmdTarget Function

Syntax SwapCommands(_nIndex1, _nIndex2)

Description Questo metodo consente di scambiare la posizione di due comandi nella Lista Comandi referenziata.

Parameter	Description
nIndex2 as Long	Indice del secondo comando che dovrà essere scambiato. E' la posizione nella lista del comando. L'indice parte dal valore zero.
nIndex1 as Long	Indice del primo comando che dovrà essere scambiato. E' la posizione nella lista del comando. L'indice parte dal valore zero.

Result Boolean

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
  
    Set objButtonRelease =  
        GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
  
    objCommandList.SwapCommands(1,3)  
  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
  
    objCommandList.SwapCommands(1,3)  
  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

1.14.13. CommandMenuCmdTarget

Functions

GetCommandBaseInterface, CommandMenuCmdTarget Function

Syntax GetCommandBaseInterface()

Description Questa funzione consente di caricare l'interfaccia CommandBaseCmdTarget relativa al tipo di comando referenziato.

Parameter	Description
None	None

Result Object: restituisce un oggetto di tipo CommandBaseCmdTarget.

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandMenu As CommandMenuCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandMenu = objCommandList.GetCommandInterfaceAtPos(0)

    Set objCommandBase = objCommandMenu.GetCommandBaseInterface

    Set objCommandBase = Nothing
    Set objCommandMenu = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandMenu As CommandMenuCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandMenu = objCommandList.GetCommandInterfaceAtPos(0)

    Set objCommandBase = objCommandMenu.GetCommandBaseInterface

    Set objCommandBase = Nothing
    Set objCommandMenu = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

Properties

Menu, CommandMenuCmdTarget Property

Syntax Menu = _String

Description Questa proprietà imposta o restituisce il nome del Menù che dovrà attivare il Comando Menu referenziato.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges

dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandMenu As CommandMenuCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandMenu = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandMenu.Menu = "MenuTest"
    objCommandList.SaveChanges

    Set objCommandMenu = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandMenu As CommandMenuCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandMenu = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandMenu.Menu = "MenuTest"
    objCommandList.SaveChanges

    Set objCommandMenu = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

MeasureBtnText, TrendCmdTarget Property

Syntax MeasureBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando Misura dell'oggetto Data Analysis. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Refresh" per aggiornare graficamente l'oggetto.

Parameter	Description
None	None

Result String

Example:

```
Dim obj As TrendCmdTarget
Public Sub Click()
    If obj Is Nothing Then Set obj =
        GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
    If obj.MeasureBtnText <> "" Then
        obj.MeasureBtnText = ""
    Else
        obj.MeasureBtnText = "MEASURE"
    End If
    obj.Refresh
End Sub
```

XPos, CommandMenuCmdTarget Property

Syntax XPos = _Long

Description Questa proprietà imposta o restituisce la posizione orizzontale dell'origine della finestra Menù che verrà aperta tramite il Comando Menu referenziato. Il valore è espresso in pixel (con il valore "-1" verrà presa la posizione del mouse).



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandMenu As CommandMenuCmdTarget
```

```

Set objButtonRelease =
GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
Set objCommandMenu = objCommandList.GetCommandInterfaceAtPos(0)

objCommandMenu.XPos = 100
objCommandList.SaveChanges

Set objCommandMenu = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
Dim objRect As DrawCmdTarget
Dim objCommandList As CommandsListCmdTarget
Dim objCommandMenu As CommandMenuCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandMenu = objCommandList.GetCommandInterfaceAtPos(0)

objCommandMenu.XPos = 100
objCommandList.SaveChanges

Set objCommandMenu = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

YPos, CommandMenuCmdTarget Property

Syntax YPos = _Long

Description Questa proprietà imposta o restituisce la posizione verticale dell'origine della finestra Menù che verrà aperta tramite il Comando Menu referenziato. Il valore è espresso in pixel (con il valore "-1" verrà presa la posizione del mouse).



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:


```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandMenu As CommandMenuCmdTarget

    Set                                     objButtonRelease                                     =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandMenu = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandMenu.YPos = 100
    objCommandList.SaveChanges

    Set objCommandMenu = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandMenu As CommandMenuCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandMenu = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandMenu.YPos = 100
    objCommandList.SaveChanges

    Set objCommandMenu = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

1.14.14. CommandReportCmdTarget

Properties

Action, CommandReportCmdTarget Property

Syntax Action= eReportMode

Description Questa proprietà imposta o restituisce l'azione che dovrà eseguire il Comando Report/Ricette referenziato. Il tipo di azione si può specificare usando l'enumeratore eReportMode oppure inserendo il corrispondente valore numerico:

```

enum_rm_Show (valore 0, Visualizza Sincrono)
enum_rm_Print (valore 1, Stampa Sincrono)
enum_rm_ShowSafe (valore 2, Visualizza Report)
enum_rm_PrintSafe (valore 3, Stampa Report)
enum_rm_MoveFirst (valore 4, Muovi Primo)
enum_rm_MoveLast (valore 5, Muovi Ultimo)
enum_rm_MovePrev (valore 6, Muovi Precedente)
enum_rm_MoveNext (valore 7, Muovi Prossimo)
enum_rm_Activate (valore 8, Attiva)
enum_rm_Save (valore 9, Salva)
enum_rm_Delete (valore 10, Elimina)
enum_rm_Requery (valore 11, Query Filtro)
enum_rm_ExecuteQuery (valore 12, Esegue Query)
enum_rm_ExportSafe (valore 13, Esporta Report)
enum_rm_DataAnalysis (valore 14, Data Analysis)

```

enum_rm_ViewTextReport (valore 15, Visualizza Report Testuale)
enum_rm_PrintTextReport (valore 16, Stampa Report Testuale)
enum_rm_SaveTextReport (valore 17, Salva Report Testuale)
enum_rm_AppendTextReport (valore 18, Appendi Report Testuale)
enum_rm_ExportRecipe (valore 19, Esporta Ricetta)
enum_rm_ImportRecipe (valore 20, Importa Ricetta)
enum_rm_ExportAndSendMail (valore 21, Esporta e Invia Email)
enum_rm_EmbeddedRptView (valore 22, Visualizza Report Interno)
enum_rm_EmbeddedRptPrint (valore 23, Stampa Report Interno)
enum_rm_EmbeddedRptSave (valore 24, Salva Report Interno)
enum_rm_EmbeddedRptMail (valore 25, Invia Report Interno)
enum_rm_ReadRecipe (valore 26, Leggi)



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result eReportMode

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.Action = enum_rm_Show
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.Action = enum_rm_Show
    objCommandList.SaveChanges
```

```

Set objCommandReport = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

DLR, CommandReportCmdTarget Property

Syntax DLR = _String

Description Questa proprietà consente leggere o impostare il nome del DataLogger o della Ricetta per il quale eseguire il Comando Report/Ricette referenziato.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.DLR = "DataLogger1"
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

```

```

objCommandReport.DLR = "DataLogger1"
objCommandList.SaveChanges

Set objCommandReport = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

EmbeddedReportName, CommandReportCmdTarget Property

Syntax EmbeddedReportName = _String

Description Questa proprietà consente leggere o impostare il nome del "Report" di Movicon per il quale dovrà poi essere eseguito il comando selezionato nel campo "Azione". Questa proprietà sarà abilitata soltanto nel caso di utilizzo di un "Report Interno" di Movicon.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set                                objButtonRelease                                =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.EmbeddedReportName = "Report1"
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()

```

```

Dim objRect As DrawCmdTarget
Dim objCommandList As CommandsListCmdTarget
Dim objCommandReport As CommandReportCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

objCommandReport.EmbeddedReportName = "Report1"
objCommandList.SaveChanges

Set objCommandReport = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

Height, CommandReportCmdTarget Property

Syntax Height = _Long

Description Questa proprietà imposta o restituisce l'altezza della finestra di anteprima del Report. Il valore è espresso in pixel (con il valore "0" verrà presa la dimensione di default). Questo parametro verrà considerato soltanto se il report è stato creato con Crystal Report.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
Dim objButtonRelease As ButtonCmdTarget
Dim objCommandList As CommandsListCmdTarget
Dim objCommandReport As CommandReportCmdTarget

Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

objCommandReport.Height = 400
objCommandList.SaveChanges

Set objCommandReport = Nothing
Set objCommandList = Nothing

```

```

        Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.Height = 400
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

Landscape, CommandReportCmdTarget Property

Syntax Landscape = _Boolean

Description Tramite questa proprietà è possibile impostare la pagina Report con un orientamento verticale o orizzontale. Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Testuale", "Visualizza Report Interno", "Salva Report Interno", "Stampa Report Interno" o "Invia Report Interno". Impostando la proprietà a "False" la pagina avrà un orientamento verticale, impostandola a "True" avrà un orientamento orizzontale.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto. Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Boolean

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

```

```

Set objButtonRelease =
GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

objCommandReport.Landscape = True
objCommandList.SaveChanges

Set objCommandReport = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
Dim objRect As DrawCmdTarget
Dim objCommandList As CommandsListCmdTarget
Dim objCommandReport As CommandReportCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

objCommandReport.Landscape = True
objCommandList.SaveChanges

Set objCommandReport = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

PageHeight, CommandReportCmdTarget Property

Syntax PageHeight = _Long

Description Attraverso questo comando è possibile impostare l'altezza del foglio di stampa. Il valore viene impostato in millimetri e il valore -1 (valore di default) consente di utilizzare la dimensione di stampa della stampante. Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Interno".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objButtonRelease =  
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.PageHeight = 400  
    objCommandList.SaveChanges  
  
    Set objCommandReport = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.PageHeight= 400  
    objCommandList.SaveChanges  
  
    Set objCommandReport = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

PageWidth, CommandReportCmdTarget Property

Syntax PageWidth = _Long

Description Attraverso questo comando è possibile impostare la larghezza del foglio di stampa. Il valore viene impostato in millimetri e il valore -1 (valore di default) consente di utilizzare la dimensione di stampa della stampante. Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Interno".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di

configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objButtonRelease = objButtonRelease  
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.PageWidth= 200  
    objCommandList.SaveChanges  
  
    Set objCommandReport = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.PageWidth= 200  
    objCommandList.SaveChanges  
  
    Set objCommandReport = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

PrinterPort, CommandReportCmdTarget Property

Syntax PrinterPort = ePrinterPorts

Description Questa proprietà imposta o restituisce la porta di stampa per il Comando Report/Ricette referenziato. Il tipo di azione si può specificare usando l'enumeratore ePrinterPorts oppure inserendo il corrispondente valore numerico:

```
enum_port_Undefined (valore -1)  
enum_port_Infrared (valore 0)  
enum_port_COM1 (valore 1)  
enum_port_COM2 (valore 2)  
enum_port_COM3 (valore 3)
```

```

enum_port_COM4 (valore 4)
enum_port_COM5 (valore 5)
enum_port_COM6 (valore 6)
enum_port_COM7 (valore 7)
enum_port_COM8 (valore 8)
enum_port_File (valore 9)
enum_port_NetworkPrinter (valore 10)
enum_port_COM9 (valore 11)
enum_port_COM10 (valore 12)
enum_port_COM11 (valore 13)
enum_port_COM12 (valore 14)
enum_port_BluetoothBroadcom (valore 15)
enum_port_BluetoothMicrosoft (valore 16)
enum_port_LPT1 (valore 17)
enum_port_USB (valore 18)

```



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result ePrinterPorts

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.PrinterPort = enum_port_LPT1
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

```

```

objCommandReport.PrinterPort = enum_port_LPT1
objCommandList.SaveChanges

Set objCommandReport = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

PrinterName, CommandReportCmdTarget Property

Syntax PrinterName = _String

Description Tramite questo campo è possibile scegliere la stampante alla quale inviare il report. La stampante potrà essere selezionata tra le stampanti locali del PC. Se non è specificata nessuna stampante nel parametro verrà usata quella impostata come stampante di default di Windows. L'opzione "Scelta Stampante" avrà comunque la priorità su questa impostazione. Nel caso in cui il progetto è impostato per la piattaforma Windows CE, l'elenco delle stampanti è fisso e riporta quelle supportate dal tool "PrintCE.dll", ovvero le seguenti stampanti:

- HP PCL 3
- Epson ESC/P 2
- Epson Stylus COLOR
- PocketJet II
- PocketJet 200
- Canon BJ (300 dpi)
- Canon BJ (360 dpi)
- Amtech
- Epson LX (9-pin)
- Adobe PDF file
- MTE W40
- Canon IP90
- Partner M1POS
- SP-T8
- Canon IP100
- Zebra
- MP-300
- O'Neil 4 inch
- O'Neil 3 inch
- HP PCL 5e

Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Testuale" o "Stampa Report Interno".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodiché al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set                                objButtonRelease                =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.PrinterName = "Movicon PDF Writer"
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.PrinterName = "Movicon PDF Writer"
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

PortSettings, CommandReportCmdTarget Property

Syntax PortSettings = _String

Description Questa proprietà imposta o restituisce la stringa di configurazione della porta di stampa per il Comando Report/Ricette referenziato. Questa impostazione viene utilizzata solo nel caso in cui nella proprietà "PrinterPort" è stata fatta una selezione tra "File", "Stampante di Rete", "Bluetooth Broadcom" o "Bluetooth Microsoft":

File: va impostato il percorso e il nome del file che il driver della stampante deve usare per salvare l'output di stampa (es. "\\FlashDrv\Output.prn")

Stampante di Rete: va impostato il percorso di rete della stampante (es. "\\ServerName\PrinterName")

Bluetooth Broadcom: vanno impostati tre valori delimitati dal carattere pipe ('|'). Il primo valore rappresenta l'indirizzo della scheda bluetooth (es. 00:0A:D9:EB:66:C7), il secondo valore rappresenta il nome del servizio da utilizzare e il terzo valore rappresenta il numero del canale

Bluetooth Microsoft: va impostato l'indirizzo della scheda bluetooth (es. 00:0A:D9:EB:66:C7)



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.PortSettings = "00:0A:D9:EB:66:C7"  
    objCommandList.SaveChanges  
  
    Set objCommandReport = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.PortSettings = "00:0A:D9:EB:66:C7"  
    objCommandList.SaveChanges  
  
    Set objCommandReport = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

RecipeCSVSeparator, CommandReportCmdTarget Property

Syntax RecipeCSVSeparator= _Integer

Description Questa proprietà imposta o restituisce il separatore per il file ".csv" quando si utilizzano i comandi "Importa Ricetta" ed "Esporta Ricetta". Il carattere di default è il ";". Il valore numerico della proprietà corrisponde al valore decimale del carattere, ad esempio 59 per il carattere ";".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Integer

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.RecipeCSVSeparator = 59  
    objCommandList.SaveChanges  
  
    Set objCommandReport = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.RecipeCSVSeparator = 59  
    objCommandList.SaveChanges
```

```

Set objCommandReport = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

Recipient,CommandReportCmdTarget Property

Syntax Recipient = _String

Description Questa proprietà permette di impostare o restituire il nome dell'utente o del gruppo utenti a cui inviare la mail con allegato il file di report creato tramite il comando "Esporta e invia eMail" o "Invia Report Interno".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.Recipient = "Administrator"
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

```

```

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

objCommandReport.Recipient = "Administrator"
objCommandList.SaveChanges

Set objCommandReport = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

ReportExportFormat, CommandReportCmdTarget Property

Syntax ReportExportFormat = eReportExportFormat

Description Questa proprietà imposta o restituisce il formato del file sul quale esportare il Report tramite il comando "Esporta Report". Il tipo di formato si può specificare usando l'enumeratore eReportExportFormat oppure inserendo il corrispondente valore numerico:

```

enum_re_Pdf (valore 0, Pdf)
enum_re_Html (valore 1, Html)
enum_re_Txt (valore 2, Txt)
enum_re_Csv (valore 3, Csv)
enum_re_Xls (valore 4, Xls)
enum_re_Mht (valore 5, Mht)
enum_re_Rtf (valore 6, Rtf)
enum_re_Jpeg (valore 7, Jpeg)

```



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result eReportExportFormat

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease

```



```

Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

objCommandReport.ReportExportFormat = enum_re_Csv
objCommandList.SaveChanges

Set objCommandReport = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.ReportExportFormat = enum_re_Csv
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

ReportReferencePeriod, CommandReportCmdTarget Property

Syntax ReportReferencePeriod = eReportPeriod

Description Questa proprietà imposta o restituisce il periodo di riferimento che dovrà essere utilizzato per l'estrazione dei dati da visualizzare/stampare nel report. Il valore del periodo si può specificare usando l'enumeratore eReportPeriod oppure inserendo il corrispondente valore numerico:

```

enum_rp_None (valore 0, Nessuno)
enum_rp_Today (valore 1, Oggi)
enum_rp_YesterdayorToday (valore 2, Ieri e Oggi)
enum_rp_CurrentWeek (valore 3, Settimana Corrente)
enum_rp_CurrentMonth (valore 4, Mese Corrente)
enum_rp_CurrentYear (valore 5, Anno Corrente)
enum_rp_Last7days (valore 6, Ultimi 7 Giorni)
enum_rp_Last30days (valore 7, Ultimi 30 Giorni)
enum_rp_Last60Days (valore 8, Ultimi 60 Giorni)
enum_rp_Last90days (valore 9, Ultimi 90 Giorni)
enum_rp_Last1year (valore 10, Ultimo Anno)
enum_rp_Last2years (valore 11, Ultimi 2 Anni)
enum_rp_Last5years (valore 12, Ultimi 5 Anni)
enum_rp_Last10years (valore 13, Ultimi 10 Anni)

```



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di

configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result eReportPeriod

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objButtonRelease =  
        GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.ReportReferencePeriod = enum_rp_Today  
    objCommandList.SaveChanges  
  
    Set objCommandReport = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.ReportReferencePeriod = enum_rp_Today  
    objCommandList.SaveChanges  
  
    Set objCommandReport = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

ReportShowFilterByDate, CommandReportCmdTarget Property

Syntax ReportShowFilterByDate= _Boolean

Description Questa proprietà, se impostata a True, visualizzerà all'apertura del Report una finestra di dialogo tramite la quale l'operatore potrà inserire la data e ora per la query di selezione dei dati da visualizzare sul Report.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.ReportShowFilterByDate = True
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.ReportShowFilterByDate = True
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

ReportShowToolbar, CommandReportCmdTarget Property

Syntax ReportShowToolbar = _Boolean

Description Questa proprietà permette di visualizzare o nascondere la barra degli strumenti nella finestra di anteprima del report. La proprietà verrà gestita soltanto se il report è stato creato con Crystal Report.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.ReportShowToolbar = True
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.ReportShowToolbar = True
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

ReportShowTree, CommandReportCmdTarget Property

Syntax ReportShowTree = _Boolean

Description Questa proprietà permette di visualizzare o nascondere la struttura dei gruppi nella finestra di anteprima del report. La proprietà verrà gestita soltanto se il report è stato creato con Crystal Report.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.ReportShowTree = True  
    objCommandList.SaveChanges  
  
    Set objCommandReport = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.ReportShowTree = True  
    objCommandList.SaveChanges  
  
    Set objCommandReport = Nothing
```

```

Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

ShowPrintDialog, CommandReportCmdTarget Property

Syntax ShowPrintDialog = _Boolean

Description Abilitando questa opzione, prima della stampa del report, verrà aperta la finestra di dialogo per la selezione della stampante da utilizzare. Sarà quindi possibile scegliere quale stampante utilizzare tra quelle installate nel PC. Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Testuale" o "Stampa Report Interno".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Boolean

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objButtonRelease = objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.ShowPrintDialog = True
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

```

```

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

objCommandReport.ShowPrintDialog = True
objCommandList.SaveChanges

Set objCommandReport = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

TextualRptBottomMargin, CommandReportCmdTarget Property

Syntax TextualRptBottomMargin = _Long

Description Questa proprietà permette impostare o restituire il margine di stampa basso. Il margine viene impostato in millimetri e il valore -1 (valore di default) consente di utilizzare l'eventuale margine di stampa di default recuperato attraverso il driver della stampante utilizzata. Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Testuale".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.TextualRptBottomMargin = 10
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

End Sub

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.TextualRptBottomMargin = 10  
    objCommandList.SaveChanges  
  
    Set objCommandReport = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

TextualRptLeftMargin, CommandReportCmdTarget Property

Syntax TextualRptLeftMargin = _Long

Description Questa proprietà permette impostare o restituire il margine di stampa sinistro. Il margine viene impostato in millimetri e il valore -1 (valore di default) consente di utilizzare l'eventuale margine di stampa di default recuperato attraverso il driver della stampante utilizzata.
Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Testuale".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget
```



```

Set objButtonRelease =
GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

objCommandReport.TextualRptLeftMargin = 10
objCommandList.SaveChanges

Set objCommandReport = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
Dim objRect As DrawCmdTarget
Dim objCommandList As CommandsListCmdTarget
Dim objCommandReport As CommandReportCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

objCommandReport.TextualRptLeftMargin = 10
objCommandList.SaveChanges

Set objCommandReport = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

TextualRptMaxPages, CommandReportCmdTarget Property

Syntax TextualRptMaxPages = _Long

Description Questa proprietà permette impostare o restituire il numero massimo di pagine stampate con un singolo comando "Visualizza Report Testuale", "Stampa Report Testuale", "Salva Report Testuale" e "Appendi Report Testuale". Impostando il valore su "0", il numero di pagine stampabili è illimitato (si consiglia di non impostare "0" per evitare di utilizzare troppa memoria o impegnare troppo la stampante in caso di errori nella formulazione della query di estrazione dei dati).



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.TextualRptMaxPages = 10  
    objCommandList.SaveChanges  
  
    Set objCommandReport = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.TextualRptMaxPages = 10  
    objCommandList.SaveChanges  
  
    Set objCommandReport = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

TextualRptOutputFile, CommandReportCmdTarget Property

Syntax TextualRptOutputFile = _String

Description Questa proprietà permette impostare o restituire il nome del file con cui salvare il report testuale. Il parametro è richiesto solo dai comandi "Salva Report testuale" e "Appendi Report Testuale". Se non viene specificato il percorso il file verrà creato nella cartella "DLOGGERS" del progetto.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set                                objButtonRelease                =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.TextualRptOutputFile = "TestReport.rtf"
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.TextualRptOutputFile = "TestReport.rtf"
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

TextualRptRightMargin, CommandReportCmdTarget Property

Syntax TextualRptRightMargin = _Long

Description Questa proprietà permette impostare o restituire il margine di stampa destro. Il margine viene impostato in millimetri e il valore -1 (valore di default) consente di utilizzare l'eventuale margine di stampa di default recuperato attraverso il driver della stampante utilizzata. Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Testuale".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.TextualRptRightMargin = 10
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.TextualRptRightMargin = 10
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

TextualRptSQLQuery, CommandReportCmdTarget Property

Syntax TextualRptSQLQuery = _String

Description Questa proprietà permette di impostare o restituire la query da utilizzare per estrarre i dati del DataLogger o Ricetta selezionato di un "Report Testuale" o di un "Report Interno".
In questo campo va inserito il testo della query che dovrà essere coerente con il database utilizzato e non è possibile inserire il nome di una variabile per rendere dinamica la query.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.
Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget

    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.TextualRptSQLQuery = "Select * From DataLogger1"
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.TextualRptSQLQuery = "Select * From DataLogger1"
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

TextualRptTemplateFile, CommandReportCmdTarget Property

Syntax TextualRptTemplateFile = _String

Description Questa proprietà permette impostare o restituire il nome del file di Layout del report testuale che, oltre alla formattazione, contiene anche i campi speciali da utilizzare. Se non viene specificato il percorso il file verrà cercato nella cartella Risorse del progetto.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.TextualRptTemplateFile = "TemplateRpt.rtf"  
    objCommandList.SaveChanges  
  
    Set objCommandReport = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)
```

```

objCommandReport.TextualRptTemplateFile = "TemplateRpt.rtf"
objCommandList.SaveChanges

Set objCommandReport = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

TextualRptTopMargin, CommandReportCmdTarget Property

Syntax TextualRptTopMargin = _Long

Description Questa proprietà permette impostare o restituire il margine di stampa superiore. Il margine viene impostato in millimetri e il valore -1 (valore di default) consente di utilizzare l'eventuale margine di stampa di default recuperato attraverso il driver della stampante utilizzata. Tale parametro verrà considerato soltanto se nel campo "Azione" è stato selezionato il comando "Stampa Report Testuale".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReportAs CommandReportCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandReport= objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.TextualRptTopMargin = 10
    objCommandList.SaveChanges

    Set objCommandReport= Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReportAs CommandReportCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandReport= objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.TextualRptTopMargin = 10  
    objCommandList.SaveChanges  
  
    Set objCommandReport= Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

Syntax Width = _Long

Description	Questa proprietà imposta o restituisce la larghezza della finestra di anteprima del Report. Il valore è espresso in pixel (con il valore "0" verrà presa la dimensione di default). Questo parametro verrà considerato soltanto se il report è stato creato con Crystal Report.
--------------------	---



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result	Long
--------	------

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandReport As CommandReportCmdTarget  
  
    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandReport.Width = 600  
    objCommandList.SaveChanges
```



```

        Set objCommandReport = Nothing
        Set objCommandList = Nothing
        Set objButtonRelease = Nothing
    End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.Width = 600
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

XPos, CommandReportCmdTarget Property

Syntax XPos = _Long

Description Questa proprietà imposta o restituisce la posizione orizzontale dell'origine della finestra di anteprima del Report. Il valore è espresso in pixel (con il valore "-1" verrà presa la posizione di default). Questo parametro verrà considerato soltanto se il report è stato creato con Crystal Report.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

```

```

Set                                objButtonRelease                                =
GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

objCommandReport.XPos = 50
objCommandList.SaveChanges

Set objCommandReport = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.XPos = 50
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

YPos, CommandReportCmdTarget Property

Syntax YPos = _Long

Description Questa proprietà imposta o restituisce la posizione verticale dell'origine della finestra di anteprima del Report. Il valore è espresso in pixel (con il valore "-1" verrà presa la posizione di default). Questo parametro verrà considerato soltanto se il report è stato creato con Crystal Report.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objButtonRelease = objButtonRelease
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.YPos = 50
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandReport.YPos = 50
    objCommandList.SaveChanges

    Set objCommandReport = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

Functions

GetCommandBaseInterface, CommandReportCmdTarget Function

Syntax GetCommandBaseInterface()

Description Questa funzione consente di caricare l'interfaccia CommandBaseCmdTarget relativa al tipo di comando referenziato.

Parameter	Description
None	None

Result Object: restituisce un oggetto di tipo CommandBaseCmdTarget.

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandReport As CommandReportCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

```

```

Set objButtonRelease =
GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

Set objCommandBase = objCommandReport.GetCommandBaseInterface

Set objCommandBase = Nothing
Set objCommandReport = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
Dim objRect As DrawCmdTarget
Dim objCommandList As CommandsListCmdTarget
Dim objCommandReport As CommandReportCmdTarget
Dim objCommandBase As CommandBaseCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandReport = objCommandList.GetCommandInterfaceAtPos(0)

Set objCommandBase = objCommandReport.GetCommandBaseInterface

Set objCommandBase = Nothing
Set objCommandReport = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

1.14.15. CommandScriptCmdTarget

Properties

Action, CommandScriptCmdTarget Property

Syntax Action= eRunScriptMode

Description Questa proprietà imposta o restituisce l'azione che dovrà eseguire il Comando Script referenziato. Il tipo di azione si può specificare usando l'enumeratore eRunScriptMode oppure inserendo il corrispondente valore numerico:

```

enum_ops_runnormal (valore 0, Avvia Normale)
enum_ops_runandwait (valore 1, Avvia Sincronizzato)
enum_ops_runsafe (valore 2, Avvia in un processo separato (Safe Mode))
enum_ops_stop (valore 3, Stop)
enum_ops_unload (valore 4, Scarica)

```



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di

configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result eRunScriptMode

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandScript As CommandScriptCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandScript = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandScript.Action = enum_ops_runnormal
    objCommandList.SaveChanges

    Set objCommandScript = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandScript As CommandScriptCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandScript = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandScript.Action = enum_ops_runnormal
    objCommandList.SaveChanges

    Set objCommandScript = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

MoreInstanceAllowed, CommandScriptCmdTarget Property

Syntax MoreInstanceAllowed= _Boolean

Description Questa proprietà, se impostata a True, consente di poter eseguire più istanze contemporaneamente dello stesso Basic Script. Per avere effetto però deve essere stato specificato un numero maggiore di uno nella proprietà "Numero Massimo Istanze" del Basic Script.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges

dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandScript As CommandScriptCmdTarget  
  
    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandScript = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandScript.MoreInstanceAllowed = False  
    objCommandList.SaveChanges  
  
    Set objCommandScript = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandScript As CommandScriptCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandScript = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandScript.MoreInstanceAllowed = False  
    objCommandList.SaveChanges  
  
    Set objCommandScript = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

Parameters, CommandScriptCmdTarget Property

Syntax Parameters = _String

Description Questa proprietà imposta o restituisce la lista dei parametri da passare allo script quando questo viene richiamato dal Comando Script referenziato. I parametri vengono sempre letti come stringhe e devono essere separati dalla virgola (",").



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandScript As CommandScriptCmdTarget  
  
    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandScript = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandScript.Parameters = "Param1,Param2"  
    objCommandList.SaveChanges  
  
    Set objCommandScript = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandScript As CommandScriptCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandScript = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandScript.Parameters = "Param1,Param2"  
    objCommandList.SaveChanges  
  
    Set objCommandScript = Nothing
```

```

Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

Script, CommandScriptCmdTarget Property

Syntax Script = _String

Description Questa proprietà imposta o restituisce il nome del Basic Script che dovrà essere eseguito o arrestato dal Comando Script referenziato.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Integer

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandScript As CommandScriptCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandScript = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandScript.Script = "Script1"
    objCommandList.SaveChanges

    Set objCommandScript = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandScript As CommandScriptCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandScript = objCommandList.GetCommandInterfaceAtPos(0)

```



```

objCommandScript.Script = "Script1"
objCommandList.SaveChanges

Set objCommandScript = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

SynchroTimeout, CommandScriptCmdTarget Property

Syntax SynchroTimeout= _Long

Description Questa proprietà imposta o restituisce il tempo di timeout massimo da considerare quando il Basic Script viene eseguito in modo Sincronizzato dal Comando Script referenziato.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandScript As CommandScriptCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandScript = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandScript.SynchroTimeout = 1000
    objCommandList.SaveChanges

    Set objCommandScript = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget

```

```

Dim objCommandList As CommandsListCmdTarget
Dim objCommandScript As CommandScriptCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandScript = objCommandList.GetCommandInterfaceAtPos(0)

objCommandScript.SynchroTimeout = 1000
objCommandList.SaveChanges

Set objCommandScript = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

Functions

GetCommandBaseInterface, CommandScriptCmdTarget Function

Syntax GetCommandBaseInterface()

Description Questa funzione consente di caricare l'interfaccia CommandBaseCmdTarget relativa al tipo di comando referenziato.

Parameter	Description
None	None

Result Object: restituisce un oggetto di tipo CommandBaseCmdTarget.

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandScript As CommandScriptCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set                                objButtonRelease                                =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandScript = objCommandList.GetCommandInterfaceAtPos(0)

    Set objCommandBase = objCommandScript.GetCommandBaseInterface

    Set objCommandBase = Nothing
    Set objCommandScript = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandScript As CommandScriptCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")

```

```

Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandScript = objCommandList.GetCommandInterfaceAtPos(0)

Set objCommandBase = objCommandScript.GetCommandBaseInterface

Set objCommandBase = Nothing
Set objCommandScript = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

1.14.16. CommandSynopticCmdTarget

Properties

Action, CommandSynopticCmdTarget Property

Syntax Action= eOpenSynopticMode

Description Questa proprietà imposta o restituisce l'azione che dovrà eseguire il Comando Sinottico referenziato. Il tipo di azione si può specificare usando l'enumeratore eOpenSynopticMode oppure inserendo il corrispondente valore numerico:

```

enum_ops_opennormal (valore 0, Apri Normale (cambio pagina))
enum_ops_openmodal (valore 1, Apri Modale (pop-up))
enum_ops_openframe (valore 2, Apri Frame (multi-monitor))
enum_ops_opensafe (valore 3, Apri in un processo separato (Safe Mode))
enum_ops_print (valore 4, Stampa)
enum_ops_close (valore 5, Chiudi e ritorna a precedente)
enum_ops_executesynapse (valore 6, Esegui Sinapsi)
enum_ops_opennext (valore 7, Apri Successivo (Cambio pagina ID successivo))
enum_ops_openprev (valore 8, Apri Precedente (Cambio pagina ID precedente))
enum_ops_captureprint (valore 9, Cattura e Stampa)
enum_ops_capturesave (valore 10, Cattura e Salva)

```



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodiché al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result eOpenSynopticMode

Example1:

```
Public Sub Click()
```

```

Dim objButtonRelease As ButtonCmdTarget
Dim objCommandList As CommandsListCmdTarget
Dim objCommandSynoptic As CommandSynopticCmdTarget

Set                                objButtonRelease                                =
GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

objCommandSynoptic.Action = enum_ops_opennormal
objCommandList.SaveChanges

Set objCommandSynoptic = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
Dim objRect As DrawCmdTarget
Dim objCommandList As CommandsListCmdTarget
Dim objCommandSynoptic As CommandSynopticCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

objCommandSynoptic.Action = enum_ops_opennormal
objCommandList.SaveChanges

Set objCommandSynoptic = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

Height, CommandSynopticCmdTarget Property

Syntax Height = _Long

Description Questa proprietà imposta o restituisce l'altezza della finestra aperta tramite il Comando Sinottico referenziato. Il valore è espresso in pixel (con il valore "0" verrà presa la dimensione di default). Questo parametro verrà considerato soltanto se il comando utilizzato è "Apri Modale" o "Apri Frame".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandSynoptic As CommandSynopticCmdTarget  
  
    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandSynoptic.Height = 400  
    objCommandList.SaveChanges  
  
    Set objCommandSynoptic = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandSynoptic As CommandSynopticCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandSynoptic.Height = 400  
    objCommandList.SaveChanges  
  
    Set objCommandSynoptic = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

KeepproportionsOnPrint, CommandSynopticCmdTarget Property

Syntax KeepproportionsOnPrint = _Boolean

Description Questa proprietà, se impostata a True, permette di stampare mantenendo le proporzioni che si vedono sullo schermo, mentre senza questa opzione (quindi di default) vengono adattate sia l'altezza che la larghezza all'intera dimensione della pagina. Questo parametro ha effetto sia sul comando "Stampa" che "Cattura e Stampa".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di

configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result **Boolean**

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandSynoptic As CommandSynopticCmdTarget  
  
    Set objButtonRelease = objButtonRelease.GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandSynoptic.KeepproportionsOnPrint = False  
    objCommandList.SaveChanges  
  
    Set objCommandSynoptic = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandSynoptic As CommandSynopticCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandSynoptic.KeepproportionsOnPrint = False  
    objCommandList.SaveChanges  
  
    Set objCommandSynoptic = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

Monitor, CommandSynopticCmdTarget Property

Syntax Monitor = _Long

Description Questa proprietà imposta o restituisce il numero del Monitor sul quale il Sinottico dovrà essere aperto dal Comando Sinottico referenziato. Questo parametro verrà considerato soltanto se il comando utilizzato è "Apri Frame".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result **Boolean**

Example1:

```

Public Sub Click()
Dim objButtonRelease As ButtonCmdTarget
Dim objCommandList As CommandsListCmdTarget
Dim objCommandSynoptic As CommandSynopticCmdTarget

Set                               objButtonRelease                =
GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

objCommandSynoptic. Monitor = 2
objCommandList.SaveChanges

Set objCommandSynoptic = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
Dim objRect As DrawCmdTarget
Dim objCommandList As CommandsListCmdTarget
Dim objCommandSynoptic As CommandSynopticCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

objCommandSynoptic. Monitor = 2
objCommandList.SaveChanges

Set objCommandSynoptic = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

ParameterFile, CommandSynopticCmdTarget Property

Syntax ParameterFile = _String

Description Questa proprietà imposta o restituisce il nome, ed eventualmente il percorso, del file di parametrizzazione, nel caso si voglia aprire il sinottico passandogli dei parametri.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Integer

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandSynoptic As CommandSynopticCmdTarget  
  
    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandSynoptic.ParameterFile = "Param1.movpar"  
    objCommandList.SaveChanges  
  
    Set objCommandSynoptic = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandSynoptic As CommandSynopticCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandSynoptic.ParameterFile = "Param1.movpar"  
    objCommandList.SaveChanges  
  
    Set objCommandSynoptic = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
    objRect = Nothing  
End Sub
```


PrintBottomMargin, CommandSynopticCmdTarget Property

Syntax PrintBottomMargin = _Long

Description Questa proprietà imposta o restituisce la dimensione del margine basso del foglio di stampa per il Comando Sinottico referenziato. Il valore è espresso in mm. Impostando "-1" verrà utilizzata la dimensione di default del foglio di stampa.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodiché al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandSynoptic As CommandSynopticCmdTarget  
  
    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandSynoptic.PrintBottomMargin = 10  
    objCommandList.SaveChanges()  
  
    Set objCommandSynoptic = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandSynoptic As CommandSynopticCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandSynoptic.PrintBottomMargin = 10  
    objCommandList.SaveChanges()  
  
    Set objCommandSynoptic = Nothing
```

```

Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

PrintLeftMargin, CommandSynopticCmdTarget Property

Syntax PrintLeftMargin = _Long

Description Questa proprietà imposta o restituisce la dimensione del margine sinistro del foglio di stampa per il Comando Sinottico referenziato. Il valore è espresso in mm. Impostando "-1" verrà utilizzata la dimensione di default del foglio di stampa.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.PrintLeftMargin = 10
    objCommandList.SaveChanges()

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")

```

```

Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

objCommandSynoptic.PrintLeftMargin = 10
objCommandList.SaveChanges()

Set objCommandSynoptic = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

PrintPageHeight, CommandSynopticCmdTarget Property

Syntax PrintPageHeight = _Long

Description Questa proprietà imposta o restituisce l'altezza del foglio di stampa per il Comando Sinottico referenziato. Il valore è espresso in mm. Impostando "-1" verrà utilizzata la dimensione di default del foglio di stampa.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodiché al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objButtonRelease = objButtonRelease
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface =
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.PrintPageHeight = 200
    objCommandList.SaveChanges()

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.PrintPageHeight = 200
    objCommandList.SaveChanges()

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

PrintPageWidth, CommandSynopticCmdTarget Property

Syntax PrintPageWidth = _Long

Description Questa proprietà imposta o restituisce la larghezza del foglio di stampa per il Comando Sinottico referenziato. Il valore è espresso in mm. Impostando "-1" verrà utilizzata la dimensione di default del foglio di stampa.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.PrintPageWidth = 250
    objCommandList.SaveChanges()

```

```

        Set objCommandSynoptic = Nothing
        Set objCommandList = Nothing
        Set objButtonRelease = Nothing
    End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.PrintPageWidth = 250
    objCommandList.SaveChanges()

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

PrintRightMargin, CommandSynopticCmdTarget Property

Syntax PrintRightMargin = _Long

Description Questa proprietà imposta o restituisce la dimensione del margine destro del foglio di stampa per il Comando Sinottico referenziato. Il valore è espresso in mm. Impostando "-1" verrà utilizzata la dimensione di default del foglio di stampa.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

```

```

Set objButtonRelease =
GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

objCommandSynoptic.PrintRightMargin = 10
objCommandList.SaveChanges()

Set objCommandSynoptic = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
Dim objRect As DrawCmdTarget
Dim objCommandList As CommandsListCmdTarget
Dim objCommandSynoptic As CommandSynopticCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

objCommandSynoptic.PrintRightMargin = 10
objCommandList.SaveChanges()

Set objCommandSynoptic = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

PrintTopMargin, CommandSynopticCmdTarget Property

Syntax PrintTopMargin = _Long

Description Questa proprietà imposta o restituisce la dimensione del margine alto del foglio di stampa per il Comando Sinottico referenziato. Il valore è espresso in mm. Impostando "-1" verrà utilizzata la dimensione di default del foglio di stampa.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandSynoptic As CommandSynopticCmdTarget  
  
    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandSynoptic.PrintTopMargin = 10  
    objCommandList.SaveChanges()  
  
    Set objCommandSynoptic = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandSynoptic As CommandSynopticCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandSynoptic.PrintTopMargin = 10  
    objCommandList.SaveChanges()  
  
    Set objCommandSynoptic = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

ResizableBorder, CommandSynopticCmdTarget Property

Syntax ResizableBorder = _Boolean

Description Questa proprietà, se impostata a True, consente di ridimensionare la finestra Sinottico aperta dal Comando Sinottico referenziato. Questo parametro verrà considerato soltanto se il comando utilizzato è "Apri Modale" o "Apri Frame".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
-----------	-------------

None	None
------	------

Result Boolean

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.ResizeableBorder = False
    objCommandList.SaveChanges

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.ResizeableBorder = False
    objCommandList.SaveChanges

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

ShowBorder, CommandSynopticCmdTarget Property

Syntax ShowBorder = _Boolean

Description Questa proprietà, se impostata a True, consente di evidenziare maggiormente il bordo esterno della finestra Sinottico aperta dal Comando Sinottico referenziato. Questo parametro verrà considerato soltanto se il comando utilizzato è "Apri Modale" o "Apri Frame".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto. Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodiché al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di

configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objButtonRelease = objButtonRelease
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.ShowBorder = False
    objCommandList.SaveChanges

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.ShowBorder= False
    objCommandList.SaveChanges

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

ShowCaption, CommandSynopticCmdTarget Property

Syntax ShowCaption = _Boolean

Description Questa proprietà, se impostata a True, consente di visualizzare la barra del titolo della finestra Sinottico aperta dal Comando Sinottico referenziato. Questo parametro verrà considerato soltanto se il comando utilizzato è "Apri Modale" o "Apri Frame".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.ShowCaption = False
    objCommandList.SaveChanges

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.ShowCaption = False
    objCommandList.SaveChanges

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

ShowMaximizedBtn, CommandSynopticCmdTarget Property

Syntax ShowMaximizedBtn = _Boolean

Description Questa proprietà, se impostata a True, consente di visualizzare il pulsante per massimizzare la finestra Sinottico aperta dal Comando Sinottico referenziato. Questo parametro verrà considerato soltanto se il comando utilizzato è "Apri Modale" o "Apri Frame". Se la proprietà "Menù di Sistema" non è stata abilitata questa impostazione non avrà alcun effetto.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.ShowMaximizedBtn = False
    objCommandList.SaveChanges

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.ShowMaximizedBtn = False
    objCommandList.SaveChanges

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

ShowMinimizedBtn, CommandSynopticCmdTarget Property

Syntax ShowMinimizedBtn = _Boolean

Description Questa proprietà, se impostata a True, consente di visualizzare il pulsante per minimizzare la finestra Sinottico aperta dal Comando Sinottico referenziato. Questo parametro verrà considerato soltanto se il comando utilizzato è "Apri Modale" o "Apri Frame". Se la proprietà "Menù di Sistema" non è stata abilitata questa impostazione non avrà alcun effetto.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandSynoptic As CommandSynopticCmdTarget  
  
    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandSynoptic.ShowMinimizedBtn = False  
    objCommandList.SaveChanges  
  
    Set objCommandSynoptic = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandSynoptic As CommandSynopticCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandSynoptic.ShowMinimizedBtn = False  
    objCommandList.SaveChanges
```

```

Set objCommandSynoptic = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

ShowSystemMenu, CommandSynopticCmdTarget Property

Syntax ShowSystemMenu= _Boolean

Description Questa proprietà, se impostata a True, consente di visualizzare il Menù di Sistema sulla barra del titolo della finestra Sinottico aperta dal Comando Sinottico referenziato. Questo parametro verrà considerato soltanto se il comando utilizzato è "Apri Modale" o "Apri Frame". Se la barra del "Titolo" non è stata abilitata questa impostazione non avrà alcun effetto.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Boolean

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objButtonRelease = objButtonRelease
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.ShowSystemMenu = False
    objCommandList.SaveChanges

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget

```

```

Dim objCommandList As CommandsListCmdTarget
Dim objCommandSynoptic As CommandSynopticCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

objCommandSynoptic.ShowSystemMenu = False
objCommandList.SaveChanges

Set objCommandSynoptic = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

SynopticName, CommandSynopticCmdTarget Property

Syntax SynopticName = _String

Description Questa proprietà imposta o restituisce il nome del Sinottico che si vuole aprire tramite il Comando Sinottico referenziato.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.SynopticName = "Screen1"
    objCommandList.SaveChanges

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing

```

End Sub

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandSynoptic As CommandSynopticCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandSynoptic.SynopticName = "Screen1"  
    objCommandList.SaveChanges  
  
    Set objCommandSynoptic = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

Width, CommandSynopticCmdTarget Property

Syntax Width = _Long

Description Questa proprietà imposta o restituisce la larghezza della finestra aperta tramite il Comando Sinottico referenziato. Il valore è espresso in pixel (con il valore "0" verrà presa la dimensione di default). Questo parametro verrà considerato soltanto se il comando utilizzato è "Apri Modale" o "Apri Frame".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandSynoptic As CommandSynopticCmdTarget  
  
    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)
```

```

objCommandSynoptic.Width = 600
objCommandList.SaveChanges

Set objCommandSynoptic = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

Example2:

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.Width = 600
    objCommandList.SaveChanges

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

XPos, CommandSynopticCmdTarget Property

Syntax XPos = _Long

Description Questa proprietà imposta o restituisce la posizione orizzontale dell'origine della finestra aperta tramite il Comando Sinottico referenziato. Il valore è espresso in pixel (con il valore "-1" verrà presa la posizione di default). Questo parametro verrà considerato soltanto se il comando utilizzato è "Apri Modale" o "Apri Frame".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

```



```

Set objButtonRelease =
GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

objCommandSynoptic.XPos = 50

Set objCommandSynoptic = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
Dim objRect As DrawCmdTarget
Dim objCommandList As CommandsListCmdTarget
Dim objCommandSynoptic As CommandSynopticCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

objCommandSynoptic.XPos = 50
objCommandList.SaveChanges

Set objCommandSynoptic = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

YPos, CommandSynopticCmdTarget Property

Syntax YPos = _Long

Description Questa proprietà imposta o restituisce la posizione verticale dell'origine della finestra aperta tramite il Comando Sinottico referenziato. Il valore è espresso in pixel (con il valore "-1" verrà presa la posizione di default). Questo parametro verrà considerato soltanto se il comando utilizzato è "Apri Modale" o "Apri Frame".



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set                                objButtonRelease                                =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.YPos = 50
    objCommandList.SaveChanges

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSynoptic.YPos = 50
    objCommandList.SaveChanges

    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

Functions

GetCommandBaseInterface, CommandSynopticCmdTarget Function

Syntax GetCommandBaseInterface()

Description Questa funzione consente di caricare l'interfaccia CommandBaseCmdTarget relativa al tipo di comando referenziato.

Parameter	Description
None	None

Result Object: restituisce un oggetto di tipo CommandBaseCmdTarget.

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set                                objButtonRelease                                =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface

```

```

Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

Set objCommandBase = objCommandSynoptic.GetCommandBaseInterface

Set objCommandBase = Nothing
Set objCommandSynoptic = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSynoptic As CommandSynopticCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandSynoptic = objCommandList.GetCommandInterfaceAtPos(0)

    Set objCommandBase = objCommandSynoptic.GetCommandBaseInterface

    Set objCommandBase = Nothing
    Set objCommandSynoptic = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub

```

1.14.17. CommandSystemCmdTarget

Properties

Action, CommandSystemCmdTarget Property

Syntax Action= eSystemMode

Description Questa proprietà imposta o restituisce l'azione che dovrà eseguire il Comando Sistema referenziato. Il tipo di azione si può specificare usando l'enumeratore eSystemMode oppure inserendo il corrispondente valore numerico:

```

enum_sm_exitWindow (valore 0, Arresta Sistema Operativo)
enum_sm_exitMovicon (valore 1, Arresta Applicazione)
enum_sm_RunExec (valore 2, Esegui Applicazione)
enum_sm_RunExecWait (valore 3, Esegui Applicazione e Attendi)
enum_sm_PlaySoundFile (valore 4, Esegui File Audio)
enum_sm_Beep (valore 5, Esegui Beep)
enum_sm_Speak (valore 6, Speech)
enum_sm_rebootWindow (valore 7, Riavvio Sistema)
enum_sm_ShowHideTraceBar (valore 8, Visualizza/Nascondi Finestra di Output)
enum_sm_WaitTime (valore 9, Tempo di Attesa)

```



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodiché al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di

configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result eSystemMode

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSystem As CommandSystemCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandSystem = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSystem.Action = enum_sm_exitMovicon
    objCommandList.SaveChanges

    Set objCommandSystem = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSystem As CommandSystemCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandSystem = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSystem.Action = enum_sm_exitMovicon
    objCommandList.SaveChanges

    Set objCommandSystem = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

CommandLine, CommandSystemCmdTarget Property

Syntax CommandLine = _String

Description Questa proprietà imposta o restituisce la stringa di comando da utilizzare per il tipo di "Azione" eseguita dal Comando Sistema referenziato.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSystem As CommandSystemCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandSystem = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSystem.CommandLine = "Calc.exe"
    objCommandList.SaveChanges

    Set objCommandSystem = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSystem As CommandSystemCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandSystem = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSystem.CommandLine = "Calc.exe"
    objCommandList.SaveChanges

    Set objCommandSystem = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

Timeout, CommandSystemCmdTarget Property

Syntax Timeout = _Long

Description Questa proprietà imposta o restituisce il tempo di timeout (espresso in millisecondi) che Movicon attenderà nel caso l'applicazione eseguita tramite il Comando Sinottico referenziato non risponda.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSystem As CommandSystemCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandSystem = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSystem.Timeout = 1000
    objCommandList.SaveChanges

    Set objCommandSystem = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSystem As CommandSystemCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandSystem = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandSystem.Timeout = 1000
    objCommandList.SaveChanges

    Set objCommandSystem = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

WorkingPath, CommandSystemCmdTarget Property

Syntax WorkingPath = _String

Description Questa proprietà imposta o restituisce la cartella di lavoro da utilizzare per il parametro "Cartella di Lavoro" per il Comando Sistema referenziato.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandSystem As CommandSystemCmdTarget  
  
    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandSystem = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandSystem.WorkingPath = "C:\Temp\  
    objCommandList.SaveChanges  
  
    Set objCommandSystem = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandSystem As CommandSystemCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandSystem = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandSystem.WorkingPath = "C:\Temp\  
    objCommandList.SaveChanges  
  
    Set objCommandSystem = Nothing  
    Set objCommandList = Nothing
```

```
        Set objRect = Nothing
End Sub
```

Functions

GetCommandBaseInterface, CommandSystemCmdTarget Function

Syntax GetCommandBaseInterface()

Description Questa funzione consente di caricare l'interfaccia CommandBaseCmdTarget relativa al tipo di comando referenziato.

Parameter	Description
None	None

Result Object: restituisce un oggetto di tipo CommandBaseCmdTarget.

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSystem As CommandSystemCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objButtonRelease = GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandSystem = objCommandList.GetCommandInterfaceAtPos(0)

    Set objCommandBase = objCommandSystem.GetCommandBaseInterface

    Set objCommandBase = Nothing
    Set objCommandSystem = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandSystem As CommandSystemCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandSystem = objCommandList.GetCommandInterfaceAtPos(0)

    Set objCommandBase = objCommandSystem.GetCommandBaseInterface

    Set objCommandBase = Nothing
    Set objCommandSystem = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```


1.14.18. CommandUsersCmdTarget

Properties

Action, CommandUsersCmdTarget Property

Syntax Action= eUserMode

Description Questa proprietà imposta o restituisce l'azione che dovrà eseguire il Comando Utenti referenziato. Il tipo di azione si può specificare usando l'enumeratore eUserMode oppure inserendo il corrispondente valore numerico:

enum_um_Logon (valore 0, Log on)
enum_um_Logoff (valore 1, Log off)
enum_um_EditUsers (valore 2, Edita Lista Utenti)



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result eUserMode

Example:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandUser As CommandUsersCmdTarget  
  
    Set objButtonRelease =  
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandUser = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandUser.Action = enum_um_Logon  
    objCommandList.SaveChanges  
  
    Set objCommandUser = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandUser As CommandUsersCmdTarget
```

```

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandUser = objCommandList.GetCommandInterfaceAtPos(0)

objCommandUser.Action = enum_um_Logon
objCommandList.SaveChanges

Set objCommandUser = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

Level, CommandUsersCmdTarget Property

Syntax Level = _Long

Description Questa proprietà imposta o restituisce il livello minimo che deve avere l'utente che esegue il Logon tramite il Comando Utenti referenziato.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandUser As CommandUsersCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandUser = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandUser.Level = 5
    objCommandList.SaveChanges

    Set objCommandUser = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget

```

```

Dim objCommandUser As CommandUsersCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandUser = objCommandList.GetCommandInterfaceAtPos(0)

objCommandUser.Level = 5
objCommandList.SaveChanges

Set objCommandUser = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

Functions

GetCommandBaseInterface, CommandUsersCmdTarget Function

Syntax GetCommandBaseInterface()

Description Questa funzione consente di caricare l'interfaccia CommandBaseCmdTarget relativa al tipo di comando referenziato.

Parameter	Description
None	None

Result Object: restituisce un oggetto di tipo CommandBaseCmdTarget.

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandUser As CommandUsersCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandUser = objCommandList.GetCommandInterfaceAtPos(0)

    Set objCommandBase = objCommandUser.GetCommandBaseInterface

    Set objCommandBase = Nothing
    Set objCommandUser = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = NothingEnd Sub

```

Example2:

```

Public Sub Click()

    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandUser As CommandUsersCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandUser = objCommandList.GetCommandInterfaceAtPos(0)

    Set objCommandBase = objCommandUser.GetCommandBaseInterface

```

```

        Set objCommandBase = Nothing
        Set objCommandUser = Nothing
        Set objCommandList = Nothing
        Set objRect = Nothing
    End Sub

```

1.14.19. CommandVariableCmdTarget

Functions

GetCommandBaseInterface, CommandVariableCmdTarget Function

Syntax GetCommandBaseInterface()

Description Questa funzione consente di caricare l'interfaccia CommandBaseCmdTarget relativa al tipo di comando referenziato.

Parameter	Description
None	None

Result Object: restituisce un oggetto di tipo CommandBaseCmdTarget.

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandVariable As CommandVariableCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)

    Set objCommandBase = objCommandVariable.GetCommandBaseInterface

    Set objCommandBase = Nothing
    Set objCommandVariable = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandVariable As CommandVariableCmdTarget
    Dim objCommandBase As CommandBaseCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)

    Set objCommandBase = objCommandVariable.GetCommandBaseInterface

```

```

Set objCommandBase = Nothing
Set objCommandVariable = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

Properties

Action, CommandVariableCmdTarget Property

Syntax Action= eSetVariableMode

Description Questa proprietà imposta o restituisce l'azione che dovrà eseguire il Comando Variabile referenziato. Il tipo di azione si può specificare usando l'enumeratore eSetVariableMode oppure inserendo il corrispondente valore numerico:

```

enum_svm_set (valore 0, Set)
enum_svm_reset (valore 1, Reset)
enum_svm_toggle (valore 2, Toggle)
enum_svm_strobe (valore 3, Impulso Temporizzato)
enum_svm_increase (valore 4, Incrementa)
enum_svm_decrease (valore 5, Decrementa)
enum_svm_alphanumeric (valore 6, Pad Alfanumerico)
enum_svm_numeric (valore 7, Pad Numerico)
enum_svm_AppendValue (valore 8, Appendi Valore)
enum_svm_BackValue (valore 9, Rimuovi Valore)
enum_svm_SwapPlusMinus (valore 10, Cambia Segno +/-)
enum_svm_AppendDecimalMode (valore 11, Appendi Decimale ON-OFF)
enum_svm_MoveValue (valore 12, Trasferisci Valore)
enum_svm_ResetStatistics (valore 13, Reset Statistiche)
enum_svm_MoveMinValue (valore 14, Trasferisci Valore Minimo)
enum_svm_MoveMaxValue (valore 15, Trasferisci Valore Massimo)
enum_svm_MoveAveValue (valore 16, Trasferisci Valore Medio)
enum_svm_SetStringID (valore 17, Set da ID Stringa)

```



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result eSetVariableMode

Example1:

```

Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget

```

```

Dim objCommandVariable As CommandVariableCmdTarget

Set objButtonRelease =
GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)

objCommandVariable.Action = enum_svm_set
objCommandList.SaveChanges

Set objCommandVariable = Nothing
Set objCommandList = Nothing
Set objButtonRelease = Nothing
End Sub

```

Example2:

```

Public Sub Click()
Dim objRect As DrawCmdTarget
Dim objCommandList As CommandsListCmdTarget
Dim objCommandVariable As CommandVariableCmdTarget

Set objRect = GetSynopticObject.GetSubObject("objRect")
Set objCommandList = objRect.GetCommandsInterfaceOnRelease
Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)

objCommandVariable.Action = enum_svm_set
objCommandList.SaveChanges

Set objCommandVariable = Nothing
Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

MaxChar, CommandVariableCmdTarget Property

Syntax MaxChar = _Long

Description Questa proprietà imposta o restituisce il numero massimo di caratteri impostabili su una variabile stringa tramite il Comando Variabile referenziato.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandVariable As CommandVariableCmdTarget  
  
    Set objButtonRelease =  
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandVariable.MaxChar = 10  
    objCommandList.SaveChanges  
  
    Set objCommandVariable = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandVariable As CommandVariableCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandVariable.MaxChar = 10  
    objCommandList.SaveChanges  
  
    Set objCommandVariable = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

MaxValue, CommandVariableCmdTarget Property

Syntax MaxValue = _String

Description Questa proprietà imposta o restituisce il valore massimo impostabile su una variabile numerica tramite il Comando Variabile referenziato.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandVariable As CommandVariableCmdTarget  
  
    Set objButtonRelease =  
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandVariable.MaxValue = "100"  
    objCommandList.SaveChanges  
  
    Set objCommandVariable = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandVariable As CommandVariableCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandVariable.MaxValue = "100"  
    objCommandList.SaveChanges  
  
    Set objCommandVariable = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

MinValue, CommandVariableCmdTarget Property

Syntax MinValue = _String

Description Questa proprietà imposta o restituisce il valore minimo impostabile su una variabile numerica tramite il Comando Variabile referenziato.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandVariable As CommandVariableCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandVariable.MinValue = "0"
    objCommandList.SaveChanges

    Set objCommandVariable = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandVariable As CommandVariableCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandVariable.MinValue = "0"
    objCommandList.SaveChanges

    Set objCommandVariable = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

MoveToVariable, CommandVariableCmdTarget Property

Syntax MoveToVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile di destinazione sulla quale trasferire il valore tramite il Comando Variabile referenziato.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di

configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandVariable As CommandVariableCmdTarget  
  
    Set objButtonRelease =  
        GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandVariable.MoveToVariable = "VAR0002"  
    objCommandList.SaveChanges  
  
    Set objCommandVariable = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandVariable As CommandVariableCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandVariable.MoveToVariable = "VAR0002"  
    objCommandList.SaveChanges  
  
    Set objCommandVariable = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

PasswordStyle, CommandVariableCmdTarget Property

Syntax PasswordStyle = _Boolean

Description Questa proprietà imposta o restituisce il valore della proprietà "Stile Password" del Comando Variabile referenziato. La proprietà "Stile Password" consente di inserire il valore sulla variabile tramite il Pad Alfanumerico in modo criptato o in chiaro.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo `SaveChanges` dell'interfaccia `CommandsListCmdTarget` per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Boolean

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandVariable As CommandVariableCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButton").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandVariable.PasswordStyle = Not objCommandVariable.PasswordStyle
    objCommandList.SaveChanges

    Set objCommandVariable = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandVariable As CommandVariableCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRectangle")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandVariable.PasswordStyle = Not objCommandVariable.PasswordStyle
    objCommandList.SaveChanges

    Set objCommandVariable = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

StrobeTime, CommandVariableCmdTarget Property

Syntax StrobeTime = _Long

Description Questa proprietà imposta o restituisce il tempo di Strobe, espresso in millisecondi, durante il quale la variabile rimarrà impostata al valore inserito nella proprietà

"Valore" quando si utilizza l'azione "Strobe" per il Comando Variabile referenziato. Scaduto il tempo la variabile assumerà nuovamente il valore precedente.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result Long

Example1:

```
Public Sub Click()
    Dim objButtonRelease As ButtonCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandVariable As CommandVariableCmdTarget

    Set objButtonRelease =
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandVariable.StrobeTime= 1000
    objCommandList.SaveChanges

    Set objCommandVariable = Nothing
    Set objCommandList = Nothing
    Set objButtonRelease = Nothing
End Sub
```

Example2:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim objCommandList As CommandsListCmdTarget
    Dim objCommandVariable As CommandVariableCmdTarget

    Set objRect = GetSynopticObject.GetSubObject("objRect")
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)

    objCommandVariable.StrobeTime= 1000
    objCommandList.SaveChanges

    Set objCommandVariable = Nothing
    Set objCommandList = Nothing
    Set objRect = Nothing
End Sub
```

Value, CommandVariableCmdTarget Property

Syntax Value = _String

Description Questa proprietà imposta o restituisce il valore che verrà inserito nella variabile tramite il Comando Variabile referenziato.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandVariable As CommandVariableCmdTarget  
  
    Set objButtonRelease =  
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandVariable.Value= "5"  
    objCommandList.SaveChanges  
  
    Set objCommandVariable = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandVariable As CommandVariableCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandVariable.Value= "5"  
    objCommandList.SaveChanges  
  
    Set objCommandVariable = Nothing  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

Variable, CommandVariableCmdTarget Property

Syntax Variable = _String

Description Questa proprietà imposta o restituisce il nome della variabile su cui eseguire il comando tramite il Comando Variabile referenziato.



Dopo avere aggiunto o modificato un comando della lista comandi dell'oggetto è necessario eseguire il metodo SaveChanges dell'interfaccia CommandsListCmdTarget per rendere effettive le modifiche sulla lista comandi dell'oggetto.

Si ricorda inoltre che le modifiche ad una lista comandi rimarranno valide soltanto fino a quando l'oggetto non verrà scaricato dalla memoria (chiusura del sinottico), dopodichè al successivo caricamento dell'oggetto verrà ripristinata la lista comandi associata in programmazione. E' tuttavia possibile rendere persistenti le modifiche di una lista comandi associando un file di configurazione all'oggetto ed eseguendo il salvataggio del file di configurazione dopo avere eseguito la modifica e il salvataggio della lista comandi.

Parameter	Description
None	None

Result String

Example1:

```
Public Sub Click()  
    Dim objButtonRelease As ButtonCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandVariable As CommandVariableCmdTarget  
  
    Set objButtonRelease =  
    GetSynopticObject.GetSubObject("objButtonRelease").GetObjectInterface  
    Set objCommandList = objButtonRelease.GetCommandsInterfaceOnRelease  
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandVariable.Variable= "VAR00001"  
    objCommandList.SaveChanges  
  
    Set objCommandVariable = Nothing  
    Set objCommandList = Nothing  
    Set objButtonRelease = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
    Dim objCommandVariable As CommandVariableCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
    Set objCommandVariable = objCommandList.GetCommandInterfaceAtPos(0)  
  
    objCommandVariable.Variable= "VAR00001"  
    objCommandList.SaveChanges  
  
    Set objCommandVariable = Nothing
```

```

Set objCommandList = Nothing
Set objRect = Nothing
End Sub

```

1.14.20. DBVariableCmdTarget

1.15. Utilizzo della DBVariableCmdTarget

L'utilizzo dell'interfaccia di programmazione "DBVariableCmdTarget" è diretto, cioè non è necessario istanziare un oggetto di tipo "DBVariableCmdTarget" ma è possibile utilizzare le sue proprietà e metodi direttamente nel codice VB Script. In pratica tutte le proprietà e funzioni della "DBVariableCmdTarget" sono disponibili direttamente dall'Intellisense indipendentemente dal contesto VB Script in cui ci si trova.

Functions

CreateNewVar, DBVariableCmdTarget Function

Syntax CreateNewVar(_IpszVariableName, _nType, _IpszStructName)

Description Questa funzione consente di creare una nuova variabile durante la fase di startup del progetto. La funzione si può quindi utilizzare soltanto dal basic script che è stato impostato come script di startup del progetto. Nel caso venga utilizzata in qualche altra parte rispetto al punto sopra specificato, il valore di ritorno sarà un oggetto "Nothing". Inoltre è buona norma verificare che la variabile oggetto che ritorna sia valida usando il metodo "IsValid" dell'interfaccia "DBVarObjCmdTarget".

Parameter	Description
IpszVariableName As String	Nome della variabile che si vuole creare a runtime. Se esiste già una variabile con quel nome, allora viene aggiunto un suffisso numerico incrementale fino a quando la variabile non ha un nome univoco. (*)Per creare una variabile di tipo Array occorre specificare questo parametro nella forma "Variabile:NumElementi" e nel parametro 'nType' indicare il tipo di elementi dell'array.
nType As Integer	Tipo di variabile da creare fra i seguenti tipi supportati: 0 : enum_VAR_TYPE_BIT 1: enum_VAR_TYPE_SIGNBYTE 2: enum_VAR_TYPE_BYTE 3: enum_VAR_TYPE_SIGNWORD 4: enum_VAR_TYPE_WORD 5: enum_VAR_TYPE_SIGNDWORD 6: enum_VAR_TYPE_DWORD 7: enum_VAR_TYPE_FLOAT 8: enum_VAR_TYPE_DOUBLE 9: enum_VAR_TYPE_STRING 10: enum_VAR_TYPE_ARRAY (*) 11: enum_VAR_TYPE_STRUCT
IpszStructName As String	Nome del prototipo struttura da utilizzare per creare la nuova variabile di tipo struttura (nType = enum_VAR_TYPE_STRUCT). In caso di variabili di tipo diverso questo parametro non viene considerato.

Result Object
Viene restituito un oggetto di tipo DBVarObjCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Option Explicit
Sub Main
    Dim objVar As DBVarObjCmdTarget
    Set objVar = CreateNewVar("VAR00002", enum_VAR_TYPE_DWORD, "") 'creo la
    variabile
    If Not objVar Is Nothing Then
        If objVar.IsValid Then
            objVar.DynamicSettings = "[DRV]Modbus TCPIP.Sta=Default
            Station|Unit=1|FC=2|SA=0"
            objVar.SetOPCServerEnabled(True)
        Else
            Debug.Print Replace("The variabile '%s' is not valid!", "%s",
            objVar.GetName)
        End If
    Else
        Debug.Print "Failed to create the variabile 'RuntimeVariable'"
    End If
End Sub
```

GetDynamicVariable, DBVariableCmdTarget Function

Syntax GetDynamicVariable(_IpszDynVariableName)

Description Questa funzione consente di recuperare il nome della variabile dinamica creata da Movicon conoscendo il nome del Tag OPC Server a cui la variabile è connessa. Questa variabile può essere poi utilizzata una volta referenziata con la funzione 'GetVariableObject'.
La sintassi per la definizione del parametro IpszDynamicVariableName è quella specificata nella proprietà "Dinamica" di una variabile:
[OPC]ServerName\GroupName.TagName

Parameter	Description
IpszDynVariableName	Nome del tag del Server OPC.

Result String

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget
    Dim sVariableName As String

    sVariableName
    = GetDynamicVariable("[OPC]Softing.OPCToolboxDemo_ServerDA.1\watch.devi
    ce 1.value 1")
    Set objVar = GetVariableObject(sVariableName)
    Do
        objVar.Value = objVar.Value + 1
        If objVar.Value >= 10000 Then objVar.Value = 0
        DoEvents
    Loop Until IsInStoppingMode
End Sub
```


GetNumDynTag, DBVariableCmdTarget Function

Syntax GetNumDynTag

Description Funzione di lettura del numero di variabili dinamiche (tag) utilizzate all'interno di Movicon.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim IRet As Long
    IRet = GetNumDynTag()
    MsgBox CStr(IRet), vbOkOnly, "Test GetNumDynTag"
End Sub
```

GetNumStructDefinitions, DBVariableCmdTarget Function

Syntax GetNumStructDefinitions

Description Funzione di lettura del numero di prototipi di struttura dichiarati all'interno del Real Tme DB di Movicon.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim IRet As Long
    IRet = GetNumStructDefinitions()
    MsgBox CStr(IRet), vbOkOnly, "Test GetNumStructDefinitions"
End Sub
```

GetNumVariables, DBVariableCmdTarget Function

Syntax GetNumVariables

Description Funzione di lettura del numero di variabili dichiarate all'interno del Real Tme DB di Movicon.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim IRet As Long
    IRet = GetNumVariables()
    MsgBox CStr(IRet), vbOkOnly, "Test GetNumVariables"
End Sub
```

GetRealTimeDBADODConn, DBVariableCmdTarget Function

Syntax GetRealTimeDBADODConn()

Description Questa funzione consente di recuperare la connessione ADODB relativa alle Impostazioni Real Time ODBC delle variabili del progetto.



Questa funzione non è supportata in Windows CE.(se utilizzata restituisce sempre un oggetto "null")

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo ADODB.Connection se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Sub Main
    Dim Conn1 As New ADODB.Connection
    Dim Rs1 As New ADODB.Recordset
    Dim contFields As Integer
    Dim sQuery As String
    Set Conn1 = GetRealTimeDBADODConn
    sQuery = "SELECT * FROM RTVar"
    Set Rs1 = CreateObject("ADODB.Recordset")
    Rs1.Open sQuery, Conn1, adOpenForwardOnly, adLockReadOnly,
    ADODB.adCmdText
    ' Loop per stampare tutti i campi del recorset
    While Not Rs1.EOF
        For contFields = 0 To (Rs1.Fields.Count-1)
            Debug.Print Rs1.Fields(contFields).Name & " = " &
            Rs1.Fields(contFields).Value
        Next
        Rs1.MoveNext
    Wend
    Rs1.Close
    Conn1.Close
End Sub
```

GetStructureDefinitionsList, DBVariableCmdTarget Function

Syntax GetStructureDefinitionsList(Void)

Description Ritorna una stringa contenente i nomi dei prototipi di struttura definiti nel progetto, separati dal carattere (pipe).

Parameter	Description
Void	

Result String

Example:

```
Option Explicit
Sub Main()
Dim aStructList() As String
Dim i As Integer
aStructList = Split(GetStructureDefinitionsList, "|")
For i=0 To UBound(aStructList)
Debug.Print aStructList(i)
Next
End Sub
```

GetTraceDBADODConn, DBVariableCmdTarget Function

Syntax GetTraceDBADODConn()

Description Questa funzione consente di recuperare la connessione ADODB relativa alle Impostazioni Trace DB (Audit) delle variabili del progetto.



Questa funzione se utilizzata su Windows CE restituisce sempre un'oggetto di tipo ADOCE.connection.3.1. Evitare inoltre la chiusura della connessione ADO usando il metodo "close", altrimenti anche Movicon non sarà più in grado di accedere a quel database.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo ADODB.Connection se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Sub Main
Dim Conn1 As New ADODB.Connection
Dim Rs1 As New ADODB.Recordset
Dim contFields As Integer
Dim sQuery As String
Set Conn1 = GetTraceDBADODConn
sQuery = "SELECT * FROM VAR00001"
```

```

Set Rs1 = CreateObject("ADODB.Recordset")
Rs1.Open sQuery, Conn1, adOpenForwardOnly, adLockReadOnly,
ADODB.adCmdText
' Loop per stampare tutti i campi del recorset
While Not Rs1.EOF
    For contFields = 0 To (Rs1.Fields.Count-1)
        Debug.Print Rs1.Fields(contFields).Name & " = " &
Rs1.Fields(contFields).Value
    Next
    Rs1.MoveNext
Wend
Rs1.Close
Conn1.Close
End Sub

```

GetTraceDBDSNConnectionString, DBVariableCmdTarget Function

Syntax GetTraceDBDSNConnectionString

Description Questa funzione consente di recuperare il nome del collegamento ODBC relativo al file di trace delle variabili del progetto.

Parameter	Description
None	None

Result String

Example:

```

Sub Main
    MsgBox "Collegamento ODBC = " &
GetTraceDBDSNConnectionString,vbExclamation,GetProjectTitle
End Sub

```

GetVariableAddressInfo, DBVariableCmdTarget Function

Syntax GetVariableAddressInfo(_lpszVariableName, _nType, _nAreaType, _nAddress, _nSubAddress)

Description Restituisce le informazioni di tipo di dato, area dati, indirizzo e sub indirizzo di una variabile presente nel Real Time DB di Movicon. Il valore di ritorno True o False, può essere utilizzato per verificare se la variabile identificata dal parametro lpszVariableName è presente nel progetto.

Il parametro nType può assumere i seguenti valori:

- 0 = Bit
- 1 = Sign Byte (8 Bit con segno)
- 2 = Byte (8 Bit senza segno)
- 3 = Sign Word (16 Bit con segno)
- 4 = Word (16 Bit senza segno)
- 5 = Sign DWord (32 Bit con segno)
- 6 = DWord (32 Bit senza segno)
- 7 = Float (32 bit precisione singola)
- 8 = Double (64 Bit doppia precisione)
- 9 = Stringa (Terminazione con 0)
- 10 = Array di Byte a lunghezza fissa
- 11 = Struttura di dati

Il parametro nAreaType può assumere i seguenti valori:
 0 = Input
 1 = Flag
 2 = Output
 3 = Non Condivise

Parameter	Description
IpszVariableName As String	Nome della variabile.
nType	Indice indicante il tipo di dato.
nAreaType	Indice indicante l'area dati.
nAddress	Indirizzo della variabile
nSubAddress	sub indirizzo. Numero del bit per le variabili di tipo bit. Numero di byte nel caso delle variabili array.

Result String

Example:

```
Sub Main
    Dim IpszVariableName As String
    Dim nType As Integer
    Dim nAreaType As Integer
    Dim nAddress As Long
    Dim nSubAddress As Integer

    IpszVariableName = "VAR00001"
    GetVariableAddressInfo(IpszVariableName, nType, nAreaType, nAddress,
    nSubAddress)

    MsgBox "Variable Name = " & IpszVariableName & vbCrLf & _
    "Variable Type = " & nType & vbCrLf & _
    "Variable Area = " & nAreaType & vbCrLf & _
    "Variable Address = " & nAddress & vbCrLf & _
    "Variable SubAddress = " & nSubAddress, vbInformation, GetProjectTitle
End Sub
```

GetVariableDescription, DBVariableCmdTarget Function

Syntax GetVariableDescription(_IpszVariableName)

Description La funzione restituisce il testo di descrizione associato alla variabile dichiarata nel progetto. Infatti nella creazione di una variabile nel Real Tme DB di Movicon è possibile assegnare ad ognuna una descrizione. Il valore di ritorno è una stringa nulla nel caso in cui la variabile non sia dichiarata nel progetto.

Parameter	Description
IpszVariableName As String	Nome della variabile.

Result String

Example:

```

Sub Main
    Dim sRet As String
    sRet = GetVariableDescription("VAR00001")
    MsgBox sRet, vbOkOnly, "Test GetVariableDescription"
End Sub

```

GetVariableObject, DBVariableCmdTarget Function

Syntax GetVariableObject(_IpszVariableName)

Description Funzione che restituisce un oggetto utilizzabile per gestire una variabile nelle proprie proprietà e metodi. Da utilizzare con il comando Set e con un tipo di variabile Object o più precisamente di tipo DBVarObjCmdTarget. Per maggiori informazioni sulle potenzialità offerte si rimanda all'help della DBVarObjCmdTarget Interface.

Parameter	Description
IpszVariableName As String	Nome della variabile.

Result Object
Viene restituito un oggetto di tipo DBVarObjCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```

Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    MsgBox objRet.GetDescription(), vbOkOnly, "Test GetVariableObject"
    Set objRet = Nothing
End Sub

```

GetVariableSize, DBVariableCmdTarget Function

Syntax GetVariableSize(_IpszVariableName)

Description Funzione di lettura del numero di byte occupati da una determinata variabile dichiarata all'interno del Real Time DB di Movicon.

Parameter	Description
IpszVariableName As String	Nome della variabile.

Result Long

Example:

```

Sub Main
    Dim lRet As Long
    lRet = GetVariableSize("VAR00001")
    MsgBox CStr(lRet), vbOkOnly, "Test GetVariableSize"
End Sub

```

GetVariableValue, DBVariableCmdTarget Function

Syntax GetVariableValue(_IpszVariableName)

Description Funzione di lettura del valore contenuto in una variabile del Real Time DB di Movicon, identificata dal nome in formato stringa (parametro IpszVarName).

Parameter	Description
IpszVariableName As String	Nome della variabile.

Result Variant

Example:

```
Sub Main
    Dim value As Variant
    value = GetVariableValue("VAR00001")
    MsgBox CStr(value), vbOkOnly, "Test GetVariableValue"
End Sub
```

GetXMLSettings, DBVariableCmdTarget Function

Syntax GetXMLSettings

Description Questa funzione restituisce il testo XML di impostazione e configurazione del Real Time DB di Movicon. Di fatto è il contenuto del file di risorse "nomeprogetto.movrealtimedb".

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    MsgBox GetXMLSettings,vbInformation,GetProjectTitle
End Sub
```

IsFirstDBInstance, DBVariableCmdTarget Function

Syntax IsFirstDBInstance

Description Questa funzione restituisce il valore "false" quando l'istanza di Movicon in esecuzione è stata eseguita a seguito di un comando "safe" (comando "Apri in Processo separato (Safe Mode)"). Ritorna "true" in tutti gli altri casi.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim vResult As Variant

    vResult = IsFirstDBInstance()
    MsgBox "IsFirstDBInstance = " & vResult,vbInformation,GetProjectTitle
End Sub
```

IsValidVariable, DBVariableCmdTarget Function

Syntax IsValidVariable(_IpszVariableName)

Description Questa funzione consente di verificare se la variabile richiesta è effettivamente una variabile valida appartenente al Real Time DB di Movicon.

Parameter	Description
IpszVariableName As String	Nome della variabile.

Result Boolean

Example:

```
Sub Main
    Dim vResult As Variant

    vResult = IsValidVariable("VAR00001")
    MsgBox "VAR00001 = " & vResult,vbInformation,GetProjectTitle
End Sub
```

PurgeDynTag, DBVariableCmdTarget Function

Syntax PurgeDynTag

Description Questa proprietà consente di liberare lo spazio di memoria occupato dalle variabili dinamiche in uso in maniera forzata. Tale operazione avviene in maniera automatica allo scadere del 'Purge Dinamic Variable Timeout' impostabile con la funzione PurgeDynTagTimer descritta nell'interfaccia DBVariableCmdTarget

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim vResult As Variant
    vResult = PurgeDynTag
    MsgBox "PurgeDynTag = " & vResult,vbInformation,GetProjectTitle
End Sub
```

QualityOf, DBVariableCmdTarget Function

Syntax QualityOf(_IpszVariableName)

Description Questa proprietà consente di leggere lo stato di qualità della variabile specificata. La proprietà è di sola lettura.

I valori restituiti sono i valori di qualità come da specifiche OPC:

0 = OPC_QUALITY_BAD
64 = OPC_QUALITY_UNCERTAIN
192 = OPC_QUALITY_GOOD
4 = OPC_QUALITY_CONFIG_ERROR
8 = OPC_QUALITY_NOT_CONNECTED
12 = OPC_QUALITY_DEVICE_FAILURE
16 = OPC_QUALITY_SENSOR_FAILURE
20 = OPC_QUALITY_LAST_KNOWN
24 = OPC_QUALITY_COMM_FAILURE
28 = OPC_QUALITY_OUT_OF_SERVICE
68 = OPC_QUALITY_LAST_USABLE
80 = OPC_QUALITY_SENSOR_CAL
84 = OPC_QUALITY_EGU_EXCEEDED
88 = OPC_QUALITY_SUB_NORMAL
216 = OPC_QUALITY_LOCAL_OVERRIDE

Non è possibile utilizzare questa proprietà nelle espressioni basic. Ad esempio, non è possibile assegnare alla proprietà "Variabile Display" di un display l'espressione "QualityOf("VAR00001")" poichè le espressioni basic supportano soltanto gli operatori matematici del WinWrap.

Parameter	Description
IpszVariableName as string	Nome della variabile di cui si vuole conoscere la qualità

Result Integer

Example:

```
Public Sub Click()
    MsgBox "Quality of the variable VAR00001 = " & CStr(QualityOf("VAR00001")),
    vbOkOnly, GetProjectTitle
End Sub
```

SetVariableValue, DBVariableCmdTarget Function

Syntax SetVariableValue(_IpszVariableName, _vtValue)

Description Questa funzione permette di scrivere un valore (numerico o stringa) in una variabile del Real Time DB. La variabile deve essere specificata per nome dal parametro `IpszVariable`.

Parameter	Description
<code>IpszVariableName As String</code>	Stringa identificatrice della Variabile nel Real Time DB.
<code>vtValue As Variant</code>	Valore da scrivere nella variabile.

Result Boolean

Example:

```
Sub Main
    Dim bRet As Boolean
    bRet = SetVariableValue("VAR00001", 11)
    MsgBox CStr(bRet), vbOkOnly, "Test SetVariableValue"
End Sub
```

VariableInUse, DBVariableCmdTarget Function

Syntax `VariableInUse(_IpszVariableName) = _Boolean`

Description Questa funzione consente di impostare la variabile come "in uso" o come "non in uso". Il valore True imposta la variabile "in uso". Il valore False imposta la variabile "non in uso".

Non è possibile utilizzare il metodo "VariableInUse" relativamente alla variabile di sistema "_SysVar_". In questo caso verrà generato un errore quando il codice basic che ha eseguito il comando viene scaricato dalla memoria. Il messaggio di errore notificato è "Nessun messaggio di errore disponibile".

Parameter	Description
<code>IpszVariableName As String</code>	Nome della variabile.

Result Boolean

Example:

```
Sub Main
    Dim vResult As Variant

    vResult = VariableInUse("VAR00001")
    MsgBox "VAR00001 InUse = " & vResult, vbInformation, GetProjectTitle
End Sub
```

Properties

EnableInUseVarMng, DBVariableCmdTarget Property

Syntax EnableInUseVarMng = _Boolean

Description Questa proprietà consente di abilitare o disabilitare la gestione delle variabili in uso del progetto.

Attenzione: se durante la fase di runtime questa proprietà viene impostata al valore "false" si avranno i seguenti comportamenti del progetto:

- Gli eventi su variabile negli basic script non verranno più eseguiti fino a quando il Basic Script non viene scaricato e ricaricato in memoria
- I sinottici caricati in memoria dovranno essere scaricati e ricaricati in memoria, altrimenti tutte le modifiche alle variabili utilizzate al loro interno non aggiorneranno più gli oggetti dei sinottici

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    MsgBox "EnableInUseVarMng = " & EnableInUseVarMng, vbInformation,
    GetProjectTitle
End Sub
```

EnableNTSecurityOPCServerTag, DBVariableCmdTarget Property

Syntax EnableNTSecurityOPCServerTag = _Boolean

Description Questa proprietà consente di abilitare o disabilitare la gestione della sicurezza nell'OPC Server.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre false)

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    MsgBox "EnableNTSecurityOPCServerTag = " &
EnableNTSecurityOPCServerTag, vbInformation, GetProjectTitle
End Sub
```

OPCServerEnableAEAck, DBVariableCmdTarget Property

Syntax OPCServerEnableAEAck = _Boolean

Description Questa proprietà consente di abilitare o disabilitare l'OPC Server AE e quindi la funzione di notifica degli allarmi ed eventi di un progetto, a tutti i Client OPC connessi.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre false)

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    MsgBox "OPCServerEnableAEAck = " & OPCServerEnableAEAck,
        vbInformation, GetProjectTitle
End Sub
```

OPCServerMinImpersonationLevel, DBVariableCmdTarget Property

Syntax OPCServerMinImpersonationLevel = _Long

Description Questa proprietà consente di leggere o impostare il livello minimo di impersonalizzazione dell'OPC server.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre il valore 0)

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    MsgBox "OPCServerMinImpersonationLevel = " &
        OPCServerMinImpersonationLevel, vbInformation, GetProjectTitle
End Sub
```

OPCServerShutdownClientsTimeout, DBVariableCmdTarget Property

Syntax OPCServerShutdownClientsTimeout = _Long

Description Questa proprietà consente di leggere o impostare il tempo minimo di attesa prima della chiusura dell'OPC server.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre il valore 0)

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    MsgBox "OPCServerShutdownClientsTimeout = " &
OPCServerShutdownClientsTimeout,vbInformation,GetProjectTitle
End Sub
```

PurgeDynTagTimer, DBVariableCmdTarget Property

Syntax PurgeDynTagTimer = _Long

Description Questa proprietà consente di leggere o impostare il numero di millisecondi dopo il quale le variabili dinamiche non utilizzate vengono rimosse dalla gestione delle variabili in uso.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    MsgBox "Purge Dynamic Tag Timer = " &
PurgeDynTagTimer,vbInformation,GetProjectTitle
End Sub
```

TraceDBChangerColName, DBVariableCmdTarget Property

Syntax TraceDBChangerColName = _String

Description Questa proprietà consente leggere o impostare il nome della "Colonna Cambiamento" della tabella di Trace delle variabili.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    MsgBox "Changer Column Name = " &
TraceDBChangerColName,vbInformation,GetProjectTitle
End Sub
```

TraceDBDefVarCharPrecision, DBVariableCmdTarget Property

Syntax TraceDBDefVarCharPrecision = _Long

Description Questa proprietà consente leggere o impostare la massima precisione per le colonne di tipo stringa della tabella di Trace delle variabili. Il numero impostato rappresenta il numero di caratteri della stringa.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    MsgBox "Char Precision = " &
TraceDBDefVarCharPrecision,vbInformation,GetProjectTitle
End Sub
```

TraceDBDsn, DBVariableCmdTarget Property

Syntax TraceDBDsn = _String

Description Questa proprietà consente leggere o impostare il nome del collegamento ODBC che Movicon utilizzerà per collegarsi al database di Trace delle variabili. Di default Movicon creerà un DSN con lo stesso nome del progetto e suffisso "_TraceDB", configurato per accedere al database specificato tramite il "PlugIn ODBC di Default" del progetto. Il nome del DSN sarà del tipo:

NomeProgetto_TraceDB

Tramite questa proprietà è però possibile personalizzare il collegamento **ODBC**, creando una base dati personalizzata anche differente da quella definita nel "PlugIn ODBC di Default" del progetto.



Questa proprietà può essere utilizzata in scrittura solo in fase di design del progetto, ad esempio nel dropping code di un simbolo Template. In fase di Runtime invece, anche se può essere modificata, non verrà applicata al TraceDB, che continuerà a registrare utilizzando il DSN iniziale.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    MsgBox "DSN = " & TraceDBDSn,vbInformation,GetProjectTitle
End Sub
```

TraceDBLocalTimeColName, DBVariableCmdTarget Property

Syntax TraceDBLocalTimeColName = _String

Description Questa proprietà consente leggere o impostare il nome della "Colonna Tempo Locale" della tabella di Trace delle variabili.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    MsgBox "Time Column Name = " &
TraceDBLocalTimeColName,vbInformation,GetProjectTitle
End Sub
```

TraceDBMaxCacheBeforeFlush, DBVariableCmdTarget Property

Syntax TraceDBMaxCacheBeforeFlush = _Long

Description Questa proprietà consente di leggere o impostare la massima dimensione della Cache prima che il sistema scarichi i dati sul file di Trace delle variabili. Il numero impostato è espresso in Byte.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    MsgBox "Time Max Cache = " &
        cstr(TraceDBMaxCacheBeforeFlush),vbInformation,GetProjectTitle
End Sub
```

TraceDBMaxError, DBVariableCmdTarget Property

Syntax TraceDBMaxError = _Long

Description Questa proprietà consente leggere o impostare il numero massimo di errori DBMS dopo i quali la connessione al database di Trace delle variabili viene considerata non valida e i dati verranno salvati su file in formato ASCII nella cartella relativa del progetto ("LOGS").

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    MsgBox "Max Error = " & TraceDBMaxError,vbInformation,GetProjectTitle
End Sub
```

TraceDBMaxNumberTrans, DBVariableCmdTarget Property

Syntax TraceDBMaxNumberTrans = _Long

Description Questa proprietà consente di leggere o impostare il numero massimo di transizioni per ciclo da aggiornare prima della loro chiusura per la connessione al database di Trace delle variabili.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    MsgBox "Max Num Transition = " &
        TraceDBMaxNumberTrans,vbInformation,GetProjectTitle
End Sub
```

TraceDBMSecColName, DBVariableCmdTarget Property

Syntax TraceDBMSecColName = _String

Description Questa proprietà consente leggere o impostare il nome della "Colonna MSec" della tabella di Trace delle variabili.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    MsgBox "MSec Column Name = " &
        TraceDBMSecColName,vbInformation,GetProjectTitle
End Sub
```

TraceDBQualityColName, DBVariableCmdTarget Property

Syntax TraceDBQualityColName = _String

Description Questa proprietà consente leggere o impostare il nome della "Colonna Qualità" della tabella di Trace delle variabili.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta

invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

Sub Main

```
MsgBox "Quality Column Name = " &  
TraceDBQualityColName,vbInformation,GetProjectTitle
```

End Sub

TraceDBRecycleDBConnection, DBVariableCmdTarget Property

Syntax TraceDBRecycleDBConnection = _Boolean

Description Questa proprietà consente abilitare/disabilitare la proprietà "Mantieni aperta la Connessione DB" del database di Trace delle variabili.

Parameter	Description
None	None

Result Boolean

Example:

Sub Main

```
MsgBox "Recycle DB Connection = " &  
TraceDBRecycleDBConnection,vbInformation,GetProjectTitle
```

End Sub

TraceDBTimeColName, DBVariableCmdTarget Property

Syntax TraceDBTimeColName = _String

Description Questa proprietà consente leggere o impostare il nome della "Colonna Tempo" della tabella di Trace delle variabili.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    MsgBox "Time Column Name = " &
TraceDBTimeColName, vbInformation, GetProjectTitle
End Sub
```

TraceDBUser, DBVariableCmdTarget Property

Syntax TraceDBUser = _String

Description Questa proprietà consente di specificare il nome utente che verrà utilizzato per la connessione ODBC per il file database di Trace delle variabili.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    MsgBox "User Name = " & TraceDBUser, vbInformation, GetProjectTitle
End Sub
```

TraceDBUserColName, DBVariableCmdTarget Property

Syntax TraceDBUserColName = _String

Description Questa proprietà consente leggere o impostare il nome della "Colonna Utente" della tabella di Trace delle variabili.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    MsgBox "User Column Name = " &
        TraceDBUserColName,vbInformation,GetProjectTitle
End Sub
```

TraceDBValueAfterColName, DBVariableCmdTarget Property

Syntax `TraceDBValueAfterColName = _String`

Description Questa proprietà consente leggere o impostare il nome della "Colonna Valore Successivo" della tabella di Trace delle variabili.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    MsgBox "Value After Column Name = " &
        TraceDBValueAfterColName,vbInformation,GetProjectTitle
End Sub
```

TraceDBValueBeforeColName, DBVariableCmdTarget Property

Syntax `TraceDBValueBeforeColName = _String`

Description Questa proprietà consente leggere o impostare il nome della "Colonna Valore Precedente" della tabella di Trace delle variabili.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description

None	None
------	------

Result String

Example:

```
Sub Main
    MsgBox "Value Before Column Name = " &
TraceDBValueBeforeColName,vbInformation,GetProjectTitle
End Sub
```

TraceDBValueColName, DBVariableCmdTarget Property

Syntax TraceDBValueColName = _String

Description Questa proprietà consente leggere o impostare il nome della "Colonna Valore" della tabella di Trace delle variabili.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    MsgBox "Value Column Name = " &
TraceDBValueColName,vbInformation,GetProjectTitle
End Sub
```

TraceDBTimeStampColName, DBVariableCmdTarget Property

Syntax TraceDBTimeStampColName = _String

Description Questa proprietà consente di leggere o impostare il nome della "Colonna TimeStamp" della tabella di Trace delle variabili.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim TraceDBTable As DBVariableCmdTarget

    Set TraceDBTable = GetRealTimeDB()
    MsgBox "TraceDBTimeStampColName = " &
        TraceDBTable.TraceDBTimeStampColName, vbInformation, GetProjectTitle
    Set TraceDBTable = Nothing
End Sub
```

TraceDBVarNameColName, DBVariableCmdTarget Property

Syntax TraceDBVarNameColName = _String

Description Questa proprietà consente di leggere o impostare il nome della "Colonna Nome Tag" della tabella di Trace delle variabili.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim TraceDBTable As DBVariableCmdTarget

    Set TraceDBTable = GetRealTimeDB()
    MsgBox "TraceDBVarNameColName = " &
        TraceDBTable.TraceDBVarNameColName, vbInformation, GetProjectTitle
    Set TraceDBTable = Nothing
End Sub
```

TraceDBVarDescNameColName, DBVariableCmdTarget Property

Syntax TraceDBVarDescNameColName = _String

Description Questa proprietà consente di leggere o impostare il nome della "Colonna Descrizione Tag" della tabella di Trace delle variabili.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim TraceDBTable As DBVariableCmdTarget  
  
    Set TraceDBTable = GetRealTimeDB()  
    MsgBox "TraceDBVarDescNameColName = " &  
        TraceDBTable.TraceDBVarDescNameColName,vbInformation,GetProjectTitle  
    Set TraceDBTable = Nothing  
End Sub
```

TraceDBVarGroupNameColName, DBVariableCmdTarget Property

Syntax TraceDBVarGroupNameColName = _String

Description Questa proprietà consente di leggere o impostare il nome della "Colonna Gruppo Tag" della tabella di Trace delle variabili.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim TraceDBTable As DBVariableCmdTarget  
  
    Set TraceDBTable = GetRealTimeDB()  
    MsgBox "TraceDBVarGroupNameColName = " &  
        TraceDBTable.TraceDBVarGroupNameColName,vbInformation,GetProjectTitle  
    Set TraceDBTable = Nothing  
End Sub
```

TraceDBVarNameColName, DBVariableCmdTarget Property

Syntax TraceDBVarNameColName = _String

Description Questa proprietà consente di leggere o impostare il nome della "Colonna Nome Tag" della tabella di Trace delle variabili.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim TraceDBTable As DBVariableCmdTarget  
  
    Set TraceDBTable = GetRealTimeDB()  
    MsgBox "TraceDBVarNameColName = " &  
        TraceDBTable.TraceDBVarNameColName, vbInformation, GetProjectTitle  
    Set TraceDBTable = Nothing  
End Sub
```

TraceUseIMDB,DBVariableCmdTarget Property

Syntax TraceUseIMDB = _Boolean

Description Questa proprietà consente di leggere il valore della proprietà 'Usa gestione IMDB' per registrare i dati di Trace delle variabili.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main  
    MsgBox "Use IMDB manager for Trace DB = " & TraceUseIMDB, vbInformation,  
        GetProjectTitle  
End Sub
```


UseSharedDynTag, DBVariableCmdTarget Property

Syntax UseSharedDynTag = _Boolean

Description Questa funzione consente di verificare se la variabile richiesta è effettivamente una variabile valida appartenente al Real Time DB di Movicon.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    MsgBox "UseSharedDynTag = " &
        UseSharedDynTag, vbInformation, GetProjectTitle
End Sub
```

1.15.1. DBVarObjCmdTarget

Functions

Description, DBVarObjCmdTarget Function

Syntax Description = _String

Description La funzione imposta o restituisce il testo di descrizione associato alla variabile referenziata. Infatti nella creazione di una variabile nel Real Time DB di Movicon è possibile assegnarle una descrizione.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    MsgBox objRet.Description(), vbOkOnly, GetProjectTitle
    Set objRet = Nothing
End Sub
```

GetAccessLevelReadMask, DBVarObjCmdTarget Function

Syntax GetAccessLevelReadMask

Description La funzione restituisce un numero che, interpretato a bit, descrive la maschera di accesso alla variabile per il permesso di accesso in lettura. Il valore è di tipo Long ma l'informazione è contenuta nella word più significativa. I livelli di accesso gestiti infatti sono 16 corrispondenti ai 16 bit più significativi.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Dim IReadMask As Long
    Dim IWriteMask As Long
    Set objRet = GetVariableObject("VAR00001")
    IReadMask = objRet.GetAccessLevelReadMask()
    IWriteMask = objRet.GetAccessLevelWriteMask()
    MsgBox "Read Mask = " & Left(Hex(IReadMask),4) & vbCrLf & _
        "Write Mask = " & Left(Hex(IWriteMask),4), _
        vbOkOnly, "Test Mask"
    Set objRet = Nothing
End Sub
```

GetAccessLevelWriteMask, DBVarObjCmdTarget Function

Syntax GetAccessLevelWriteMask

Description La funzione restituisce un numero che, interpretato a bit, descrive la maschera di accesso alla variabile per il permesso di accesso in scrittura. Il valore è di tipo Long ma l'informazione è contenuta nella word più significativa. I livelli di accesso gestiti infatti sono 16 corrispondenti ai 16 bit più significativi.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Dim IReadMask As Long
    Dim IWriteMask As Long
    Set objRet = GetVariableObject("VAR00001")
    IReadMask = objRet.GetAccessLevelReadMask()
    IWriteMask = objRet.GetAccessLevelWriteMask()
    MsgBox "Read Mask = " & Left(Hex(IReadMask),4) & vbCrLf & _
        "Write Mask = " & Left(Hex(IWriteMask),4), _
        vbOkOnly, "Test Mask"
```

```

        Set objRet = Nothing
End Sub

```

GetAddress, DBVarObjCmdTarget Function

Syntax GetAddress

Description La funzione restituisce un numero che identifica l'indirizzo assoluto in byte della variabile dichiarata nel Real Time DB del progetto in riferimento all'area dati associata (Input, Output o Flag).
Nel caso si tratti di una variabile non condivisa (senza indirizzo assoluto assegnato) il valore tornato sarà 0. Nel caso si tratti di una variabile di tipo bit, l'indirizzo restituito non specifica la posizione del bit all'interno del byte; in questo caso per sapere l'esatto indirizzo del bit è necessario utilizzare la funzione GetBitNumber().

Parameter	Description
None	None

Result Long

Example:

```

Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    MsgBox CStr(objRet.GetAddress()), vbOkOnly, GetProjectTitle
    Set objRet = Nothing
End Sub

```

GetAlarmListName, DBVarObjCmdTarget Function

Syntax GetAlarmListName(_nIndex)

Description La funzione restituisce il nome dell'allarme i-esimo relativo alla variabile referenziata.

Parameter	Description
nIndex as Long	i-esim allarme relativo alla variabile referenziata.

Result String

Example:

```

Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    If objRet.GetAlarmListNum() > 0 Then
        MsgBox "GetAlarmListName: " & CStr(objRet.GetAlarmListName(0)),
            vbOkOnly, GetProjectTitle
    End If
    Set objRet = Nothing
End Sub

```

GetAlarmListNum, DBVarObjCmdTarget Function

Syntax GetAlarmListNum

Description La funzione restituisce il numero delle liste allarmi relative alla variabile referenziata.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    MsgBox "GetAlarmListNum: " & CStr(objRet.GetAlarmListNum()), vbOkOnly,
    GetProjectTitle
    Set objRet = Nothing
End Sub
```

GetAlarmObject, DBVarObjCmdTarget Function

Syntax GetAlarmObject(_IpszAlarmName)

Description La funzione restituisce l'oggetto allarme identificato dal nome in formato stringa (parametro IpszAlarmName). Nel caso in cui l'allarme sia stato associato alla variabile (allarme come template) non è più sufficiente specificare solo il nome dell'allarme, ma si dovrà specificare anche il nome della variabile:

GetAlarmObject(<Nome Allarme> <Nome Variabile>)

Parameter	Description
IpszAlarmName as String	nome dell'allarme da reperire.

Result Object
Viene restituito un oggetto di tipo AlarmCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example1:

```
Public Sub Click()
    Dim objRet As DBVarObjCmdTarget
    Dim objAlarm As AlarmCmdTarget

    Set objRet = GetVariableObject("VAR00001")
    Set objAlarm = objRet.GetAlarmObject("Alarm00001") 'where Alarmr00001 is
    the alarm set by VAR00001
    If Not objAlarm Is Nothing Then
        Debug.Print objAlarm.Name
        Set objAlarm = Nothing
    End If
```

```

        Set objRet = Nothing
    End Sub

```

Example2:

```

Public Sub Click()
    Dim objRet As DBVarObjCmdTarget
    Dim objAlarm As AlarmCmdTarget

    Set objRet = GetVariableObject("VAR00001")
    Set objAlarm = objRet.GetAlarmObject("Alarm00001 VAR00001") 'where
    Alarmr00001 is the alarm associate to VAR00001
    If Not objAlarm Is Nothing Then
        Debug.Print objAlarm.Name
        Set objAlarm = Nothing
    End If
    Set objRet = Nothing
End Sub

```

GetAreaType, DBVarObjCmdTarget Function

Syntax GetAreaType

Description La funzione restituisce un numero che identifica il tipo di area dati della variabile dichiarata nel progetto.

I valori restituiti possono essere:
 0 = enum_VAR_AREA_INPUT
 1 = enum_VAR_AREA_FLAG
 2 = enum_VAR_AREA_OUTPUT
 -2147220992 = enum_VAR_AREA_TYPE_E_UNKNOWN

Parameter	Description
None	None

Result enum eVariableAreaType

Example:

```

Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    MsgBox CStr(objRet.GetAreaType()), vbOkOnly, GetProjectTitle
    Set objRet = Nothing
End Sub

```

GetBitNumber, DBVarObjCmdTarget Function

Syntax GetBitNumber

Description La funzione restituisce un numero, per una variabile di tipo bit con indirizzo assoluto (area Flag, Input, Output), che identifica la sua posizione nel byte all'indirizzo assoluto della stessa. La posizione può assumere valori da 0 a 7. L'indirizzo assoluto della variabile può essere reperito mediante la funzione GetAddress(). Nel caso si tratti di una variabile non condivisa (senza indirizzo assoluto assegnato) oppure di un tipo differente dal Bit il valore tornato sarà 0.

Parameter	Description
-----------	-------------

None	None
------	------

Result Integer

Example:

```
'FirstBit has absolute address 3.2
'SecondBit has absolute address 3.5
Public Sub Click()
    Dim objRet1 As DBVarObjCmdTarget
    Dim objRet2 As DBVarObjCmdTarget
    Set objRet1 = GetVariableObject("FirstBit")
    Set objRet2 = GetVariableObject("SecondBit")
    'GetAddress return 3 - GetBitNumber return 2
    MsgBox "FirstBit has address " & CStr(objRet1.GetAddress()) & " and the
position is " & CStr(objRet1.GetBitNumber()), vbOkOnly, GetProjectTitle
    'GetAddress return 3 - GetBitNumber return 5
    MsgBox "SecondBit has address " & CStr(objRet2.GetAddress()) & " and the
position is " & CStr(objRet2.GetBitNumber()), vbOkOnly, GetProjectTitle
    Set objRet1 = Nothing
    Set objRet2 = Nothing
End Sub
```

GetDataLoggerListNum, DBVarObjCmdTarget Function

Syntax GetDataLoggerListNum

Description La funzione restituisce il numero di Data Logger relativi alla variabile referenziata.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    MsgBox "GetDataLoggerListNum: " & CStr(objRet.GetDataLoggerListNum()),
vbOkOnly, GetProjectTitle
    Set objRet = Nothing
End Sub
```

GetDataLoggerName, DBVarObjCmdTarget Function

Syntax GetDataLoggerName(_nIndex)

Description La funzione restituisce il nome del Data Logger i-esimo all'interno del quale la variabile è memorizzata.

Parameter	Description
nIndex as Long	i-esimo Data Logger all'interno del quale viene memorizzato il valore della variabile referenziata.

Result String

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Dim numObj As Variant
    Set objRet = GetVariableObject("VAR00001")
    numObj = objRet.GetDataLoggerListNum()
    If numObj > 0 Then
        MsgBox "DataLoggerName 0: " & objRet.GetDataLoggerName(0),
        vbOkOnly, GetProjectTitle
    End If
    Set objRet = Nothing
End Sub
```

GetEventListName, DBVarObjCmdTarget Function

Syntax GetEventListName(_nIndex)

Description La funzione restituisce il nome della i-esima lista evento relativa alla variabile referenziata.

Parameter	Description
nIndex as Long	i-esima lista evento relativa alla variabile referenziata.

Result String

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    If objRet.GetEventListNum() > 0 Then
        MsgBox "GetEventListName: " & CStr(objRet.GetEventListName(0)),
        vbOkOnly, GetProjectTitle
    End If
    Set objRet = Nothing
End Sub
```

GetEventListNum, DBVarObjCmdTarget Function

Syntax GetEventListNum

Description La funzione restituisce il numero di liste eventi relative alla variabile referenziata.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    MsgBox "GetAlarmListNum: " & CStr(objRet.GetEventListNum()), vbOkOnly,
    GetProjectTitle
    Set objRet = Nothing
End Sub
```

GetEventObject, DBVarObjCmdTarget Function

Syntax GetEventObject

Description La funzione restituisce l'oggetto evento identificato dal nome in formato stringa (parametro lpszEventName).

Parameter	Description
lpszEventName as String	nome dell'evento da reperire.

Result Object
Viene restituito un oggetto di tipo EventCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()
    Dim objRet As DBVarObjCmdTarget
    Dim objEvent As EventCmdTarget

    Set objRet = GetVariableObject("VAR00001")
    Set objEvent = objRet.GetEventObject("EventoVar00001") 'supposto il nome
    dell'evento associato a VAR00001 sia EventoVar00001
    If Not objEvent Is Nothing Then
        Debug.Print objEvent.Enabled
        Set objEvent = Nothing
    End If
    Set objRet = Nothing
End Sub
```

GetInitialTimeInUse, DBVarObjCmdTarget Function

Syntax GetInitialTimeInUse

Description La funzione restituisce la data e l'orario del momento in cui la variabile è entrata in uso nel progetto.

Parameter	Description
None	None

Result Date

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Dim dateInitUse As Date
    Dim dateLastUse As Date
    Dim dateNotInUse As Date
    Set objRet = GetVariableObject("VAR00001")
    dateInitUse = objRet.GetInitialTimeInUse()
    dateLastUse = objRet.GetLastTimeInUse()
    dateNotInUse = objRet.GetTimeNotInUse()
    MsgBox "Init = " & Format(dateInitUse,"hh.nn.ss - dd/mm/yyyy") & vbCrLf & _
        "Last = " & Format(dateLastUse,"hh.nn.ss - dd/mm/yyyy") & vbCrLf & _
        "Not in use = " & Format(dateNotInUse,"hh.nn.ss - dd/mm/yyyy"), _
        vbOkOnly, "Test Use"
    Set objRet = Nothing
End Sub
```

GetInUseCount, DBVarObjCmdTarget Function

Syntax GetInUseCount

Description La funzione restituisce il numero degli oggetti che usano la variabile referenziata.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    MsgBox CStr(objRet.GetInUseCount()), vbOkOnly, GetProjectTitle
    Set objRet = Nothing
End Sub
```

GetInUseObjectAt, DBVarObjCmdTarget Function

Syntax GetInUseObjectAt(_nIndex)

Description La funzione restituisce l'oggetto che usa la variabile referenziata all'indice desiderato.

Parameter	Description
nIndex as Long	i-esimo oggetto che ha in uso la variabile referenziata.

Result Object
Viene restituito un oggetto di tipo DrawCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()
    Dim objRet As DBVarObjCmdTarget
    Dim nobj As Variant
    Dim objRet1 As DrawCmdTarget
    Set objRet = GetVariableObject("VAR1")
    If Not objRet Is Nothing Then
        Debug.Print "InUseCount is -> " & objRet.GetInUseCount()
        nobj = objRet.GetInUseCount()
        For i = 1 To nobj Step 1
            On Error Resume Next
            Set objRet1 = objRet.GetInUseObjectAt(i)
            If Not objRet1 Is Nothing Then
                'do something with object
                Debug.Print "Object_i name is -> " & objRet1.ObjectName
            Set objRet1 = Nothing
        End If
        On Error GoTo 0
    Next i
    Set objRet = Nothing
End If
End Sub
```

GetInUseObjectNameAt, DBVarObjCmdTarget Function

Syntax GetInUseObjectNameAt(_nIndex)

Description La funzione restituisce il nome degli oggetti che usano la variabile referenziata all'indice desiderato.

Parameter	Description
nIndex	i-esimo oggetto che ha in uso la variabile referenziata.

Result Long

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Dim nobj As Variant
    Set objRet = GetVariableObject("VAR00001")
    nobj = objRet.GetInUseCount()
    MsgBox "In uso " & CStr(objRet.GetInUseCount()) & " volte", vbOkOnly,
    GetProjectTitle
    For i = 0 To (nobj-1) Step 1
        MsgBox CStr(objRet.GetInUseObjectNameAt(i)), vbOkOnly, GetProjectTitle
    Next i
    Set objRet = Nothing
End Sub
```

GetLastTimeInUse, DBVarObjCmdTarget Function

Syntax GetLastTimeInUse

Description La funzione restituisce la data e l'orario del momento in cui la variabile è stata utilizzata l'ultima volta all'interno del progetto.

Parameter	Description
None	None

Result Date

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Dim dateInitUse As Date
    Dim dateLastUse As Date
    Dim dateNotInUse As Date
    Set objRet = GetVariableObject("VAR00001")
    dateInitUse = objRet.GetInitialTimeInUse()
    dateLastUse = objRet.GetLastTimeInUse()
    dateNotInUse = objRet.GetTimeNotInUse()
    MsgBox "Init = " & Format(dateInitUse,"hh.nn.ss - dd/mm/yyyy") & vbCrLf & _
        "Last = " & Format(dateLastUse,"hh.nn.ss - dd/mm/yyyy") & vbCrLf & _
        "Not in use = " & Format(dateNotInUse,"hh.nn.ss - dd/mm/yyyy"), _
        vbOkOnly, "Test Use"
    Set objRet = Nothing
End Sub
```

GetMemberObjectFromIndex, DBVarObjCmdTarget Function

Syntax GetMemberObjectFromIndex(_nIndex)

Description Questa funzione permette di reperire la variabile membro all'interno di una variabile di tipo struttura a partire dalla posizione passata come parametro all'interno della stessa. Se la variabile di partenza non è di tipo struttura o alla posizione passata come parametro non è definita variabile membro, l'oggetto restituito è Nothing.

Parameter	Description
nIndex As Long	Nome membro.

Result Object
Viene restituito un oggetto di tipo DBVarObjCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()
    Dim objRet As DBVarObjCmdTarget
    Dim Member0 As DBVarObjCmdTarget
    Set objRet = GetVariableObject("_SysVar_")
    Set Member0 = objRet.GetMemberObjectFromIndex(0)
    If Not Member0 Is Nothing Then
        Debug.Print "Member0 name is -> " & Member0.GetName 'return
        SimSinDouble
    End If
End Sub
```

```

        Set Member0 = Nothing
    End If
    Set objRet = Nothing
End Sub

```

GetMemberObjectFromName, DBVarObjCmdTarget Function

Syntax GetMemberObjectFromName(_IpszMember)

Description Questa funzione permette di reperire la variabile membro all'interno di una variabile di tipo struttura a partire dal nome della variabile membro passata come parametro. Se la variabile di partenza non è di tipo struttura o il nome passato come parametro non si riferisce a nessuna variabile membro, l'oggetto restituito è Nothing.

Parameter	Description
IpszMember As String	Nome membro.

Result Object
Viene restituito un oggetto di tipo DBVarObjCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```

Public Sub Click()
    Dim objRet As DBVarObjCmdTarget
    Dim Member0 As DBVarObjCmdTarget
    Set objRet = GetVariableObject("_SysVar_")
    Set Member0 = objRet.GetMemberObjectFromName("SimSinDouble")
    If Not Member0 Is Nothing Then
        Debug.Print "Member0 type is -> " & Member0.GetType 'return
        enum_VAR_TYPE_DOUBLE = 8
        Set Member0 = Nothing
    End If
    Set objRet = Nothing
End Sub

```

GetName, DBVarObjCmdTarget Function

Syntax GetName

Description La funzione restituisce il nome simbolico della variabile dichiarata nel progetto.

Parameter	Description
None	None

Result String

Example:

```

Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    MsgBox objRet.GetName(), vbOkOnly, GetProjectTitle
    Set objRet = Nothing
End Sub

```

End Sub

GetNumObjectsInHeap, DBVarObjCmdTarget Function

Syntax GetNumObjectsInHeap

Description

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    MsgBox CStr(objRet.GetNumObjectsInHeap()), vbOkOnly, "Test
    GetNumObjectsInHeap"
    Set objRet = Nothing
End Sub
```

GetStructName, DBVarObjCmdTarget Function

Syntax GetStructName

Description La funzione restituisce il nome simbolico del prototipo di struttura della variabile dichiarata nel progetto. Se la variabile referenziata non è di tipo struttura viene restituita una stringa vuota.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    If CStr(objRet.GetType()) = 11 Then ' 11 = Struct type
        MsgBox CStr(objRet.GetStructName()), vbOkOnly, "Test
        GetStructName"
    End If
    Set objRet = Nothing
End Sub
```

GetStructParentObject, DBVarObjCmdTarget Function

Syntax GetStructParentObject

Description La funzione restituisce il puntatore all'oggetto struttura relativo alla variabile referenziata.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo DBVarObjCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Dim objParent as DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    If CStr(objRet.GetType()) = 11 Then ' 11 = Struct type
        Set objParent = objRet.GetStructParentObject()
        'do something
    End If
    Set objParent = Nothing
    Set objRet = Nothing
End Sub
```

GetTimeNotInUse, DBVarObjCmdTarget Function

Syntax GetTimeNotInUse

Description La funzione restituisce la data e l'orario del momento in cui la variabile ha cessato di essere utilizzata all'interno del progetto (quando il progetto è stato fermato la volta precedente).

Parameter	Description
None	None

Result Date

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Dim dateInitUse As Date
    Dim dateLastUse As Date
    Dim dateNotInUse As Date
    Set objRet = GetVariableObject("VAR00001")
    dateInitUse = objRet.GetInitialTimeInUse()
    dateLastUse = objRet.GetLastTimeInUse()
    dateNotInUse = objRet.GetTimeNotInUse()
    MsgBox "Init = " & Format(dateInitUse,"hh.nn.ss - dd/mm/yyyy") & vbCrLf & _
        "Last = " & Format(dateLastUse,"hh.nn.ss - dd/mm/yyyy") & vbCrLf & _
        "Not in use = " & Format(dateNotInUse,"hh.nn.ss - dd/mm/yyyy"), _
        vbOkOnly, "Test Use"
```

```
        Set objRet = Nothing
End Sub
```

GetTimeStamp, DBVarObjCmdTarget Function

Syntax GetTimeStamp

Description La funzione restituisce la data e l'orario dell'ultimo aggiornamento della variabile.

Parameter	Description
None	None

Result Date

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Dim dTimeStamp As Date

    Set objRet = GetVariableObject("VAR00001")
    dTimeStamp = objRet.GetTimeStamp()
    MsgBox "TimeStamp = " & Format(dTimeStamp,"hh.nn.ss - dd/mm/yyyy"),vbOkOnly, "Test Use"
    Set objRet = Nothing
End Sub
```

GetTimeStampMS, DBVarObjCmdTarget Function

Syntax GetTimeStampMS

Description La funzione restituisce il tempo in millisecondi dell'ultimo aggiornamento della variabile.

Parameter	Description
None	None

Result Integer

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Dim dTimeStamp As Date
    Dim nMS As Integer
    Set objRet = GetVariableObject("VAR00001")
    dTimeStamp = objRet.GetTimeStamp()
    nMS = objRet.GetTimeStampMS()
    MsgBox "TimeStamp = " & Format(dTimeStamp,"hh.nn.ss - dd/mm/yyyy - ") & CStr(nMS) & " milliseconds",vbOkOnly, "Test Use"
```

```
Set objRet = Nothing
End Sub
```

GetType, DBVarObjCmdTarget Function

Syntax GetType()

Description La funzione restituisce un numero che identifica il tipo della variabile dichiarata nel progetto.

I valori restituiti possono essere:

```
0 = enum_VAR_TYPE_BIT
1 = enum_VAR_TYPE_SIGNBYTE
2 = enum_VAR_TYPE_BYTE
3 = enum_VAR_TYPE_SIGNWORD
4 = enum_VAR_TYPE_WORD
5 = enum_VAR_TYPE_SIGNDWORD
6 = enum_VAR_TYPE_DWORD
7 = enum_VAR_TYPE_FLOAT
8 = enum_VAR_TYPE_DOUBLE
9 = enum_VAR_TYPE_STRING
10 = enum_VAR_TYPE_ARRAY
11 = enum_VAR_TYPE_STRUCT
-2147220992 = enum_VAR_TYPE_E_UNKNOWN
```

Parameter	Description
None	None

Result enum eVariableType

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    MsgBox CStr(objRet.GetType()), vbOkOnly, "Test GetType"
    Set objRet = Nothing
End Sub
```

GetXMLSettings, DBVarObjCmdTarget Function

Syntax GetXMLSettings

Description La funzione restituisce la stringa di definizione della variabile nel progetto in formato XML.

Parameter	Description
None	None

Result String

Example:

```

Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    MsgBox CStr(objRet.GetXMLSettings()), vbOkOnly, GetProjectTitle
    Set objRet = Nothing
End Sub

```

IsOPCServerEnabled, DBVarObjCmdTarget Function

Syntax IsOPCServerEnabled

Description La funzione restituisce il valore booleano True se è stato abilitato l'OPC Server sia sul Real Time DB che nelle proprietà "Opzione" della variabile in esame.



Questa funzione non è supportata in Windows CE.(se utilizzata restituisce sempre false)

Parameter	Description
None	None

Result Boolean

Example:

```

Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    MsgBox CStr(objRet.IsOPCServerEnabled()), vbOkOnly, "Test IsOPCServerEnabled"
    Set objRet = Nothing
End Sub

```

IsOPCServerOnRequest, DBVarObjCmdTarget Function

Syntax IsOPCServerOnRequest

Description La funzione restituisce il booleano True se la variabile in esame è pubblicata nello spazio dell'OPC Server e se qualche Client vi è collegato.



Questa funzione non è supportata in Windows CE.(se utilizzata restituisce sempre false)

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    MsgBox CStr(objRet.IsOPCServerOnRequest()), vbOkOnly, GetProjectTitle
    Set objRet = Nothing
End Sub
```

IsShared, DBVarObjCmdTarget Function

Syntax IsOPCServerEnabled

Description La funzione restituisce il booleano True se l'area di memoria della variabile è stata definita come condivisa (Input, Output, Flag).

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    MsgBox CStr(objRet.IsShared()), vbOkOnly, GetProjectTitle
    Set objRet = Nothing
End Sub
```

IsValid, DBVarObjCmdTarget Function

Syntax IsValid

Description La funzione restituisce il booleano True se a variabile ha effettivamente un valore valido.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    MsgBox CStr(objRet.IsValid()), vbOkOnly, GetProjectTitle
    Set objRet = Nothing
End Sub
```

ResetStatisticData, DBVarObjCmdTarget Function

Syntax ResetStatisticData

Description Questa funzione consente di azzerare la statistica dati per la variabile in questione. Tutti i valori verranno ricalcolati a partire dal momento in cui viene richiamato il reset.

Parameter	Description
None	None

Result None

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Set objRet = GetVariableObject("VAR00001")
    objRet.ResetStatisticData()
    Set objRet = Nothing
End Sub
```

SetOPCServerEnabled, DBVarObjCmdTarget Function

Syntax SetOPCServerEnabled(_newVal)

Description Questa funzione consente di decidere se pubblicare una variabile nell'address space dell'OPC Server DA. Il valore di ritorno indica che l'operazione è avvenuta con successo e la variabile sarà effettivamente pubblicata nell'OPC Server. La funzione ritorna 'False' nel caso in cui venga eseguita quando l'OPC Server DA è già in esecuzione; di conseguenza si può utilizzare solo durante la fase di startup del progetto e quindi soltanto nel basic script che è stato impostato come script di startup.



Il basic script di startup viene eseguito in modo sincrono rispetto alle risorse del progetto avviate (anche se è attiva l'opzione Thread Separato) attendendo al massimo un tempo uguale al valore impostato come timeout sincrono nelle proprietà della risorsa basic script stessa. Di conseguenza potrebbe essere necessario aumentare questo valore se occorre utilizzare questa funzione in uno script che richiede diverso tempo di elaborazione prima che riprenda l'avvio delle restanti risorse di Movicon.

Parameter	Description
newVal As Boolean	True : La variabile sarà pubblicata nell'address space dell'opc server. False : La variabile non sarà pubblicata nell'address space dell'opc server.

Result Boolean

Example:

Option Explicit

```

Sub Main
    Dim objVar As DBVarObjCmdTarget
    Set objVar = CreateNewVar("VAR00002", enum_VAR_TYPE_DWORD, "") 'creo la
    variabile
    If Not objVar Is Nothing Then
        If objVar.IsValid Then
            objVar.DynamicSettings = "[DRV]Modbus TCPIP.Sta=Default
            Station|Unit=1|FC=2|SA=0"
            objVar.SetOPCServerEnabled(True)
        Else
            Debug.Print Replace("The variabile '%s' is not valid!", "%s",
            objVar.GetName)
        End If
    Else
        Debug.Print "Failed to create the variabile 'RuntimeVariable'"
    End If
End Sub

```

Example2:

Option Explicit

Sub Main

```

    Dim objVar As DBVarObjCmdTarget
    Dim bRet As Boolean
    Set objVar = GetVariableObject("VAR00001") 'creo il riferimento alla variabile
    If Not objVar Is Nothing Then
        If objVar.IsValid Then
            objVar.DynamicSettings = "[DRV]Modbus TCPIP.Sta=Default
            Station|Unit=1|FC=2|SA=1"
            bRet = objVar.SetOPCServerEnabled(False)
            If bRet Then
                Debug.Print Replace("Variabile '%s' released from the OPC Server
                DA e/o XML address space!", "%s", objVar.GetName)
            Else
                Debug.Print Replace("Failed to release the variabile '%s' from the
                OPC Server DA e/o XML address space!", "%s", objVar.GetName)
            End If
        Else
            Debug.Print Replace("The variabile '%s' is not valid!", "%s",
            objVar.GetName)
        End If
    Else
        Debug.Print "Failed to create a reference to the variabile '" & objVar.GetName &
        """"
    End If
End Sub

```

SetStructName, DBVarObjCmdTarget Function

Syntax SetStructName(BSTR newVal)

Description Imposta il nome del Prototipo di Struttura per la Variabile se il suo tipo è VAR_TYPE_STRUCT e se viene eseguita a Design time, ad esempio da una Risorsa VB Script o dal Dropping Code della Libreria Simboli. Restituisce TRUE se l'impostazione ha successo, FALSE se il tipo della variabile non è VAR_TYPE_STRUCT o se viene eseguita a runtime.

Parameter	Description
_newVal As String	Nome del Prototipo Struttura

Result Boolean

Example:

```

Option Explicit
Const NEW_STRUCT_NAME As String = "NewStructurePrototipeName"
Sub Main()
Dim objVar As DBVarObjCmdTarget
Dim bRet As Boolean
Set objVar = GetVariableObject("VAR00001")
If objVar.GetType = enum_VAR_TYPE_STRUCT Then
    bRet = objVar.SetStructName(STRUCT_NAME)

    If Not bRet Then Debug.Print "SetStructName() Function filed!"
End If
Set objVar = Nothing
End Sub

```

SetTimeStamp, DBVarObjCmdTarget Function

Syntax SetTimeStamp()

Description Questa funzione consente di impostare il TimeStamp della variabile con la data attuale del sistema.

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub Click()
Dim objVar As DBVarObjCmdTarget
Set objVar = GetVariableObject("VAR00001")
objVar.SetTimeStamp()
MsgBox "Variable TimeStamp = " & CStr(objVar.GetTimeStamp()),
vbInformation, GetProjectTitle
Set objVar = Nothing
End Sub

```

SetTimeStampFromDate, DBVarObjCmdTarget Function

Syntax SetTimeStampFromDate(_NewDate, _nMSVal)

Description Imposta il valore del TimeStamp con una Data, Ora e Millisecondi sulla variabile

Parameter	Description
_nMSVal As Integer	Valore in millisecondi
_NewDate As Date	Valore di Data e Ora ad esempio "Now"

Result None

Example:

Option Explicit

```
Public Sub Click()  
Dim objVar As DBVarObjCmdTarget
```

```
    Set objVar = GetVariableObject("VAR00001")
```

```
    objVar.SetTimeStampFromDate(Now,333)
```

```
    Set objVar = Nothing
```

```
End Sub
```

SetType, DBVarObjCmdTarget Function

Syntax SetType(enum_VariableType nType)

Description Imposta il tipo della variabile.
E' efficace solo se eseguita a Design time quindi dalla Risorsa VB Script o dal Dropping Code della Libreria Simboli.
Imposta valori corrispondenti ai tipi di Movicon [VAR_TYPE_BIT, VAR_TYPE_STRUCT].
Ritorna TRUE se l'impostazione va a buon fine.

Parameter		Description
_nType enum_VariableType	As	Valore enumerale del tipo di variabile

Result Boolean

Example:

```
Option Explicit  
Sub Main()  
Dim objVar As DBVarObjCmdTarget  
Dim bRet As Boolean  
    Set objVar = GetVariableObject("VAR00001")  
    bRet = objVar.SetType(enum_VAR_TYPE_WORD)  
    If Not bRet Then Debug.Print "SetType() Function filed!"  
    Set objVar = Nothing  
End Sub
```

Properties

AviFileProp, DBVarObjCmdTarget Property

Syntax AviFileProp = _String

Description Questa proprietà consente di leggere o impostare la stringa associata alla proprietà "File Avi" di una variabile. Una volta impostata potrà essere interpretata dall'OPC Client connesso se dispone delle funzionalità necessarie.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget
    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Avi file Prop = " & objVar.AviFileProp,vbInformation,GetProjectTitle
    Set objVar = Nothing
End Sub
```

BGColorProp, DBVarObjCmdTarget Property

Syntax BGColorProp = _Long

Description Questa proprietà consente di leggere o impostare per la variabile selezionata il colore di sfondo iniziale. Questa proprietà potrà essere interpretata dall'OPC Client connesso se dispone delle funzionalità necessarie.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget
    Set objVar = GetVariableObject("VAR00001")
    MsgBox "BGColor Prop = " & objVar.BGColorProp,vbInformation,GetProjectTitle
    Set objVar = Nothing
End Sub
```

BlinkProp, DBVarObjCmdTarget Property

Syntax TRACEAddMsgLog = _Boolean

Description Questa proprietà consente di leggere o impostare lo stato di lampeggio iniziale. Questa proprietà potrà essere interpretata dall'OPC Client connesso se dispone delle funzionalità necessarie.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget
```

```

Set objVar = GetVariableObject("VAR00001")
MsgBox "Blink Prop = " & objVar.BlinkProp,vbInformation,GetProjectTitle
Set objVar = Nothing
End Sub

```

BmpFileProp, DBVarObjCmdTarget Property

Syntax BmpFileProp = _String

Description Questa proprietà consente di leggere o impostare la stringa associata alla proprietà "File Bitmap" di una variabile. Una volta impostata potrà essere interpretata dall'OPC Client connesso se dispone delle funzionalità necessarie.

Parameter	Description
None	None

Result String

Example:

```

Sub Main
    Dim objVar As DBVarObjCmdTarget
    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Bitmap file Prop = " &
    objVar.BmpFileProp,vbInformation,GetProjectTitle
    Set objVar = Nothing
End Sub

```

CloseBitString, DBVarObjCmdTarget Property

Syntax CloseBitString = _String

Description Questa proprietà consente di leggere o impostare la stringa associata alla proprietà "Stringa per contatto Chiuso" di una variabile. In pratica è la stringa che in runtime verrà associata allo stato logico "1" della variabile.

Parameter	Description
None	None

Result String

Example:

```

Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    objVar.CloseBitString = "CLOSE"
    Set objVar = Nothing
End Sub

```


DynamicSettings, DBVarObjCmdTarget Property

Syntax DynamicSettings = _String

Description Questa proprietà consente di leggere o impostare la stringa di connessione dinamica per la variabile specificata. Corrisponde alla proprietà "Indirizzo Dinamico" di una variabile.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    objVar.DynamicSettings = "[DRV]Modbus
    TCPIP.Sta=Station1|Unit=1|FC=2|SA=100"
    MsgBox "Dynamic Settings = " &
    objVar.DynamicSettings,vbInformation,GetProjectTitle
    Set objVar = Nothing
End Sub
```

EnableFactor, DBVarObjCmdTarget Property

Syntax EnableFactor = _Boolean

Description Questa proprietà consente di leggere o impostare la funzionalità di Normalizzazione Dato di una variabile collegata al Driver di Comunicazione. Se abilitata per le variabili che arrivano dal campo, verrà letto e scritto direttamente il valore scalato sulla variabile. La normalizzazione avviene sulla base delle impostazioni dei valori delle proprietà FactorGain e FactorOffset.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim objVar As DBVarObjCmdTarget
    Set objVar = GetVariableObject("VAR00001")
    MsgBox("EnableFactor is: " & objVar.EnableFactor,vbOkOnly,GetProjectTitle)
    Set objVar = Nothing
End Sub
```

EnableNetworkServer, DBVarObjCmdTarget Property

Syntax EnableNetworkServer= _Boolean

Description Abilitando questa proprietà è possibile rendere disponibile la variabile per la condivisione con altri applicativi Movicon tramite le funzionalità di Networking.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objVar As DBVarObjCmdTarget  
    Set objVar = GetVariableObject("VAR00001")  
    MsgBox("EnableScalingFactor is: " &  
        objVar.EnableNetworkServer,vbOkOnly,GetProjectTitle)  
    Set objVar = Nothing  
End Sub
```

EnableScalingFactor, DBVarObjCmdTarget Property

Syntax EnableScalingFactor = _Boolean

Description Questa proprietà consente di leggere o impostare la funzionalità di Normalizzazione Dato di una variabile collegata al Driver di Comunicazione. Se abilitata per le variabili che arrivano dal campo, verrà letto e scritto direttamente il valore scalato sulla variabile. La normalizzazione avviene sulla base delle impostazioni dei valori 'Valore Max. Normalizzato', 'Valore Min. Normalizzato', 'Valore Max. Non Normalizzato', 'Valore Min. Non Normalizzato'.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objVar As DBVarObjCmdTarget  
    Set objVar = GetVariableObject("VAR00001")  
    MsgBox("EnableScalingFactor is: " &  
        objVar.EnableScalingFactor,vbOkOnly,GetProjectTitle)  
    Set objVar = Nothing  
End Sub
```

EngineeringUnit, DBVarObjCmdTarget Property

Syntax EngineeringUnit = _String

Description Questa proprietà consente di leggere o impostare la stringa associata alla proprietà "Unità Ingegneristica" di una variabile. In pratica rappresenta l'unità di misura che verrà visualizzata insieme al valore della variabile.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    objVar.EngineeringUnit = "mm"
    Set objVar = Nothing
End Sub
```

FactorGain, DBVarObjCmdTarget Property

Syntax FactorGain = _Double

Description Questa proprietà consente di leggere o impostare il valore di gain per la normalizzazione della variabile.

Parameter	Description
None	None

Result Double

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    objVar.FactorGain = 10
    Set objVar = Nothing
End Sub
```

FactorOffset, DBVarObjCmdTarget Property

Syntax FactorOffset = _Double

Description Questa proprietà consente di leggere o impostare il valore di offset per la normalizzazione della variabile.

Parameter	Description
None	None

Result Double

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    objVar.FactorOffset = 5
    Set objVar = Nothing
End Sub
```

FGColorProp, DBVarObjCmdTarget Property

Syntax FGColorProp = _Long

Description Questa proprietà consente di leggere o impostare per la variabile selezionata il colore di contorno iniziale. Questa proprietà potrà essere interpretata dall'OPC Client connesso se dispone delle funzionalità necessarie.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget
    Set objVar = GetVariableObject("VAR00001")
    MsgBox "FGColor Prop = " & objVar.FGColorProp, vbInformation, GetProjectTitle
    Set objVar = Nothing
End Sub
```

Group, DBVarObjCmdTarget Property

Syntax Group = _String

Description Questa proprietà restituisce l'eventuale gruppo di variabili di appartenenza ('cartella di appartenenza') della variabile referenziata.

Parameter	Description
None	None

Result String

Example:

```

Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Group = " & objVar.Group,vbInformation,GetProjectTitle
End Sub

```

HtmlFileProp DBVarObjCmdTarget Property

Syntax HtmlFileProp = _String

Description Questa proprietà consente di leggere o impostare la stringa associata alla proprietà "File Html" di una variabile. Una volta impostata potrà essere interpretata dall'OPC Client connesso se dispone delle funzionalità necessarie.

Parameter	Description
None	None

Result String

Example:

```

Sub Main
    Dim objVar As DBVarObjCmdTarget
    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Html file Prop = " &
    objVar.HtmlFileProp,vbInformation,GetProjectTitle
    Set objVar = Nothing
End Sub

```

InheritQuality, DBVarObjCmdTarget Property

Syntax InheritQuality = _Boolean

Description Tramite questa proprietà è possibile leggere o impostare la proprietà "Eredita Qualità" della variabile struttura tramite la quale è possibile fare in modo che la qualità della variabile struttura venga aggiornata in base alla qualità dei singoli membri.
Dopo che la proprietà è stata abilitata la qualità della variabile struttura verrà modificata solo al prossimo variare della qualità di uno dei membri.

Parameter	Description
None	None

Result Boolean

Example:

```

Dim objVar As DBVarObjCmdTarget
Public Sub SymbolLoading()
    Set objVar = GetVariableObject("Struct1")
    objVar.InheritQuality = Not objVar.InheritQuality
    Set objVar = Nothing
End Sub

```

InUse, DBVarObjCmdTarget Property

Syntax InUse = _Boolean

Description Questa proprietà consente di leggere o impostare lo stato "In Uso" della variabile specificata. La forzatura dello stato In Uso della variabile verrà gestita dal driver di comunicazione o dall'OPC ecc. Ad esempio forzando la proprietà a "True" la variabile risulterà sempre in uso e quindi verrà mantenuta aggiornata dal driver di comunicazione. Viceversa forzando la proprietà a "False" la variabile risulterà sempre non in uso e quindi verrà mantenuta aggiornata dal driver di comunicazione in base ai tempi di refresh per le variabili non in uso.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "In Use = " & objVar.InUse,vbInformation,GetProjectTitle
End Sub
```

InverseFactor, DBVarObjCmdTarget Property

Syntax InverseFactor = _Boolean

Description Abilitando questa proprietà verrà eseguita la normalizzazione della variabile con fattore inverso.

Se InverseFactor assume il valore Booleano True l'istruzione sarà:
 $(\text{Value} - \text{Offset}) / \text{Gain}$

Se InverseFactor assume il valore Booleano False l'istruzione sarà:
 $(\text{Value} * \text{Gain}) + \text{Offset}$

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim objVar As DBVarObjCmdTarget
    Set objVar = GetVariableObject("VAR00001")
    MsgBox("EnableScalingFactor is: " &
        objVar.InverseFactor,vbOkOnly,GetProjectTitle)
    Set objVar = Nothing
End Sub
```

InverseScaling, DBVarObjCmdTarget Property

Syntax InverseScaling = _Boolean

Description Abilitando questa proprietà verrà eseguita la normalizzazione della variabile con un andamento inverso rispetto a quello definito. Ad esempio supponendo di impostare :

Valore Max. Non Normalizzato = 100
Valore Min. Non Normalizzato = 0
Valore Max. Normalizzato = 1000
Valore Min. Normalizzato = 0

Quando la variabile assumerà il valore reale 0 il valore scalato sarà 1000 e quando la variabile assumerà il valore 100 il valore scalato sarà 0.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objVar As DBVarObjCmdTarget  
    Set objVar = GetVariableObject("VAR00001")  
    MsgBox("EnableScalingFactor is: " & objVar.InverseScaling  
    ,vbOkOnly,GetProjectTitle)  
    Set objVar = Nothing  
End Sub
```

LastChangeComment, DBVarObjCmdTarget Property

Syntax LastChangeComment = _String

Description Questa proprietà restituisce l'ultimo commento inserito alla variazione della variabile. La proprietà ha significato se è stata abilitata la proprietà 'Trace Comment'.

Parameter	Description
None	None

Result String

Example:

```
Sub Main  
    Dim objVar As DBVarObjCmdTarget  
  
    Set objVar = GetVariableObject("VAR00001")  
    MsgBox "Last Change Comment = " &  
    objVar.LastChangeComment,vbInformation,GetProjectTitle  
End Sub
```

MapRealTimeODBCUpdateQuality, DBVarObjCmdTarget Property

Syntax MapRealTimeODBCUpdateQuality = _Boolean

Description Questa proprietà consente di leggere o impostare la funzionalità di Update Quality nella sezione Real Time ODBC (finestra Proprietà) per la variabile specificata.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Dim result As Boolean
    Set objRet = GetVariableObject("VAR00001")
    result = objRet.MapRealTimeODBCUpdateQuality ()
    MsgBox "MapRealTimeODBCUpdateQuality = " & result ,vbOkOnly,
    GetProjectTitle
    Set objRet = Nothing
End Sub
```

MapRealTimeToDB, DBVarObjCmdTarget Property

Syntax MapRealTimeToDB = _Boolean

Description Questa proprietà consente di leggere o impostare la funzionalità di condivisione con il Database adibito a questo scopo per la variabile specificata. In questo modo la variabile sarà resa disponibile su un Database e quindi accessibile anche da altre applicazioni. Corrisponde alla proprietà "Abilita" delle "Proprietà Real Time ODBC di una Variabile".



La proprietà non è gestita a Runtime ma solo a Sviluppo per esempio attraverso l'uso della programmazione avanzata dei Simboli Template inseriti nella Libreria dei Simboli (Vedi capitolo 'Dropping Code dei Template').

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget
```



```

Set objVar = GetVariableObject("VAR00001")
MsgBox "Map RealTime To DB = " &
objVar.MapRealTimeToDB,vbInformation,GetProjectTitle
End Sub

```

MapRealTimeToDBMode, DBVarObjCmdTarget Property

Syntax MapRealTimeToDBMode = _Byte

Description Questa proprietà consente di leggere o impostare la modalità con cui la variabile specificata verrà scambiata con il Database adibito a questo scopo. Corrisponde alla proprietà "Modalità" delle "Proprietà Real Time ODBC di una Variabile".

I valori di impostazione e di ritorno sono:

0 = Input
1 = Output
2 = Input/Output



La proprietà non è gestita a Runtime ma solo a Sviluppo per esempio attraverso l'uso della programmazione avanzata dei Simboli Template inseriti nella Libreria dei Simboli (Vedi capitolo 'Dropping Code dei Template').

Parameter	Description
None	None

Result Byte

Example:

```

Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Map RealTime To DB Mode = " &
objVar.MapRealTimeToDBMode,vbInformation,GetProjectTitle
End Sub

```

MapRealTimeToDBRefreshTime, DBVarObjCmdTarget Property

Syntax MapRealTimeToDBRefreshTime = _Long

Description Questa proprietà consente di leggere o impostare il tempo di rinfresco con il quale vengono lette le variabili dal Database associato. Corrisponde alla proprietà "Tempo di Rinfresco in Lettura" delle "Proprietà Real Time ODBC di una Variabile".



La proprietà non è gestita a Runtime ma solo a Sviluppo per esempio attraverso l'uso della programmazione avanzata dei Simboli Template inseriti nella Libreria dei Simboli (Vedi capitolo 'Dropping Code dei Template').

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Map RealTime To DB Refresh Time = " &
objVar.MapRealTimeToDBRefreshTime,vbInformation,GetProjectTitle
End Sub
```

NetworkClientEnable, DBVarObjCmdTarget Property

Syntax NetworkClientEnable = _Boolean

Description Questa proprietà consente di abilitare la connessione Networking tra la variabile del progetto locale ed un'altra stazione Movicon definita come Server.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Network Client Enable = " &
objVar.NetworkClientEnable,vbInformation,GetProjectTitle
End Sub
```

NetworkClientMode, DBVarObjCmdTarget Property

Syntax NetworkClientMode = _Byte

Description Questa proprietà consente di leggere o impostare la modalità con cui la variabile specificata verrà scambiata col Server in Networking. Corrisponde alla proprietà "Modalità" delle "Proprietà Client di Networking di una Variabile".

I valori di impostazione e di ritorno sono:

0 = Input

1 = Output
2 = Input/Output

Parameter	Description
None	None

Result Byte

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Network Client Mode = " &
    objVar.NetworkClientMode,vbInformation,GetProjectTitle
End Sub
```

NetworkClientServerName, DBVarObjCmdTarget Property

Syntax NetworkClientServerName = _String

Description Questa proprietà consente di leggere o impostare il nome del Server di Networking al quale la variabile specificata dovrà essere connessa. Corrisponde alla proprietà "Server di Networking" delle "Proprietà Client di Networking di una Variabile".

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Network Client Server Name = " &
    objVar.NetworkClientServerName,vbInformation,GetProjectTitle
End Sub
```

NetworkClientUpdateQuality, DBVarObjCmdTarget Property

Syntax NetworkClientUpdateQuality = _Boolean

Description Questa proprietà consente di leggere o impostare la funzionalità di Update Quality nella sezione Network Client (finestra Proprietà) per la variabile specificata.

Parameter	Description
-----------	-------------

None	None
------	------

Result Boolean

Example:

```
Sub Main
    Dim objRet As DBVarObjCmdTarget
    Dim result As Boolean
    Set objRet = GetVariableObject("VAR00001")
    result = objRet.NetworkClientUpdateQuality ()
    MsgBox "NetworkClientUpdateQuality = " & result ,vbOkOnly, GetProjectTitle
    Set objRet = Nothing
End Sub
```

NetworkServerEnable, DBVarObjCmdTarget Property

Syntax NetworkServerEnable = _Boolean

Description Questa proprietà consente di leggere o impostare la funzionalità di Network Server per la variabile specificata. In questo modo la variabile sarà resa disponibile alla connessione da eventuali Client di rete. Corrisponde alla proprietà "Abilitazione Network Server" delle "Proprietà Opzioni di una Variabile".

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Network Server Enable = " &
objVar.NetworkServerEnable,vbInformation,GetProjectTitle
End Sub
```

NetworkServerIsWritable, DBVarObjCmdTarget Property

Syntax NetworkServerIsWritable = _Boolean

Description Questa proprietà consente di leggere o impostare la funzionalità di scrittura della variabile sul Server specificata da parte degli eventuali Client che si connettono.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Network Server Is Writable = " &
    objVar.NetworkServerIsWritable,vbInformation,GetProjectTitle
End Sub
```

OPCGroupName, DBVarObjCmdTarget Property

Syntax OPCGroupName = _String

Description Questa proprietà consente di leggere o impostare il nome del gruppo OPC di appartenenza (nella struttura di progetto) della variabile referenziata.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    Debug.Print objVar.OPCGroupName
End Sub
```

OPCServerAccessRights,DBVarObjTarget Property

Syntax EnableScalingFactor = eOPCServerMode

Description Questa proprietà permette di selezionare il tipo di accesso ad un item OPC Server di Movicon da parte dei Client OPC.
I valori consentiti possono essere specificati utilizzando l'enumeratore eOPCServerMode oppure inserendo il corrispondente valore numerico:

enum_opc_readable (valore 1, leggibile)
enum_opc_writeable (valore 2, scrivibile)
enum_opc_readable_writeable (valore 3, leggibile-scrivibile)

Nel caso venga impostato il tipo di accesso da "leggibile-scrivibile" a "leggibile", un Client OPC potrà solo leggere il valore, ma nella proprietà dell'item questo risulterà ancora pubblicato come lettura e scrittura.

Parameter	Description
-----------	-------------

None	None
------	------

Result eOPCServerMode

Example:

```
Sub Main
    Dim var1 As DBVarObjCmdTarget
    Set var1=GetVariableObject("VAR00001")
    var1.OPCServerAccessRights = enum_opc_readable
    Set var1=Nothing
End Sub
```

OpenBitString, DBVarObjCmdTarget Property

Syntax OpenBitString = _String

Description Questa proprietà consente di leggere o impostare la stringa associata alla proprietà "Stringa per contatto Aperto" di una variabile. In pratica è la stringa che in runtime verrà associata allo stato logico "0" della variabile.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    objVar.OpenBitString = "OPEN"
End Sub
```

Quality, DBVarObjCmdTarget Property

Syntax Quality = _Integer

Description Questa proprietà consente di leggere lo stato di qualità della variabile specificata. La proprietà è di sola lettura.

I valori restituiti sono i valori di qualità come da specifiche OPC:

```
0 = OPC_QUALITY_BAD
64 = OPC_QUALITY_UNCERTAIN
192 = OPC_QUALITY_GOOD
4 = OPC_QUALITY_CONFIG_ERROR
8 = OPC_QUALITY_NOT_CONNECTED
12 = OPC_QUALITY_DEVICE_FAILURE
16 = OPC_QUALITY_SENSOR_FAILURE
20 = OPC_QUALITY_LAST_KNOWN
24 = OPC_QUALITY_COMM_FAILURE
28 = OPC_QUALITY_OUT_OF_SERVICE
68 = OPC_QUALITY_LAST_USABLE
80 = OPC_QUALITY_SENSOR_CAL
```

84 = OPC_QUALITY_EGU_EXCEEDED
88 = OPC_QUALITY_SUB_NORMAL
216 = OPC_QUALITY_LOCAL_OVERRIDE

Parameter	Description
None	None

Result Integer

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "VAR00001 = " & objVar.Quality, vbInformation, GetProjectTitle
End Sub
```

ScaleMax, DBVarObjCmdTarget Property

Syntax ScaleMax = _Double

Description Questa proprietà consente di leggere o impostare il valore massimo normalizzato della variabile specificata. Corrisponde alla proprietà "Valore max normalizzato" di una variabile.

Parameter	Description
None	None

Result Double

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Scale Max = " & objVar.ScaleMax, vbInformation, GetProjectTitle
    Set objVar = Nothing
End Sub
```

ScaleMin, DBVarObjCmdTarget Property

Syntax ScaleMin = _Double

Description Questa proprietà consente di leggere o impostare il valore minimo normalizzato della variabile specificata. Corrisponde alla proprietà "Valore min normalizzato" di una variabile.

Parameter	Description
None	None

Result Double

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Min = " & objVar.ScaleMin, vbInformation, GetProjectTitle
    Set objVar = Nothing
End Sub
```

ScaleRawMax, DBVarObjCmdTarget Property

Syntax ScaleRawMax = _Double

Description Questa proprietà consente di leggere o impostare il valore minimo della variabile specificata da normalizzare. Corrisponde alla proprietà "Valore max non normalizzato" di una variabile.

Parameter	Description
None	None

Result Double

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Scale Raw Max = " &
    objVar.ScaleRawMax, vbInformation, GetProjectTitle
    Set objVar = Nothing
End Sub
```

ScaleRawMin, DBVarObjCmdTarget Property

Syntax ScaleRawMin = _Double

Description Questa proprietà consente di leggere o impostare il valore minimo della variabile specificata da normalizzare. Corrisponde alla proprietà "Valore min non normalizzato" di una variabile.

Parameter	Description
None	None

Result Double

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
```



```

        MsgBox "Scale Raw Min = " &
objVar.ScaleRawMin,vbInformation,GetProjectTitle
Set objVar = Nothing
End Sub

```

SharedRetentive, DBVarObjCmdTarget Property

Syntax SharedRetentive = _Boolean

Description Attiva o disattiva la proprietà di Ritentività della variabile. La proprietà è disponibile ad esempio a Sviluppo nel contesto Dropping Code di un Simbolo di Libreria e se utilizzata a Runtime non è permanente: al successivo riavvio del Runtime il valore viene ripristinato con quello impostato a sviluppo.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Public Sub Click()
Dim objVar As DBVarObjCmdTarget
Set objVar = GetVariableObject("VAR00001")
objVar.SharedRetentive = True
MsgBox "Shared Retentive = " & objVar.SharedRetentive, vbInformation,
GetProjectTitle
End Sub

```

SndFileProp DBVarObjCmdTarget Property

Syntax SndFileProp = _String

Description Questa proprietà consente di leggere o impostare la stringa associata alla proprietà "File Suono" di una variabile. Il file sonoro associato dovrà essere di tipo ".wav". Una volta impostata potrà essere interpretata dall'OPC Client connesso se dispone delle funzionalità necessarie.

Parameter	Description
None	None

Result String

Example:

```

Sub Main
Dim objVar As DBVarObjCmdTarget
Set objVar = GetVariableObject("VAR00001")
MsgBox "Sound file Prop = " & objVar.SndFileProp,vbInformation,GetProjectTitle
Set objVar = Nothing
End Sub

```

StatisticData DBVarObjCmdTarget Property

Syntax StatisticData = _Boolean

Description Questa proprietà è di sola lettura e consente di sapere se la statistica dati per la variabile in questione è abilitata o no. Sarà in questo caso possibile avere a disposizione il valore minimo, massimo e medio della variabile ed altre informazioni.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget
    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Enable Statistic Data = " &
        objVar.StatisticData, vbInformation, GetProjectTitle
    Set objVar = Nothing
End Sub
```

StatisticDataAverage, DBVarObjCmdTarget Property

Syntax StatisticDataAverage = _Variant

Description Questa proprietà consente di leggere il valore statistico medio di una variabile. Per poter utilizzare questa proprietà è necessario impostare a TRUE la proprietà StatisticData. La proprietà è di sola lettura.

Parameter	Description
None	None

Result Variant

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget
    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Statistic Data Average Value = " &
        objVar.StatisticDataAverage, vbInformation, GetProjectTitle
    Set objVar = Nothing
End Sub
```

StatisticDataMaxValue, DBVarObjCmdTarget Property

Syntax StatisticDataMaxValue = _Variant

Description Questa proprietà consente di leggere il valore statistico minimo di una variabile. Per poter utilizzare questa proprietà è necessario impostare a TRUE la proprietà StatisticData. La proprietà è di sola lettura.

Parameter	Description
None	None

Result Variant

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget
    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Statistic Data Max Value = " &
    objVar.StatisticDataMaxValue, vbInformation, GetProjectTitle
    Set objVar = Nothing
End Sub
```

StatisticDataMinValue, DBVarObjCmdTarget Property

Syntax StatisticDataMinValue = _Variant

Description Questa proprietà consente di leggere il valore statistico massimo di una variabile. Per poter utilizzare questa proprietà è necessario impostare a TRUE la proprietà StatisticData. La proprietà è di sola lettura.

Parameter	Description
None	None

Result Variant

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget
    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Statistic Data Min Value = " &
    objVar.StatisticDataMinValue, vbInformation, GetProjectTitle
    Set objVar = Nothing
End Sub
```

StatisticDataNumSamples, DBVarObjCmdTarget Property

Syntax StatisticDataNumSamples = _Long

Description Questa proprietà consente di leggere il numero di campionamenti eseguiti su una variabile. Per poter utilizzare questa proprietà è necessario impostare a TRUE la proprietà StatisticData. La proprietà è di sola lettura.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget
    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Statistic Data Num. Samples = " &
    objVar.StatisticDataNumSamples,vbInformation,GetProjectTitle
    Set objVar = Nothing
End Sub
```

StatisticTotalTimeOn, DBVarObjCmdTarget Property

Syntax StatisticTotalTimeOn

Description Questa proprietà restituisce un valore di tipo double che indica il tempo totale espresso in secondi in cui la variabile ha avuto un valore diverso da zero. Questo dato viene salvato nel file di retentività della variabile. La proprietà è di sola lettura.

Parameter	Description
None	None

Result Double

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget
    Set objVar = GetRealTimeDB.GetVariableObject("Alarm01")
    MsgBox " TotalTimeOn = " & CStr(objVar.StatisticTotalTimeOn)
    Set objVar = Nothing
End Sub
```

TraceAddMsgLog, DBVarObjCmdTarget Property

Syntax TraceAddMsgLog = _Boolean

Description Questa proprietà consente di leggere o impostare la funzionalità registrazione nel "SysLog" (Log Storico dei Messaggi di Sistema) di un messaggio ogni volta che per la variabile specificata subisce un cambiamento del suo valore. Corrisponde alla proprietà "Inserimento Msg nel SysLog" di una variabile.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Trace Add Msg Log = " &
    objVar.TraceAddMsgLog,vbInformation,GetProjectTitle
End Sub
```

TraceAddDescCol, DBVarObjCmdTarget Property

Syntax TraceAddDescCol = _Boolean

Description Questa proprietà consente di abilitare o disabilitare la registrazione della Descrizione Tag nella tabella di Trace per la variabile specificata. Corrisponde alla proprietà "Aggiungi Colonna Descrizione Tag" di una variabile.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Trace Add Tag Description= " &
    objVar.TraceAddDescCol,vbInformation,GetProjectTitle
End Sub
```

TraceAddGroupCol, DBVarObjCmdTarget Property

Syntax TraceAddGroupCol = _Boolean

Description Questa proprietà consente di abilitare o disabilitare la registrazione della nome del Gruppo della variabile nella tabella di Trace. Corrisponde alla proprietà "Aggiungi Colonna Gruppo Tag" di una variabile.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Trace Add Tag Group= " &
objVar.TraceAddGroupCol,vbInformation,GetProjectTitle
End Sub
```

TraceComment, DBVarObjCmdTarget Property

Syntax TraceComment = _Boolean

Description Questa proprietà, se impostata al valore booleano True, consente di inserire un commento nel Database di trace della variabile ogni volta che la variabile subisce un cambiamento. Tale commento verrà registrato nel campo "Azione" al posto del testo che Movicon avrebbe scritto di default. Per maggiori informazioni fare riferimento alle Proprietà Opzioni di Tracciatura di una Variabile.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Trace Comment = " &
objVar.TraceComment,vbInformation,GetProjectTitle
End Sub
```

TraceEnable, DBVarObjCmdTarget Property

Syntax TraceEnable = _Boolean

Description Questa proprietà consente di leggere o impostare la funzionalità di tracciatura per la variabile specificata. Corrisponde alla proprietà "Abilitazione Trace" di una variabile.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Trace Enable = " & objVar.TraceEnable,vbInformation,GetProjectTitle
End Sub
```

TraceEnableFromToTime, DBVarObjCmdTarget Property

Syntax TRACEEnableFromToTime = _Boolean

Description Questa proprietà consente di leggere o impostare la funzionalità di fascia oraria giornaliera per la registrazione di tracciatura per la variabile specificata. Corrisponde alla proprietà "Abilita Fascia Oraria Giornaliera" di una variabile. Abilitando questa proprietà le registrazioni di tracciatura della variabile verranno eseguite soltanto all'interno dell'orario specificato dalle proprietà di Ora di Inizio e Fine Fascia Oraria.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "TRACE Enable From To Time = " &
    objVar.TRACEEnableFromToTime,vbInformation,GetProjectTitle
End Sub
```

TraceFromTime, DBVarObjCmdTarget Property

Syntax TraceFromTime = _Date

Description Questa proprietà consente di leggere o impostare l'orario di inizio tracciatura per la variabile specificata. Corrisponde alla proprietà "Ora di Inizio Fascia Oraria" di una variabile. Questa proprietà viene considerata solo se si è abilitata la proprietà "Abilita Fascia Oraria Giornaliera".

Parameter	Description
None	None

Result Date

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Trace From Time = " &
        objVar.TraceFromTime,vbInformation,GetProjectTitle
End Sub
```

TraceMaxAgeDays, DBVarObjCmdTarget Property

Syntax TraceMaxAgeDays = _Long

Description Questa proprietà consente di leggere o impostare il tempo massimo espresso in giorni dopo il quale i dati di tracciatura per la variabile specificata inizieranno a ricircolare. Corrisponde alla proprietà "Tempo Max. di Registrazione" di una variabile. Il "Tempo Max. di Registrazione" viene calcolato sommando le impostazioni delle proprietà "TraceMaxAgeDays", "TraceMaxAgeHours" e "TraceMaxAgeMins".

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Trace Max Age Days = " &
        objVar.TraceMaxAgeDays,vbInformation,GetProjectTitle
End Sub
```


TraceMaxAgeHours, DBVarObjCmdTarget Property

Syntax TraceMaxAgeHours = _Long

Description Questa proprietà consente di leggere o impostare il tempo massimo espresso in ore dopo il quale i dati di tracciatura per la variabile specificata inizieranno a ricircolare. Corrisponde alla proprietà "Tempo Max. di Registrazione" di una variabile. Il "Tempo Max. di Registrazione" viene calcolato sommando le impostazioni delle proprietà "TraceMaxAgeDays", "TraceMaxAgeHours" e "TraceMaxAgeMins".

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Trace Max Age Hours = " &
        objVar.TraceMaxAgeHours,vbInformation,GetProjectTitle
End Sub
```

TraceMaxAgeMins, DBVarObjCmdTarget Property

Syntax TraceMaxAgeMins = _Long

Description Questa proprietà consente di leggere o impostare il tempo massimo espresso in minuti dopo il quale i dati di tracciatura per la variabile specificata inizieranno a ricircolare. Corrisponde alla proprietà "Tempo Max. di Registrazione" di una variabile. Il "Tempo Max. di Registrazione" viene calcolato sommando le impostazioni delle proprietà "TraceMaxAgeDays", "TraceMaxAgeHours" e "TraceMaxAgeMins".

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Trace Max Age Minutes= " &
        objVar.TraceMaxAgeMins,vbInformation,GetProjectTitle
End Sub
```

TraceTableName, DBVarObjCmdTarget Property

Syntax TraceTableName = _String

Description Questa proprietà consente di leggere o impostare il nome della Tabella di tracciatura per la variabile specificata. Corrisponde alla proprietà "Nome Tabella" di una variabile.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Trace Table Name = " &
objVar.TraceTableName,vbInformation,GetProjectTitle
End Sub
```

TraceToTime, DBVarObjCmdTarget Property

Syntax TraceToTime = _Date

Description Questa proprietà consente di leggere o impostare l'orario di fine tracciatura per la variabile specificata. Corrisponde alla proprietà "Ora di Inizio Fascia Oraria" di una variabile. Questa proprietà viene considerata solo se si è abilitata la proprietà "Abilita Fascia Oraria Giornaliera".

Parameter	Description
None	None

Result Date

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "Trace to Time = " &
objVar.TraceToTime,vbInformation,GetProjectTitle
End Sub
```

Value, DBVarObjCmdTarget Property

Syntax Value = _Variant

Description Questa proprietà consente di leggere o impostare il valore di una variabile.

Parameter	Description
None	None

Result Variant

Example:

```
Sub Main
    Dim objVar As DBVarObjCmdTarget

    Set objVar = GetVariableObject("VAR00001")
    MsgBox "VAR00001 = " & objVar.Value,vbInformation,GetProjectTitle
End Sub
```

1.15.2. DisplayEditCmdTarget

Functions

GetComboListInterface, DisplayEditCmdTarget Function

Syntax GetComboListInterface()

Description Questa proprietà restituisce l'oggetto lista di una combobox. I metodi e le proprietà saranno quelle specificate nell'interfaccia ListBoxCmdTarget.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo ListBoxCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()
    If IsCombo Then
        Dim obj As ListBoxCmdTarget
        Set obj = GetObjectInterface.GetComboListInterface
        obj.AddString("prova!!")
    End If
End Sub
```

IsCombo, DisplayEditCmdTarget Function

Syntax IsCombo()

Description Questa proprietà restituisce True se l'oggetto fa parte di una ComboBox di Movicon.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dbug.Print IsCombo
End Sub
```

IsSpin, DisplayEditCmdTarget Function

Syntax IsSpin()

Description Questa proprietà restituisce True se l'oggetto è uno spin di Movicon.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dbug.Print IsSpin
End Sub
```

LoadExtSettings, DisplayEditCmdTarget Function

Syntax LoadExtSettings()

Description Questa funzione permette di caricare la configurazione dell'oggetto dal file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".XML".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim objSymbol As LoadExtSettings, DisplayEditCmdTarget Function
    Set objSymbol =
    GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
    If objSymbol Is Nothing Then Exit Sub
    objSymbol.ExtSettingsFile = "test.xml"
    objSymbol.LoadExtSettings
```

```
        Set objSymbol = Nothing
End Sub
```

RecalcLayout, DisplayEditCmdTarget Function

Syntax RecalcLayout()

Description Questa funzione aggiorna l'aspetto grafico dell'oggetto alla luce delle modifiche effettuate.

Parameter	Description
None	None

Result None

Example:

```
Public Sub Click()
    If IsCombo Then
        GetObjectInterface.RecalcLayout
    End If
End Sub
```

SaveExtSettings, DisplatEditCmdTarget Function

Syntax SaveExtSettings

Description Questa funzione permette di salvare la configurazione dell'oggetto nel file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim objSymbol As DisplayEditCmdTarget
    Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
    If objSymbol Is Nothing Then Exit Sub
    objSymbol.ExtSettingsFile = "test.sxml"
    objSymbol.SaveExtSettings
    Set objSymbol = Nothing
End Sub
```

Properties

EditingPassword, DisplayEditCmdTarget Property

Syntax EditingPassword = _Boolean

Description Questa proprietà abilita o disabilita la visualizzazione in formato password del valore del display. Una volta modificata la proprietà sarà necessario eseguire la funzione "RecalcLayout" per rendere attiva la modifica.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objSyn As SynopticCmdTarget  
    Dim objDipslay As DisplayEditCmdTarget  
  
    Set objSyn = GetSynopticObject  
    If Not objSyn Is Nothing Then  
        Set objDipslay = objSyn.GetSubObject("MyDisplay").GetObjectInterface  
        If Not objDipslay Is Nothing Then  
            objDipslay.EditingPassword = Not objDipslay.EditingPassword  
            objDipslay.RecalcLayout  
            Set objDipslay = Nothing  
        End If  
        Set objSyn = Nothing  
    End If  
End Sub
```

ExtSettingsFile, DisplayEditCmdTarget Property

Syntax ExtSettingsFile = _String

Description Questa proprietà imposta o restituisce il file di configurazione esterna per l'oggetto referenziato. Il file può essere specificato anche in fase di design nella proprietà "File di Configurazione" dell'oggetto. L'estensione prevista per il file è ".XML".

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim objSymbol As DisplayEditCmdTarget  
    Set objSymbol =  
    GetSynopticObject.GetSubObject("TestObject").GetObjectInterface  
    If objSymbol Is Nothing Then Exit Sub  
    objSymbol.ExtSettingsFile = "test.xml"
```

```

objSymbol.SaveExtSettings
Set objSymbol= Nothing
End Sub

```

FormatData, DisplayEditCmdTarget Property

Syntax FormatData = _String

Description Questa proprietà imposta o restituisce un testo per identificare il formato della variabile che si vuole rappresentare. Accetta un valore di tipo String.

I tipi di formattazione disponibili sono rappresentati dalla sintassi del tipo:

x: dove il numero di x identifica il numero di cifre da visualizzare
x.x: dove il numero di x dopo il punto decimale indica il numero di cifre decimali da visualizzare

Il tipo di formattazione con le cifre decimali "x.x" ha senso se si inserisce una variabile in formato "virgola mobile". Nel caso di un numero "intero" infatti le cifre decimali rimangono sempre a zero.

Parameter	Description
None	None

Result String

Example:
Public Sub Click()
 FormatData = InputBox("Insert value")
End Sub

FormatVariable, DisplayEditCmdTarget Property

Syntax FormatVariable = _String

Description Questa proprietà restituisce o imposta il nome della variabile il cui valore verrà utilizzato per determinare il formato di visualizzazione della variabile associata al Display.

Parameter	Description
None	None

Result String

Example:
Public Sub Click()
 Dim objDisplay As DisplayEditCmdTarget
 Set objDisplay =
 GetSynopticObject.GetSubObject("Display1").GetObjectInterface
 MsgBox "Display1 Format Variable = " & objDisplay.**FormatVariable**,
 vbInformation,GetProjectTitle
 Set objDisplay = Nothing
End Sub

HasSpin, DisplayEditCmdTarget Property

Syntax HasSpin = _Boolean

Description Questa proprietà abilita o disabilita la possibilità di associare un pulsante Spin al display, per poter così incrementare o decrementare il valore della variabile senza dover utilizzare una tastiera.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    HasSpin = True  
End Sub
```

InvertSelection, DisplayEditCmdTarget Property

Syntax InvertSelection = _Boolean

Description Tramite questa proprietà è possibile definire se la lista della combo-box si deve aprire verso l'alto o verso il basso. Impostandola al valore "False" la lista verrà aperta verso il basso, impostandola al valore "True" la lista verrà aperta verso l'alto. Una volta modificata la proprietà sarà necessario eseguire la funzione "RecalcLayout" per rendere attiva la modifica.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objSyn As SynopticCmdTarget  
    Dim objDipslay As DisplayEditCmdTarget  
  
    Set objSyn = GetSynopticObject  
    If Not objSyn Is Nothing Then  
        Set objDipslay = objSyn.GetSubObject("MyDisplay").GetObjectInterface  
        If Not objDipslay Is Nothing Then  
            objDipslay.InvertSelection = Not objDipslay.InvertSelection  
            objDipslay.RecalcLayout  
            Set objDipslay = Nothing  
        End If  
        Set objSyn = Nothing  
    End If  
End Sub
```


IsEditable, DisplayEditCmdTarget Property

Syntax IsEditable = _Boolean

Description Impostando a false questa proprietà è possibile rendere il display di sola lettura.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    IsEditable = True  
End Sub
```

IsSpinHoriz, DisplayEditCmdTarget Property

Syntax IsSpinHoriz = _Boolean

Description Questa proprietà permette di definire se il pulsante di Spin dovrà essere verticale oppure orizzontale.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    IsSpinHoriz = True  
End Sub
```

PromptPad, DisplayEditCmdTarget Property

Syntax LoadExtSettings

Description Questa proprietà abilita o disabilita la possibilità di visualizzare il Pad Numerico o Alfanumerico quando l'utente esegue un click del mouse sul display editabile. Il Pad Numerico compare se la variabile è di tipo numerico, altrimenti comparirà il Pad Alfanumerico.

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub Click()
    Dim objDisplay As DisplayEditCmdTarget
    Set objDisplay = objDisplay
    GetSynopticObject.GetSubObject("Display1").GetObjectInterface
    objDisplay.PromptPad = Not objDisplay.PromptPad
    Set objDisplay = Nothing
End Sub

```

ScaleUnit, DisplayEditCmdTarget Property

Syntax ScaleUnit = _String

Description Questa proprietà imposta o restituisce un testo per identificare l'unità di misura della variabile che si vuole rappresentare. Accetta un valore di tipo String.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    ScaleUnit = InputBox("Insert value")
End Sub

```

SpinStep, DisplayEditCmdTarget Property

Syntax SpinStep = _Double

Description Questa proprietà permette di impostare il valore di incremento o decremento che sarà applicato alla variabile tramite i pulsanti di spin.

Parameter	Description
None	None

Result Double

Example:

```

Public Sub Click()
    Dim dRet As Double
    SpinStep = "VAR0001"
    dRet = SpinStep
    MsgBox "SpinStep = " & sRet, vbOkOnly, GetProjectTitle
End Sub

```

TimeToWaitToIncrease, DisplayEditCmdTarget Property

Syntax TimeToWaitToIncrease = _Long

Description Questa proprietà restituisce o imposta il tempo di ritardo all'abilitazione della funzione di Incremento/Decremento veloce. Tenendo infatti premuto con il mouse uno dei due pulsanti spin per il tempo definito, allo scadere del tempo la variabile inizierà a Incrementare/Decrementare in modo rapido.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
    Dim sRet As Long
    TimeToWaitToIncrease = "VAR0001"
    sRet = ValMin
    MsgBox "ValMin = " & sRet, vbOkOnly, GetProjectTitle
End Sub
```

ValMax, DisplayEditCmdTarget Property

Syntax ValMax = _Double

Description Questa proprietà restituisce o imposta il valore massimo che potrà assumere la variabile del Display.



*Il "**Val. Max.**" non verrà considerato se si è specificata una variabile nella proprietà "Var. Max." del gruppo "**Proprietà Variabili dei Display**" per gestire la soglia in modo dinamico.*

Parameter	Description
None	None

Result Double

Example:

```
Public Sub Click()
    Dim sRet As Double
    ValMax = 250
    sRet = ValMax
    MsgBox "ValMax = " & sRet, vbOkOnly, GetProjectTitle
End Sub
```

ValMin, DisplayEditCmdTarget Property

Syntax ValMin = _Double

Description Questa proprietà restituisce o imposta il valore minimo che potrà assumere la variabile del Display.



*Il "**Val. Min.**" non verrà considerato se si è specificata una variabile nella proprietà "Var. Min." del gruppo "**Proprietà Variabili dei Display**" per gestire la soglia in modo dinamico.*

Parameter	Description
None	None

Result Double

Example:

```
Public Sub Click()
    Dim sRet As Double
    ValMin = 0
    sRet = ValMin
    MsgBox "ValMin = " & sRet, vbOkOnly, GetProjectTitle
End Sub
```

ValueToDisplay, DisplayEditCmdTarget Property

Syntax ValueToDisplay = eEditDisplayStatistics

Description Questa proprietà consente di leggere o impostare il tipo di dato da visualizzare per la variabile associata al display. I valori disponibili possono essere selezionati utilizzando l'enumeratore eEditDisplayStatistics o inserendo direttamente il valore numerico:

enum_eds_none: valore attuale (valore 0)
enum_eds_min: valore minimo (valore 1)
enum_eds_max: valore massimo (valore 2)
enum_eds_average: valore medio (valore 3)
enum_eds_TotalTime: tempo totale di ON (valore 4)
enum_eds_min_Day: valore minimo giornaliero (valore 5)
enum_eds_max_Day: valore massimo giornaliero (valore 6)
enum_eds_average_Day: valore medio giornaliero (valore 7)
enum_eds_TotalTime_Day: tempo totale di ON giornaliero (valore 8)
enum_eds_min_Week: valore minimo settimanale (valore 9)
enum_eds_max_Week: valore massimo settimanale (valore 10)
enum_eds_average_Week: valore medio settimanale (valore 11)
enum_eds_TotalTime_Week: tempo totale di ON settimanale (valore 12)
enum_eds_min_Month: valore minimo mensile (valore 13)
enum_eds_max_Month: valore massimo mensile (valore 14)
enum_eds_average_Month: valore medio mensile (valore 15)
enum_eds_TotalTime_Month: tempo totale di ON mensile (valore 16)
enum_eds_min_Year: valore minimo annuale (valore 17)
enum_eds_max_Year: valore massimo annuale (valore 18)
enum_eds_average_Year: valore medio annuale (valore 19)
enum_eds_TotalTime_Year: tempo totale di ON annuale (valore 20)
enum_eds_TimeStamp: tempo ultimo aggiornamento, TimeStamp (valore 21)

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    MsgBox "Value Displayed = " & ValueToDisplay, vbOkOnly, GetProjectTitle
End Sub
```

Variable, DisplayEditCmdTarget Property

Syntax Variable = _String

Description Questa proprietà restituisce o imposta il nome della variabile che verrà visualizzata o impostata tramite il Display.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sRet As String  
    Variable = "VAR0001"  
    sRet = Variable  
    MsgBox "Variable = " & sRet, vbOkOnly, GetProjectTitle  
End Sub
```

VariableMax, DisplayEditCmdTarget Property

Syntax VariableMax = _String

Description Questa proprietà restituisce o imposta il nome della variabile il cui valore verrà utilizzato come valore massimo impostabile sulla "Variabile". In questo modo si rende dinamico il valore della soglia, che quindi potrà essere modificato anche durante la fase di Runtime.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sRet As String  
    VariableMax = "VAR0001"  
    sRet = VariableMax  
    MsgBox "VariableMin = " & sRet, vbOkOnly, GetProjectTitle  
End Sub
```

VariableMin, DisplayEditCmdTarget Property

Syntax VariableMin = _String

Description Questa proprietà restituisce o imposta il nome della variabile il cui valore verrà utilizzato come valore minimo impostabile sulla "Variabile". In questo modo si rende dinamico il valore della soglia, che quindi potrà essere modificato anche durante la fase di Runtime.

Parameter	Description
-----------	-------------

None	None
------	------

Result String

Example:

```
Public Sub Click()
    Dim sRet As String
    VariableMin = "VAR0001"
    sRet = VariableMin
    MsgBox "VariableMin = " & sRet, vbOkOnly, GetProjectTitle
End Sub
```

1.15.3. DLRCmdTarget

Functions

GetADOConn, DLRCmdTarget Function

Syntax GetADOConn()

Description Questa funzione restituisce un oggetto di tipo ADODB Connection per la connessione al Data Logger tramite codice script.



Questa funzione se utilizzata su Windows CE restituisce sempre un'oggetto di tipo ADOCE.connection.3.1. Evitare inoltre la chiusura della connessione ADO usando il metodo "close", altrimenti anche Movicon non sarà più in grado di accedere a quel database.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Dim Conn1 As New ADODB.Connection
    Dim Rs1 As New ADODB.Recordset
    Dim contFields As Integer
    Dim sQuery As String
    Set Conn1 = objDLR.GetADOConn
    sQuery = "SELECT * FROM Log5sec"
    Set Rs1 = CreateObject("ADODB.Recordset")
    Rs1.Open sQuery, Conn1, adOpenForwardOnly, adLockReadOnly,
    ADODB.adCmdText
    ' Loop per stampare tutti i campi del recorset
    While Not Rs1.EOF
        For contFields = 0 To (Rs1.Fields.Count-1)
            Debug.Print Rs1.Fields(contFields).Name & " = " &
            Rs1.Fields(contFields).Value
        Next
        Rs1.MoveNext
    Wend
    Rs1.Close
    Conn1.Close
```

```

End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub

```

GetColumn, DLRCmdTarget Function

Syntax GetColumn(_IpszName)

Description Questa funzione restituisce l'oggetto colonna specificato dal parametro IpszName. I metodi e le proprietà disponibili per l'oggetto restituito sono descritte nell'interfaccia DLRCmdTarget.

Parameter	Description
IpszName As String	Nome della colonna da reperire.

Result Object
Viene restituito un oggetto di tipo DLRCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```

Dim objDLR As DLRCmdTarget
Public Sub Click()
    If Not(objDLR.IsRecipe) Then
        Dim obj As DLRCmdTarget
        Set obj = objDLR.GetColumn("Col00001")
        Debug.Print obj.Variable
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub

```

GetColumnNameList, DLRCmdTarget Function

Syntax GetColumnNameList()

Description Questa funzione restituisce una stringa contenente la lista delle colonne definite per il Data Logger/ricetta. I campi colonna sono separati da virgole e nel caso il nome di una colonna contenga degli spazi verrà racchiuso tra due parentesi quadre.

Parameter	Description
None	None

Result String

Example:

```

Dim objDLR As DLRCmdTarget

Public Sub Click()
    Set objDLR = GetDataLoggerRecipe("Recipe1")
    Debug.Print objDLR.GetColumnNameList 'return string like: [ID Ricetta],
    Column1, Column2, ...
    Set objDLR = Nothing
End Sub

```

End Sub

GetDNSConnectionString, DLRCmdTarget Function

Syntax GetDNSConnectionString()

Description Questa funzione permette di impostare o leggere un eventuale collegamento ODBC personalizzato. Di default Movicon crea un file in formato MS Access nella cartella "LOGS" del progetto nominato:

NomeProgetto_HisLog.mdb

Tramite questa funzione è però possibile personalizzare il collegamento ODBC, creando una base dati differente da MS Access ed eventualmente con un nome differente.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.GetDNSConnectionString
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

GetNextTickLocalTime, DLRCmdTarget Function

Syntax GetNextTickLocalTime()

Description Questa funzione restituisce in formato stringa l'ora e il giorno della prossima registrazione su Data Logger. Il valore restituito indica la data e l'ora della registrazione tenendo come riferimento l'orario locale.

Parameter	Description
None	None

Result Date

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.GetNextTickLocalTime
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```


GetNextTickTime, DLRCmdTarget Function

Syntax GetNextTickTime()

Description Questa funzione restituisce in formato stringa l'ora e il giorno della prossima registrazione su Data Logger. Il valore restituito indica la data e l'ora della registrazione tenendo come riferimento l'orario di Greenwich.

Parameter	Description
None	None

Result Date

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.GetNextTickTime
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

Properties

ActivateVariable, DLRCmdTarget Property

Syntax ActivateVariable = _String

Description Questa proprietà imposta o restituisce il nome della una variabile del Real Time DB di Movicon da utilizzare per eseguire l'attivazione della ricetta selezionata. Tramite questo comando i dati caricati sulle variabili temporanee della ricetta verranno attivati anche sulle variabili di ricetta vere e proprie.
La variabile verrà poi impostata al valore "zero" da Movicon una volta eseguita l'operazione.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        Debug.Print objDLR.ActivateVariable
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

CRWReportFile, DLRCmdTarget Property

Syntax CRWReportFile = _String

Description Questa proprietà imposta o restituisce il nome del file di Report associato al Data Logger/ricetta referenziato.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre una stringa vuota)

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.CRWReportFile
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

DeleteVariable, DLRCmdTarget Property

Syntax DeleteVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile del Real Time DB di Movicon da utilizzare per eseguire la cancellazione della ricetta selezionata. Tramite questo comando la ricetta selezionata verrà eliminata dal Database.
La variabile verrà poi impostata al valore "zero" da Movicon una volta eseguita l'operazione.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        Debug.Print objDLR.DeleteVariable
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

DSN, DLRCmdTarget Property

Syntax DSN = _String

Description Questa impostazione permette di definire o reperire un eventuale collegamento **ODBC** personalizzato per il Data Logger o Ricetta referenziati. Di default Movicon creerà un DSN con lo stesso nome del progetto e suffisso "_DLR", configurato per accedere al database specificato tramite il "PlugIn ODBC di Default" del progetto. Il nome del DSN sarà del tipo:

NomeProgetto_DLR

Tramite questa proprietà è però possibile personalizzare il collegamento **ODBC**, creando una base dati personalizzata anche differente da quella definita nel "PlugIn ODBC di Default" del progetto.



Questa proprietà può essere utilizzata in scrittura solo in fase di design del progetto, ad esempio nel dropping code di un simbolo Template. In fase di Runtime invece, anche se può essere modificata, non verrà applicata al DataLogger/Ricetta, che continuerà a registrare utilizzando il DSN iniziale.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.DSN
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

DurationDays, DLRCmdTarget Property

Syntax DurationDays = _Long

Description Questa proprietà permette di leggere o definire per quanti giorni i dati verranno memorizzati prima di riciclare. Il tempo introdotto dovrà essere ponderato in base alla presunta frequenza di registrazione dei dati per evitare di creare delle tabelle contenenti quantità di dati eccessive. In pratica più la frequenza di registrazione è elevata più il tempo massimo di registrazione dovrebbe essere diminuito.



Il tempo massimo di registrazione va inserito in base alle proprie necessità, ma anche tenendo conto sia della frequenza di registrazione sia del tipo di database che si andrà ad utilizzare. Utilizzando ad esempio un database di tipo MS Access si avranno maggiori limitazioni sulla quantità di dati registrati rispetto ad un database SQL Server.

Parameter	Description
None	None

Result Long

Example:

```

Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.DurationDays
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub

```

DurationHours, DLRCmdTarget Property

Syntax DurationHours = _Long

Description Questa proprietà permette di leggere o definire per quante ore, oltre ai giorni impostati nella proprietà DurationDays, i dati verranno memorizzati prima di riciclare. Il tempo introdotto dovrà essere ponderato in base alla presunta frequenza di registrazione dei dati per evitare di creare delle tabelle contenenti quantità di dati eccessive. In pratica più la frequenza di registrazione è elevata più il tempo massimo di registrazione dovrebbe essere diminuito.



Il tempo massimo di registrazione va inserito in base alle proprie necessità, ma anche tenendo conto sia della frequenza di registrazione sia del tipo di database che si andrà ad utilizzare. Utilizzando ad esempio un database di tipo MS Access si avranno maggiori limitazioni sulla quantità di dati registrati rispetto ad un database SQL Server.

Parameter	Description
None	None

Result Long

Example:

```

Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.DurationHours
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub

```

DurationMinutes, DLRCmdTarget Property

Syntax DurationMinutes = _Long

Description Questa proprietà permette di leggere o definire per quanti minuti, oltre ai giorni impostati nella proprietà DurationDays e alle ore impostate con la proprietà DurationHours, i dati verranno memorizzati prima di riciclare. Il tempo introdotto dovrà essere ponderato in base alla presunta frequenza di registrazione dei dati per evitare di creare delle tabelle contenenti quantità di dati eccessive. In pratica più la frequenza di registrazione è elevata più il tempo massimo di registrazione dovrebbe essere diminuito.



Il tempo massimo di registrazione va inserito in base alle proprie necessità, ma anche tenendo conto sia della frequenza di registrazione sia del tipo di database che si andrà ad utilizzare. Utilizzando ad esempio un database di tipo MS Access si avranno maggiori limitazioni sulla quantità di dati registrati rispetto ad un database SQL Server.

Parameter	Description

None	None
------	------

Result Long

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.DurationMinutes
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

Enabled, DLRCmdTarget Property

Syntax Enabled = _Boolean

Description Questa proprietà imposta o restituisce la proprietà 'Abilitato' del Data Logger/ricetta referenziato.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.Enabled
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

EnableTimeFrom, DLRCmdTarget Property

Syntax EnableTimeFrom = _Date

Description Questa proprietà imposta o restituisce l'orario di inizio registrazioni dati del Data Logger quando è abilitata l'opzione "Abilita Fascia Oraria Giornaliera".

Parameter	Description
None	None

Result Date

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.EnableTimeFrom
End Sub
```

```

Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub

```

EnableTimeFromTo, DLRCmdTarget Property

Syntax EnabledTimeFromTo = _Boolean

Description Questa proprietà imposta o restituisce la proprietà di registrazione all'interno di una fascia oraria giornaliera. Attivando questa proprietà, sarà necessario specificare in che fascia oraria della giornata la registrazione dei dati sarà consentita (salvo il consenso alla registrazione determinato dalla "Variabile Abilita"). Al di fuori degli orari stabiliti, tutti i comandi di registrazione impartiti all'oggetto Data Logger saranno ignorati tranne la registrazione eseguita con la variabile "Registra su Comando" che rimarrà invece attiva. Questa proprietà è presa in considerazione da Movicon solo se la proprietà di registrazione 'a tempo' è abilitata per lo stesso Data Logger; questa proprietà è verificabile tramite codice script con la proprietà "RecordOnTime".



La registrazione dei dati su fasce di orario, soprattutto se queste possono essere variabili, può anche essere eseguita utilizzando opportunamente la "Variabile Abilita".

Parameter	Description
None	None

Result Boolean

Example:

```

Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.EnabledTimeFromTo
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub

```

EnableTimeTo, DLRCmdTarget Property

Syntax EnableTimeTo = _Date

Description Questa proprietà imposta o restituisce l'orario di fine registrazioni dati del Data Logger quando è abilitata l'opzione "Abilita Fascia Oraria Giornaliera".

Parameter	Description
None	None

Result Date

Example:

```

Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.EnableTimeTo
End Sub

```

```

Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub

```

EnableTimeVariable, DLRCmdTarget Property

Syntax EnableTimeVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile del Real Time DB di Movicon da utilizzare per eseguire le registrazioni su comando del Data Logger. Questa selezione ha significato soltanto se è stata abilitata la proprietà "Registra su Comando". La variabile può essere dichiarata di qualsiasi tipo (bit, byte, word, ecc.), poiché Movicon eseguirà la registrazione quando il valore contenuto nella variabile sarà "diverso da zero". A registrazione avvenuta, Movicon provvederà a forzare il valore zero sulla variabile di registrazione.

Parameter	Description
None	None

Result String

Example:

```

Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.EnableTimeVariable
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub

```

ExecuteVariable, DLRCmdTarget Property

Syntax ExecuteVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile del Real Time DB di Movicon da utilizzare per eseguire l'eventuale query impostata per una Ricetta. La query che verrà eseguita dovrà essere contenuta all'interno della variabile "Variabile di Query". La variabile verrà poi impostata al valore "zero" da Movicon una volta eseguita l'operazione.

Parameter	Description
None	None

Result String

Example:

```

Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        Debug.Print objDLR.ExecuteVariable
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub

```

Filter, DLRCmdTarget Property

Syntax Filter = _String

Description Questa proprietà dà la possibilità di eseguire un filtro dei dati di ricetta (clausola WHERE).

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        Debug.Print objDLR.Filter
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

FilterVariable, DLRCmdTarget Property

Syntax FilterVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile del Real Time DB di Movicon da utilizzare per eseguire un filtro dei dati di ricetta (clausola WHERE).

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        Debug.Print objDLR.FilterVariable
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

InsertVariable, DLRCmdTarget Property

Syntax InsertVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile del Real Time DB di Movicon da utilizzare per eseguire il salvataggio della ricetta selezionata. Tramite questo comando i dati caricati sulle variabili temporanee della ricetta verranno salvati sul Database. La variabile verrà poi impostata al valore "zero" da Movicon una volta eseguita l'operazione.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        Debug.Print objDLR.InsertVariable
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

IsRecipe, DLRCmdTarget Property

Syntax IsRecipe = _Boolean

Description Questa proprietà restituisce il valore booleano True se l'oggetto di tipo DLRCmdTarget referenziato è una ricetta. Restituisce il valore False se è un Data Logger.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.IsRecipe
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

LocalTimeColName, DLRCmdTarget Property

Syntax LocalTimeColName = _String

Description Questa proprietà restituisce il nome della Colonna Tempo Locale della tabella del Data Logger. Se non viene specificato nulla verrà utilizzato il nome di default. La Colonna Tempo Locale indica la data e l'ora della registrazione tenendo come riferimento l'orario locale.



La proprietà è di sola lettura.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.LocalTimeColName
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

MaxCacheBeforeFlush, DLRCmdTarget Property

Syntax MaxCacheBeforeFlush = _Long

Description Questa proprietà imposta o restituisce la massima dimensione della Cache prima che il sistema scarichi i dati su file. Il numero impostato è espresso in Byte.

Parameter	Description
None	None

Result Long

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.MaxCacheBeforeFlush
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

MaxError, DLRCmdTarget Property

Syntax MaxError = _Long

Description Questa proprietà imposta o restituisce il numero massimo di errori DBMS dopo i quali la connessione viene considerata non valida e i dati verranno salvati su file in formato ASCII nelle cartelle relative del progetto ("DLOGGERS", "LOGS", "DATA").

Parameter	Description
None	None

Result Long

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.MaxError
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

MaxNumberTrans, DLRCmdTarget Property

Syntax MaxNumberTrans = _Long

Description Questa proprietà imposta o restituisce il numero massimo di transizioni per ciclo da aggiornare prima della loro chiusura.

Parameter	Description
None	None

Result Long

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.MaxNumberTrans
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

MoveFirstVariable, DLRCmdTarget Property

Syntax MoveFirstVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile del Real Time DB di Movicon da utilizzare per eseguire lo spostamento al primo record dell'eventuale RecordSet di dati selezionato.

Il fronte di salita di questa variabile permetterà di rappresentare, nelle variabili associate alle Colonne del database, i valori estratti nel RecordSet.

Ad esempio, filtrando un gruppo di dati dal database mediante l'esecuzione di un comando di Filtro, verrà generato un RecordSet in memoria contenente i dati filtrati. Attivando ad un valore diverso da zero la "Variabile Muovi Primo", nelle variabili associate alle colonne del database sarà scritto il valore estratto nel RecordSet relativo al primo Record.

La variabile verrà poi impostata al valore "zero" da Movicon una volta eseguita l'operazione.

Parameter	Description
None	None

Result String

Example:

```

Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        Debug.Print objDLR.MoveFirstVariable
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub

```

MoveLastVariable, DLRCmdTarget Property

Syntax MoveLastVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile del Real Time DB di Movicon da utilizzare per eseguire lo spostamento all'ultimo record dell'eventuale RecordSet di dati selezionato.
 Il fronte di salita di questa variabile permetterà di rappresentare, nelle variabili associate alle Colonne del database, i valori estratti nel RecordSet.
 Ad esempio, filtrando un gruppo di dati dal database mediante l'esecuzione di un comando di Filtro, verrà generato un RecordSet in memoria contenente i dati filtrati. Attivando ad un valore diverso da zero la "Variabile Muovi Ultimo", nelle variabili associate alle colonne del database sarà scritto il valore estratto nel RecordSet relativo all'ultimo Record.
 La variabile verrà poi impostata al valore "zero" da Movicon una volta eseguita l'operazione.

Parameter	Description
None	None

Result String

Example:

```

Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        Debug.Print objDLR.MoveLastVariable
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub

```

MoveNextVariable, DLRCmdTarget Property

Syntax MoveNextVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile del Real Time DB di Movicon da utilizzare per eseguire lo spostamento al record successivo rispetto a quello corrente dell'eventuale RecordSet di dati selezionato.
 Il fronte di salita di questa variabile permetterà di rappresentare, nelle variabili associate alle Colonne del database, i valori estratti nel RecordSet.
 Ad esempio, filtrando un gruppo di dati dal database mediante l'esecuzione di un comando di Filtro, verrà generato un RecordSet in memoria contenente i dati filtrati. Attivando ad un valore diverso da zero la "Variabile Muovi Prossimo", nelle variabili associate alle colonne del database sarà scritto il valore estratto nel RecordSet relativo al record successivo rispetto a quello corrente.
 La variabile verrà poi impostata al valore "zero" da Movicon una volta eseguita l'operazione.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        Debug.Print objDLR.MoveNextVariable
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

MovePrevVariable, DLRCmdTarget Property

Syntax MovePrevVariable = _String

Description

Questa proprietà imposta o restituisce il nome della variabile del Real Time DB di Movicon da utilizzare per eseguire lo spostamento al record precedente rispetto a quello corrente dell'eventuale RecordSet di dati selezionato.

Il fronte di salita di questa variabile permetterà di rappresentare, nelle variabili associate alle Colonne del database, i valori estratti nel RecordSet.

Ad esempio, filtrando un gruppo di dati dal database mediante l'esecuzione di un comando di Filtro, verrà generato un RecordSet in memoria contenente i dati filtrati. Attivando ad un valore diverso da zero la "Variabile Muovi Precedente", nelle variabili associate alle colonne del database sarà scritto il valore estratto nel RecordSet relativo al record precedente rispetto a quello corrente.

La variabile verrà poi impostata al valore "zero" da Movicon una volta eseguita l'operazione.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        Debug.Print objDLR.MovePrevVariable
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

MSecColName, DLRCmdTarget Property

Syntax MSecColName = _String

Description

Questa proprietà restituisce il nome della Colonna MSec della tabella del Data Logger. Se non viene specificato nulla verrà utilizzato il nome di default. La Colonna MSec indica i millisecondi relativi all'orario di registrazione.



La proprietà è di sola lettura.

Parameter	Description
None	None

Result

String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.MSecColName
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

Name, DLRCmdTarget Property

Syntax

Name = _String

Description

Questa proprietà restituisce il nome dell'oggetto Data Logger/ricetta referenziato.

Parameter	Description
None	None

Result

String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.Name
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

PrintVariable, DLRCmdTarget Property

Syntax

PrintVariable = _String

Description

Questa proprietà imposta o restituisce il nome della variabile del Real Time DB di Movicon che, se posta allo stato logico "diverso da zero" effettuerà la stampa dei dati registrati. La variabile verrà poi impostata al valore "zero" da Movicon una volta eseguita l'operazione. La stampa è comunque sempre riferita all'eventuale Report specificato nella proprietà "File di Report" del Data Logger. Il report è in formato Report Designer (.rep) o Crystal Report (.RPT) ed è associato alla tabella dati del Data Logger.

Per un ulteriore approfondimento sulla gestione dei Reports e della stampa dei reports, consultare la sezione "Reports".



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre una stringa vuota)

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.PrintVariable
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

Query, DLRCmdTarget Property

Syntax Query = _String

Description Questa proprietà imposta o restituisce la query in linguaggio standard SQL sui dati da selezionare dal Database.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        Debug.Print objDLR.Query
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

QueryVariable, DLRCmdTarget Property

Syntax QueryVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile del Real Time DB di Movicon da utilizzare per eseguire una query in linguaggio standard SQL sui dati da selezionare dal Database.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        Debug.Print objDLR.QueryVariable
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

ReadVariable, DLRCmdTarget Property

Syntax ReadVariable= _String

Description Questa proprietà imposta o restituisce il nome della variabile del RealTimeDB di Movicon da utilizzare per eseguire il comando di lettura dal campo dei valori di ricetta e di aggiornare con tali valori le variabili temporanee della ricetta selezionata.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        objDLR.ReadVariable= "VarNameUsedToRead"
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("Recipe")
End Sub
```

ReasonColName, DLRCmdTarget Property

Syntax ReasonColName = _String

Description Questa proprietà restituisce il nome della Colonna Ragione della tabella del Data Logger. Se non viene specificato nulla verrà utilizzato il nome di default. La Colonna Ragione indica da quale evento è stata eseguita la registrazione, (comando, cambiamento o tempo).



La proprietà è di sola lettura.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.ReasonColName
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

RecipeIndexName, DLRCmdTarget Property

Syntax RecipeIndexName = _String

Description Questa proprietà imposta o restituisce il nome del campo indice della ricetta.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        Debug.Print objDLR.RecipeIndexName
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

RecordOnChange, DLRCmdTarget Property

Syntax RecordOnChange = _Boolean

Description Questa proprietà imposta o restituisce la proprietà di registrazione 'a cambiamento' del Data Logger referenziato. Se la proprietà restituisce il valore booleano True significa che il Data Logger registra su cambio di stato delle variabili associate alle colonne. La registrazione su cambio di stato non influisce sulla registrazione a comando o a tempo, in quanto tutte le modalità possono coesistere. Ad ogni cambiamento di valore di una delle variabili (colonne) associate all'oggetto Data Logger, Movicon effettuerà una registrazione, che si aggiungerà a quelle su comando o a tempo qualora queste fossero gestite.



La registrazione sarà effettuata in conformità con le impostazioni di abilitazione alla registrazione "Variabile Abilita". In caso contrario il comando Registra sarà ignorato.

Occorre aggiungere alcune precisazioni sul funzionamento della registrazione a cambiamento nel caso in cui sia stato inserita la "Variabile Abilita" del Data Logger.

Il funzionamento in questo caso è il seguente:

- Movicon registra al cambiamento di una variabile associata ad una colonna del Data Logger se la variabile di abilitazione è diversa da zero
- Movicon registra sul fronte di salita della variabile di abilitazione, se una variabile associata ad una colonna del Data Logger è cambiata di valore rispetto all'ultima registrazione effettuata da Movicon
- Movicon NON registra nulla se il flag di abilitazione è al valore zero
- Movicon NON registra sul fronte di salita del flag di abilitazione se tutte le variabili associate alle colonne del Data Logger sono rimaste invariate rispetto all'ultima registrazione. Questo anche se le variabili sono cambiate durante la disabilitazione del Data Logger, ma poi sono ritornate tutte quante ai valori dell'ultima registrazione

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.RecordOnVariable
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

RecordOnChangeDeadBand, DLRCmdTarget Property

Syntax RecordOnChangeDeadBand = _Double

Description Questa proprietà imposta o restituisce il valore di banda morta per il Data Logger desiderato.

Parameter	Description
None	None

Result Double

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print CStr(objDLR.RecordOnChangeDeadBand)
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

RecordOnChangeDeadBandPercent, DLRCmdTarget Property

Syntax RecordOnChangeDeadBandPercent = _Boolean

Description Questa proprietà abilita o disabilita il controllo della banda morta in percentuale per il Data Logger desiderato.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print CStr(objDLR.RecordOnChangeDeadBandPercent)
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

RecordOnChangeEnableDeadBand, DLRCmdTarget Property

Syntax RecordOnChangeEnableDeadBand = _Boolean

Description Questa proprietà abilita o disabilita la gestione della banda morta per il Data Logger desiderato.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print CStr(objDLR.RecordOnChangeEnableDeadBand)
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

RecordOnlyWhenQualityGood, DLRCmdTarget Property

Syntax RecordOnlyWhenQualityGood = _Boolean

Description Questa proprietà permette di abilitare la registrazione dei dati soltanto quando tutte le variabili associate alle colonne del Data Logger hanno la proprietà "Qualità" Buona. Lasciando l'impostazione a False le registrazioni avverranno indipendentemente dalla qualità delle variabili.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.RecordOnlyWhenQualityGood
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

RecordOnTime, DLRCmdTarget Property

Syntax RecordOnTime = _Boolean

Description Questa proprietà imposta o restituisce la proprietà di registrazione 'a tempo' del Data Logger referenziato.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.RecordOnTime
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

RecordOnVariable, DLRCmdTarget Property

Syntax RecordOnVariable = _Boolean

Description Questa proprietà imposta o restituisce la proprietà di registrazione 'a comando' del Data Logger referenziato. Se la proprietà restituisce il valore booleano True significa che il Data Logger registra solo quando la variabile associata passa dal valore "zero" ad un valore "maggiore di zero". La variabile in questione deve appartenere al Real Time DB di Movicon e può essere dichiarata di qualsiasi tipo (bit, byte, word, ecc.), poiché Movicon eseguirà la registrazione quando il valore contenuto nella variabile sarà diverso da zero. A registrazione avvenuta, Movicon provvederà a forzare il valore zero sulla variabile di registrazione.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.RecordOnVariable
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

RecVariable, DLRCmdTarget Property

Syntax RecVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile associata per la registrazione su comando.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.RecVariable
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

RecycleDBConnection, DLRCmdTarget Property

Syntax RecycleDBConnection = _Boolean

Description Questa proprietà, se abilitata, permette di mantenere aperta la connessione **DBMS**, e quindi di utilizzare la stessa connessione per tutte le transizioni che verranno eseguite. Se invece la proprietà viene disabilitata, la connessione **DBMS** verrà aperta quando viene richiesta una transizione e poi verrà chiusa.



Si consiglia di disabilitare la proprietà "Mantieni aperta la Connessione DB" solo nel caso in cui le registrazioni sono poco frequenti.

Parameter	Description
-----------	-------------

None	None
------	------

Result Boolean

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.RecycleDBConnection
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

ResetVariable, DLRCmdTarget Property

Syntax ResetVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile del Real Time DB di Movicon che, se posta allo stato logico "diverso da zero" effettuerà la cancellazione di tutti i valori registrati nella tabella fino a quel momento, per poi ricominciare un nuovo set di registrazioni. La variabile verrà poi impostata al valore "zero" da Movicon una volta eseguita l'operazione. Il successivo comando di registrazione comporterà la scrittura dei dati sul primo record di una tabella vuota.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.ResetVariable
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

Sort, DLRCmdTarget Property

Syntax Sort = _String

Description Questa proprietà imposta o restituisce il nome della una variabile del Real Time DB di Movicon da utilizzare per eseguire l'ordinamento dei dati di ricetta (clausola ORDER BY).

Parameter	Description
None	None

Result String

Example:

```

Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        Debug.Print objDLR.Sort
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub

```

SortVariable, DLRCmdTarget Property

Syntax SortVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile del Real Time DB di Movicon da utilizzare per eseguire l'ordinamento dei dati di ricetta (clausola ORDER BY).

Parameter	Description
None	None

Result String

Example:

```

Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        Debug.Print objDLR.SortVariable
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub

```

StatusVariable, DLRCmdTarget Property

Syntax StatusVariable = _String

Description Questa proprietà imposta o restituisce il nome della una variabile del Real Time DB di Movicon nella quale potrà essere riportato lo stato di esecuzione dell'eventuale query. Nel momento in cui viene eseguita una query, Movicon permette di informare le logiche sullo stato esecutivo della query impostata, notificando lo stato mediante la gestione dei seguenti bit sulla variabile impostata:

Bit 0 = query in Esecuzione
 Bit 1 = BOF
 Bit 2 = EOF
 Bit 3 = Record Cancellato
 Bit 4 = Errore

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If objDLR.IsRecipe Then
        Debug.Print objDLR.StatusVariable
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

TableName, DLRCmdTarget Property

Syntax TableName = _String

Description Questa proprietà restituisce il nome della tabella associata al Data Logger/ricetta referenziato.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.TableName
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

TimeColName, DLRCmdTarget Property

Syntax TimeColName = _String

Description Questa proprietà restituisce il nome della Colonna Tempo della tabella del Data Logger. Se non viene specificato nulla verrà utilizzato il nome di default. La Colonna Tempo indica la data e l'ora della registrazione tenendo come riferimento l'orario di Greenwich.



La proprietà è di sola lettura.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.TimeColName
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

TimeRecHour, DLRCmdTarget Property

Syntax TimeRecHour = _Byte

Description Questa proprietà imposta o restituisce il tempo di campionamento espresso in ore del Data Logger. Il valore ha significato soltanto se è abilitata la proprietà "Registra a Tempo".

Parameter	Description
None	None

Result Byte

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.TimeRecHour
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

TimeRecMin, DLRCmdTarget Property

Syntax TimeRecMin = _Byte

Description Questa proprietà imposta o restituisce il tempo di campionamento espresso in minuti del Data Logger. Il valore ha significato soltanto se è abilitata la proprietà "Registra a Tempo".

Parameter	Description
None	None

Result Byte

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.TimeRecMin
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

TimeRecMSec, DLRCmdTarget Property

Syntax TimeRecMSec = _Integer

Description Questa proprietà imposta o restituisce il tempo di campionamento espresso in millisecondi del Data Logger. Il valore ha significato soltanto se è abilitata la proprietà "Registra a Tempo".

Parameter	Description
None	None

Result Integer

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.TimeRecMSec
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

TimeRecSec, DLRCmdTarget Property

Syntax TimeRecSec = _Byte

Description Questa proprietà imposta o restituisce il tempo di campionamento espresso in secondi del Data Logger. Il valore ha significato soltanto se è abilitata la proprietà "Registra a Tempo".

Parameter	Description
None	None

Result Byte

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.TimeRecSec
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

UserColName, DLRCmdTarget Property

Syntax UserColName = _String

Description Questa proprietà restituisce il nome della Colonna Utente della tabella del Data Logger. Se non viene specificato nulla verrà utilizzato il nome di default. La Colonna Utente indica il nome dell'utente attivo al momento della registrazione.



La proprietà è di sola lettura.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.UserColName
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

UserName, DLRCmdTarget Property

Syntax UserName = _String

Description Questa proprietà imposta o restituisce il nome dell'utente utilizzato per la connessione **ODBC**.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.UserName
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

VarCharsMax, DLRCmdTarget Property

Syntax VarCharsMax = _Long

Description Questa proprietà imposta o restituisce la massima precisione per le colonne di tipo stringa. Il numero impostato rappresenta il numero di caratteri della stringa.

Parameter	Description
None	None

Result Long

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Debug.Print objDLR.VarCharsMax
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

1.15.4. DLRColumnCmdTarget

Properties

AddNumUpdatesCol, DLRColumnCmdTarget Property

Syntax AddNumUpdatesCol = _Boolean

Description Questa proprietà abilita o disabilita la proprietà 'Aggiungi Colonna Num. Aggiornamenti'; se abilitata, permette di aggiungere nella tabella del Database una colonna riferita alla variabile che riporta il numero di variazioni che ha subito la variabile nell'intervallo di tempo che è intercorso tra una registrazione e la successiva.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If Not(objDLR.IsRecipe) Then
        Dim obj As DLRColumnCmdTarget
        Set obj = objDLR.GetColumn("Col00001")
        Debug.Print obj.AddNumUpdatesCol
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

AddQualityColumn, DLRColumnCmdTarget Property

Syntax AddQualityColumn = _Boolean

Description Questa proprietà abilita o disabilita la proprietà 'Aggiungi Colonna Qualità'; se abilitata, permette di aggiungere nella tabella del Database una colonna riferita alla

variabile che riporta lo stato di Qualità della variabile stessa nel momento in cui verrà eseguita la registrazione.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If Not(objDLR.IsRecipe) Then
        Dim obj As DLRColumnCmdTarget
        Set obj = objDLR.GetColumn("Col00001")
        Debug.Print obj.AddQualityColumn
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

Name, DLRColumnCmdTarget Property

Syntax Name = _String

Description Questa proprietà imposta o restituisce il nome della colonna associata al Data Logger/ricetta referenziato.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Dim obj As DLRColumnCmdTarget
    Set obj = objDLR.GetColumn("Col00001")
    Debug.Print obj.Name
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

NumUpdatesColumnName, DLRColumnCmdTarget Property

Syntax NumUpdatesColumnName = _String

Description Questa proprietà permette di definire un nome personalizzato della colonna Num. Aggiornamenti quando la proprietà "Aggiungi Colonna Num. Aggiornamenti" è

abilitata o la proprietà AddNumUpdatesCol è impostata a True. Se il campo viene lasciato vuoto verrà utilizzato il nome di default (NomeColonna_NumUpdates).

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If Not(objDLR.IsRecipe) Then
        Dim obj As DLRColumnCmdTarget
        Set obj = objDLR.GetColumn("Col00001")
        Debug.Print obj.NumUpdatesColumnName
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

QualityColumnName, DLRColumnCmdTarget Property

Syntax QualityColumnName = _String

Description Questa proprietà permette di definire un nome personalizzato della colonna Qualità quando la proprietà "Aggiungi Colonna Qualità" è abilitata o la proprietà AddQualityColumn è impostata a True. Se il campo viene lasciato vuoto verrà utilizzato il nome di default (NomeColonna_Quality).

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If Not(objDLR.IsRecipe) Then
        Dim obj As DLRColumnCmdTarget
        Set obj = objDLR.GetColumn("Col00001")
        Debug.Print obj.QualityColumnName
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

RecipeIndex, DLRCmdTarget Property

Syntax RecipeIndex = _Boolean

Description Tramite questa proprietà è possibile definire se la colonna in questione dovrà essere l'indice di ricetta, ovvero l'identificativo della ricetta. Per ogni ricetta sarà possibile definire un solo indice di ricetta.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If (objDLR.IsRecipe) Then
        Dim obj As DLRCmdTarget
        Set obj = objDLR.GetColumn("Col00001")
        Debug.Print obj.RecipeIndex
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

RecipeTempVariable, DLRCmdTarget Property

Syntax RecipeTempVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile del Real Time DB di Movicon che sarà utilizzata come variabile di appoggio temporaneo del dato della ricetta. Soltanto quando verrà eseguito il comando di attivazione della ricetta il valore contenuto nella "Variabile Temporanea" verrà copiato nella "Variabile" di output della ricetta.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If (objDLR.IsRecipe) Then
        Dim obj As DLRCmdTarget
        Set obj = objDLR.GetColumn("Col00001")
        Debug.Print obj.RecipeTempVariable
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

RecordType, DLRColumnCmdTarget Property

Syntax RecordType = _String

Description Questa proprietà permette di definire quale valore della variabile dovrà essere registrato sul database.
Le possibilità sono le seguenti:

- **Istantaneo:** verrà registrato il valore istantaneo della variabile, ovvero il valore contenuto nella variabile nell'istante della registrazione
- **Minimo:** verrà registrato il valore minimo assunto della variabile nell'intervallo di tempo che intercorre tra una registrazione e la successiva
- **Massimo:** verrà registrato il valore massimo assunto della variabile nell'intervallo di tempo che intercorre tra una registrazione e la successiva
- **Medio:** verrà registrato il valore medio assunto della variabile nell'intervallo di tempo che intercorre tra una registrazione e la successiva

I valori possibili sono

Istantaneo - 0
Minimo - 1
Massimo - 2
Medio - 3

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If Not(objDLR.IsRecipe) Then
        Dim obj As DLRColumnCmdTarget
        Set obj = objDLR.GetColumn("Col00001")
        Debug.Print obj.RecordType
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

StatisticAverageValue, DLRColumnCmdTarget Property

Syntax StatisticAverageValue = _Variant

Description Questa proprietà restituisce il valore statistico medio della variabile associata alla colonna.

Parameter	Description
None	None

Result Variant

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If Not(objDLR.IsRecipe) Then
        Dim obj As DLRColumnCmdTarget
        Set obj = objDLR.GetColumn("Col00001")
        Debug.Print obj.StatisticAverageValue
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

StatisticMaxValue, DLRColumnCmdTarget Property

Syntax StatisticMaxValue = _Variant

Description Questa proprietà restituisce il valore statistico massimo della variabile associata alla colonna tra una registrazione e la successiva

Parameter	Description
None	None

Result Variant

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If Not(objDLR.IsRecipe) Then
        Dim obj As DLRColumnCmdTarget
        Set obj = objDLR.GetColumn("Col00001")
        Debug.Print obj.StatisticMaxValue
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

StatisticMinValue, DLRColumnCmdTarget Property

Syntax StatisticMinValue = _Variant

Description Questa proprietà restituisce il valore statistico minimo della variabile associata alla colonna.

Parameter	Description
None	None

Result Variant

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If Not(objDLR.IsRecipe) Then
        Dim obj As DLRColumnCmdTarget
        Set obj = objDLR.GetColumn("Col00001")
        Debug.Print obj.StatisticMinValue
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

StatisticNumUpdates, DLRColumnCmdTarget Property

Syntax StatisticNumUpdates = _Variant

Description Questa proprietà restituisce il numero di aggiornamenti della variabile associata alla colonna.

Parameter	Description
None	None

Result Variant

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    If Not(objDLR.IsRecipe) Then
        Dim obj As DLRColumnCmdTarget
        Set obj = objDLR.GetColumn("Col00001")
        Debug.Print obj.StatisticNumUpdates
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

Variable, DLRColumnCmdTarget Property

Syntax Variable = _String

Description Questa proprietà, che è di sola lettura, restituisce il nome della variabile associata alla colonna del Data Logger/ricetta referenziato.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRCmdTarget
Public Sub Click()
    Dim obj As DLRColumnCmdTarget
    Set obj = objDLR.GetColumn("Col00001")
    Debug.Print obj.Variable ' i.e.: return string "VAR00001"
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetDataLoggerRecipe("DLR1")
End Sub
```

1.15.5. DLRWndCmdTarget

Events

OnFilter, DLRWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di applicare un filtro per l'estrazione dei dati dal Data Logger.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

OnPrint, DLRWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di effettuare la stampa dei dati caricati nella finestra di visualizzazione.



Questo evento non è supportato in Windows CE.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

OnRefresh, DLRWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di effettuare un rinfresco dei dati caricati nella finestra di visualizzazione.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

Functions

EditCopy, DLRWndCmdTarget Function

Syntax EditCopy()

Description Questa proprietà esegue una copia negli appunti del contenuto della riga selezionata.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.EditCopy
End Sub
Public Sub SymbolLoading()
    Set objDLR =
        GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

EditLayout, DLRWndCmdTarget Function

Syntax EditLayout()

Description Questa funzione apre la finestra di configurazione dei campi da visualizzare nella finestra di visualizzazione del Data Logger.



La funzione verrà eseguita soltanto se nell'oggetto Finestra è stata abilitata la proprietà "Mostra Finestra di Controllo". In caso contrario la "Finestra Scelta Campo" non verrà aperta e la funzione ritornerà il valore "False"

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.EditLayout
End Sub
Public Sub SymbolLoading()
    Set objDLR =
        GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

LoadExtSettings, DLRWndCmdTarget Function

Syntax LoadExtSettings

Description Questa funzione permette di caricare la configurazione dell'oggetto dal file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".XML".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
Dim objSymbol As DLRWndCmdTarget  
Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface  
If objSymbol Is Nothing Then Exit Sub  
objSymbol.ExtSettingsFile = "test.sxml"  
objSymbol.LoadExtSettings  
Set objSymbol = Nothing  
End Sub
```

RecalcLayout, DLRWndCmdTarget Function

Syntax RecalcLayout()

Description Questa funzione esegue un'aggiornamento grafico dell'oggetto. E' necessario eseguire tale funzione dopo che e' stata modificata una proprietà il cui impatto grafico sia particolarmente consistente.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRWndCmdTarget  
Public Sub Click()  
    objDLR.RecalcLayout  
End Sub  
Public Sub SymbolLoading()  
    Set objDLR = GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface  
End Sub
```

Refresh, DLRWndCmdTarget Function

Syntax Refresh()

Description Questa funzione esegue un'aggiornamento dei dati visualizzati nell'oggetto. E' necessario eseguire tale funzione dopo che e' stata modificata la query di interrogazione sul database.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.Refresh
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

SaveExtSettings, DLRWndCmdTarget Function

Syntax SaveExtSettings

Description Questa funzione permette di salvare la configurazione dell'oggetto nel file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
Dim objSymbol As DLRWndCmdTarget
Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
If objSymbol Is Nothing Then Exit Sub
objSymbol.ExtSettingsFile = "test.sxml"
objSymbol.SaveExtSettings
Set objSymbol = Nothing
End Sub
```

Properties

AutoLayout, DLRWndCmdTarget Property

Syntax AutoLayout = _Boolean

Description Abilitando questa proprietà, il layout della lista verrà impostato in modo automatico. Questo significa che le colonne della tabella verranno automaticamente

ridimensionate in modo che risultino visibili tutte quante nell'area della Finestra Data Logger/Ricette. Disabilitando invece questa proprietà, all'apertura della finestra le colonne avranno la dimensione definita in programmazione, con la possibilità che le ultime sulla destra non siano visibili se non attivando la barra di scorrimento orizzontale.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.AutoLayout
End Sub
Public Sub SymbolLoading()
    Set objDLR =
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

ButtonPos, DLRWndCmdTarget Property

Syntax ButtonPos = _Integer

Description Questa impostazione restituisce la posizione in cui dovranno apparire i pulsanti nella finestra di visualizzazione dei dati.

Le possibilità sono:
 0 = sinistra
 1 = sopra
 2 = destra
 3 = sotto

Parameter	Description
None	None

Result Integer

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    If Not objDLR Is Nothing Then
        MsgBox "objDLR 's ButtonPos is " &
        objDLR.ButtonPos,vbInformation,GetProjectTitle
        objDLR.ButtonPos = 2
        objDLR.RecalcLayout
    Else
        MsgBox "objDLRWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR =
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

ButtonSize, DLRWndCmdTarget Property

Syntax ButtonSize = _Integer

Description Questa impostazione restituisce la dimensione dei pulsanti che verranno poi visualizzati nella finestra di visualizzazione dei dati.

Le possibilità sono:

0 = piccolo

1 = medio

2 = grande

Parameter	Description
None	None

Result Integer

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    If Not objDLR Is Nothing Then
        MsgBox "objDLR 's ButtonSize is " &
            objDLR.ButtonSize,vbInformation,GetProjectTitle
        objDLR.ButtonSize = 2
        objDLR.RecalcLayout
    Else
        MsgBox "objDLRWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR =
        GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

Clickable, DLRWndCmdTarget Property

Syntax Clickable = _Boolean

Description Questa proprietà permette di definire se l'operatore potrà interagire con la finestra di Visualizzazione. Impostando a False questa proprietà il controllo non potrà infatti essere gestito nè con il mouse nè con la tastiera. In questo caso quindi non sarà possibile, ad esempio, ordinare le colonne, vedere l'eventuale help, eseguire eventuali comandi presenti nella finestra.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.Project
End Sub
Public Sub SymbolLoading()
```



```

Set objDLR = GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub

```

DLR, DLRWndCmdTarget Property

Syntax DLR = _String

Description Questa proprietà imposta o restituisce il nome del Data Logger associato alla finestra di visualizzazione.

Parameter	Description
None	None

Result String

Example:

```

Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.DLR
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub

```

ExtSettingsFile, DLRWndCmdTarget Property

Syntax ExtSettingsFile = _String

Description Questa proprietà imposta o restituisce il file di configurazione esterna per l'oggetto referenziato. Il file può essere specificato anche in fase di design nella proprietà "File di Configurazione" dell'oggetto. L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
Dim objSymbol As DLRWndCmdTarget
Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
If objSymbol Is Nothing Then Exit Sub
objSymbol.ExtSettingsFile = "test.sxml"
objSymbol.SaveExtSettings
Set objSymbol= Nothing
End Sub

```

FilterBtnText, DLRWndCmdTarget Property

Syntax FilterBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando di stampa dei dati nella finestra di visualizzazione del data logger. Se non specificato nulla, Movicon utilizza il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    If Not objDLR Is Nothing Then
        MsgBox "objDLR 's FilterBtnText is " & objDLR.FilterBtnText
        ,vbInformation,GetProjectTitle
        objDLR.FilterBtnText = "Filter data"
        objDLR.Refresh
    Else
        MsgBox "objDLRWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR =
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

FilterFromDate, DLRWndCmdTarget Property

Syntax FilterFromDate = _Date

Description Questa proprietà imposta o restituisce il filtro 'Data iniziale' per la visualizzazione dei messaggi nella finestra di Trace di Movicon.

Parameter	Description
None	None

Result Date

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    If Not objDLR Is Nothing Then
        MsgBox "objDLR 's FilterFromDate is " &
        objDLR.FilterFromDate,vbInformation,GetProjectTitle
        objDLR.FilterFromDate = Now
        objDLR.Refresh
    Else
        MsgBox "objDLRWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub
Public Sub SymbolLoading()
```

```

Set objDLR = GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub

```

FilterToDate, DLRWndCmdTarget Property

Syntax FilterToDate = _Date

Description Questa proprietà imposta o restituisce il filtro 'Data finale' per la visualizzazione dei messaggi nella finestra di Trace di Movicon.

Parameter	Description
None	None

Result Date

Example:

```

Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    If Not objDLR Is Nothing Then
        MsgBox "objDLR 's FilterToDate is " & objDLR.FilterToDate
        ,vbInformation,GetProjectTitle
        objDLR.FilterToDate = Now
        objDLR.Refresh
    Else
        MsgBox "objDLRWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub

```

FilterUser, DLRWndCmdTarget Property

Syntax FilterUser = _String

Description Questa proprietà imposta o restituisce il filtro 'Utente' per la visualizzazione dei messaggi nella finestra di Trace di Movicon.

Parameter	Description
None	None

Result String

Example:

```

Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    If Not objDLR Is Nothing Then
        MsgBox "objDLR 's FilterUser is " &
        objDLR.FilterUser,vbInformation,GetProjectTitle
        objDLR.FilterUser = "User00001"
        objDLR.Refresh
    End If
End Sub

```

```

Else
    MsgBox "objDLRWnd is nothing",vbInformation,GetProjectTitle
End If
End Sub
Public Sub SymbolLoading()
    Set objDLR =
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub

```

GraphicButtons, DLRWndCmdTarget Property

Syntax GraphicButtons = _Boolean

Description Abilitando questa proprietà i pulsanti della Finestra DataLogger-Ricette verranno disegnati usando un'icona invece del testo. Il testo opportuno verrà invece visualizzato come tooltip posizionando il mouse sopra ad un pulsante.



Il tooltip non viene gestito nella versione Windows CE.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```

Sub Click()
    GraphicButtons = True
    RecalcLayout
End Sub

```

IncludeMilliseconds, DLRWndCmdTarget Property

Syntax IncludeMilliseconds = _Boolean

Description Impostando a True questa proprietà, verranno inclusi anche i millisecondi al formato "Ora" delle colonne della finestra che supportano questo tipo di dato.

Parameter	Description
None	None

Result Boolean

Example:

```

Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.IncludeMilliseconds

```

```

End Sub
Public Sub SymbolLoading()
    Set objDLR = GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub

```

MaxCount, DLRWndCmdTarget Property

Syntax MaxCount = _Long

Description Questa proprietà imposta o restituisce il numero massimo di righe che la Finestra Data Logger/Ricette potrà visualizzare.

Parameter	Description
None	None

Result Long

Example:

```

Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.MaxCount
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub

```

NetworkBackupServerName, DLRWndCmdTarget Property

Syntax NetworkBackupServerName = _String

Description Questa proprietà imposta o restituisce il nome dell'eventuale Server di Networking di Backup usato per prelevare i dati da visualizzare nella Finestra DataLogger/Ricette quando il Server Primario, ovvero quello impostato nella proprietà "NetowrkServerName", è in timeout.



Per visualizzare i dati di un Server è necessario che il Data Logger/Ricetta sia presente anche sul progetto Client, al fine di poter recuperare la struttura del Database. Tuttavia sul progetto Client il Data Logger/Ricetta può essere creato soltanto come struttura, senza quindi dover associare alcuna variabile alle colonne.

Parameter	Description
None	None

Result String

Example:

```

Dim objDLRWnd As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLRWnd.NetworkBackupServerName
End Sub
Public Sub SymbolLoading()
    Set                                     objDLRWnd                                     =
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub

```

NetworkServerName, DLRWndCmdTarget Property

Syntax NetworkServerName = _String

Description Questa proprietà restituisce il nome dell'eventuale Server di Networking dal quale dovranno essere prelevati i dati da visualizzare nella Finestra DataLogger/Ricette.



Per visualizzare i dati di un Server è necessario che il Data Logger/Ricetta sia presente anche sul progetto Client, al fine di poter recuperare la struttura del Database. Tuttavia sul progetto Client il Data Logger/Ricetta può essere creato soltanto come struttura, senza quindi dover associare alcuna variabile alle colonne.

Parameter	Description
None	None

Result String

Example:

```

Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.NetworkServerName
End Sub
Public Sub SymbolLoading()
    Set                                     objDLR                                     =
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub

```

PrintBtnText, DLRWndCmdTarget Property

Syntax PrintBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando di stampa dei dati nella finestra di visualizzazione del data logger. Se non specificato nulla, Movicon utilizza il testo di default.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre una stringa vuota)

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    If Not objDLR Is Nothing Then
        MsgBox "objDLR 's PrintBtnText is " & objDLR.PrintBtnText
        ,vbInformation,GetProjectTitle
        objDLR.PrintBtnText = "Print data"
        objDLR.Refresh
    Else
        MsgBox "objDLRWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub
Public Sub SymbolLoading()
    Set                                objDLR
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

Project, DLRWndCmdTarget Property

Syntax Project = _String

Description Questa proprietà imposta o restituisce il nome del progetto figlio dal quale si vogliono recuperare i dati da visualizzare. Se il campo viene lasciato vuoto verrà utilizzato il progetto corrente.



In questa proprietà va inserito esclusivamente il nome di un eventuale progetto figlio del progetto corrente.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.Project
End Sub
Public Sub SymbolLoading()
    Set                                objDLR
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

Query, DLRWndCmdTarget Property

Syntax Query = _String

Description Questa proprietà permette di impostare una query in linguaggio standard SQL personalizzata per l'estrazione o la richiesta di dati nel database.

Parameter	Description
-----------	-------------

None	None
------	------

Result String

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    objDLR.Query = "SELECT * FROM TabellaDLR WHERE CoIVAR0001 >= 5"
    objDLR.Refresh
End Sub
Public Sub SymbolLoading()
    Set objDLR =
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

RefreshBtnText, DLRWndCmdTarget Property

Syntax RefreshBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando di rinfresco dei dati nella finestra di visualizzazione del data logger. Se non specificato nulla, Movicon utilizza il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    If Not objDLR Is Nothing Then
        MsgBox "objDLR 's RefreshBtnText is " &
        objDLR.RefreshBtnText,vbInformation,GetProjectTitle
        objDLR.RefreshBtnText = "Refresh data"
        objDLR.Refresh
    Else
        MsgBox "objDLRWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR =
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

ShowFilterBtn, DLRWndCmdTarget Property

Syntax ShowFilterBtn = _Boolean

Description Questa proprietà permette di visualizzare il pulsante di comando per l'operazione di filtro dei dati nella finestra di visualizzazione Data Logger.

Parameter	Description
-----------	-------------

None	None
------	------

Result Boolean

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    If Not objDLR Is Nothing Then
        MsgBox "objDLR 's ShowFilterBtn is " & objDLR.ShowFilterBtn
        ,vbInformation,GetProjectTitle
        objDLR.ShowFilterBtn = Not objDLR.ShowFilterBtn
        objDLR.RecalcLayout
    Else
        MsgBox "objDLRWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub
Public Sub SymbolLoading()
    Set                                objDLR                                =
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

ShowPrintBtn, DLRWndCmdTarget Property

Syntax ShowPrintBtn = _Boolean

Description Questa proprietà permette di visualizzare il pulsante di comando per l'operazione di stampa dei dati nella finestra di visualizzazione Data Logger.
La stampa viene eseguita utilizzando il file di report che dovrà essere specificato nella proprietà "File di Report". Movicon "passa" al report le stesse impostazioni di filtro per la stampa dei dati, che coincidono con ciò che è visualizzato nella finestra di Visualizzazione in questione.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre false)

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    If Not objDLR Is Nothing Then
        MsgBox "objDLR 's ShowPrintBtn is " & objDLR.ShowPrintBtn
        ,vbInformation,GetProjectTitle
        objDLR.ShowPrintBtn = Not objDLR.ShowPrintBtn
        objDLR.RecalcLayout
    Else
        MsgBox "objDLRWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub
Public Sub SymbolLoading()
    Set                                objDLR                                =
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

ShowRefreshBtn, DLRWndCmdTarget Property

Syntax ShowRefreshBtn = _Boolean

Description Questa proprietà permette di visualizzare il pulsante di comando per l'operazione di rinfresco dei dati nella finestra di visualizzazione Data Logger.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    If Not objDLR Is Nothing Then
        MsgBox "objDLR 's ShowRefreshBtn is " & objDLR.ShowRefreshBtn
        ,vbInformation,GetProjectTitle
        objDLR.ShowRefreshBtn = Not objDLR.ShowRefreshBtn
        objDLR.RecalcLayout
    Else
        MsgBox "objDLRWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub
Public Sub SymbolLoading()
    Set objDLR =
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

SortBy, DLRWndCmdTarget Property

Syntax SortBy = _String

Description Questa proprietà imposta o restituisce il filtro 'Ordina Per' per la visualizzazione dei messaggi nella finestra di Trace di Movicon.

I campi possibili sono:

Col
LocalCol
MSecCol
ReasonCol
TimeCol
UserCol

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
```

```

        If Not objDLR Is Nothing Then
            MsgBox "objDLR 's SortBy is " & objDLR.SortBy
            ,vbInformation,GetProjectTitle
            objDLR.SortBy = "Col"
            objDLR.Refresh
        Else
            MsgBox "objDLRWnd is nothing",vbInformation,GetProjectTitle
        End If
    End Sub
Public Sub SymbolLoading()
    Set objDLR =
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub

```

SubItemReason, DLRWndCmdTarget Property

Syntax SubItemReason = _String

Description Questa proprietà permette di definire il testo che dovrà apparire come nome della colonna "Ragione". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```

Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.SubItemReason
End Sub
Public Sub SymbolLoading()
    Set objDLR =
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub

```

SubItemReasonPos, DLRWndCmdTarget Property

Syntax SubItemReasonPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Motivo" all'interno della Finestra DataLogger-Ricette. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```

Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.SubItemReasonPos
End Sub
Public Sub SymbolLoading()
    Set                                     objDLR
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub

```

SubItemReasonWidth, DLRWndCmdTarget Property

Syntax SubItemReasonWidth= _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione del Data Logger. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```

Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.SubItemReasonWidth
End Sub
Public Sub SymbolLoading()
    Set                                     objDLR
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub

```

SubItemTime, DLRWndCmdTarget Property

Syntax SubItemTime = _String

Description Questa proprietà permette di definire il testo che dovrà apparire come nome della colonna "Tempo Evento". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```

Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.SubItemTime
End Sub
Public Sub SymbolLoading()

```

```

        Set objDLR = GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
    End Sub

```

SubItemTimePos, DLRWndCmdTarget Property

Syntax SubItemTimePos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Tempo Evento" all'interno della Finestra DataLogger-Ricette. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```

Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.SubItemTimePos
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub

```

SubItemTimeWidth, DLRWndCmdTarget Property

Syntax SubItemTimeWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione del Data Logger. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```

Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.SubItemTimeWidth
End Sub
Public Sub SymbolLoading()
    Set objDLR = GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub

```

End Sub

SubItemUser, DLRWndCmdTarget Property

Syntax SubItemUser = _String

Description Questa proprietà permette di definire il testo che dovrà apparire come nome della colonna "Utente". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.SubItemUser
End Sub
Public Sub SymbolLoading()
    Set objDLR =
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

SubItemUserPos, DLRWndCmdTarget Property

Syntax SubItemUserPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Utente" all'interno della Finestra DataLogger-Ricette. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.SubItemUserPos
End Sub
Public Sub SymbolLoading()
    Set objDLR =
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

SubItemUserWidth, DLRWndCmdTarget Property

Syntax SubItemUserWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione del Data Logger. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Dim objDLR As DLRWndCmdTarget
Public Sub Click()
    Debug.Print objDLR.SubItemUserWidth
End Sub
Public Sub SymbolLoading()
    Set objDLR =
    GetSynopticObject.GetSubObject("DLRWindow").GetObjectInterface
End Sub
```

1.15.6. DrawCmdTarget

Events

Click, Generic Event

Description Evento notificato quando viene premuto il tasto sinistro o destro del mouse all'interno dell'area del disegno.

Parameter	Description
Nessuno	Nessuno

DbClick, Generic Event

Description Evento generato quando viene premuto per due volte in rapida successione il tasto sinistro o destro del mouse all'interno dell'area del disegno. I tempi del doppio click sono stabiliti nelle impostazioni del sistema operativo.

Parameter	Description
Nessuno	Nessuno

KeyDown, Generic Event

Description Evento notificato quando viene premuto un tasto della tastiera. L'evento restituisce le variabili, di tipo integer, KeyCode e Shift. L'evento viene generato indipendentemente dal focus.

Parameter	Description
KeyCode As Integer	Codice VBA del tasto premuto. Il codice VBA e' un insieme di costanti che, oltre ai normali caratteri alfanumeri, senza pero' distinzione di maiuscolo o minuscolo, contempla anche gli altri tasti della tastiera come, ad esemio i tasti funzione, il Caps Lock, ecc.
Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT

KeyPress, Generic Event

Description Evento notificato quando viene premuto e rilasciato un tasto della tastiera. L'evento restituisce la variabile di tipo integer KeyAscii contenente il codice ASCII del tasto premuto. L'evento viene generato solo se il focus è fissato sul disegno.

Parameter	Description
Keyascii As Integ	Codice ASCII del tasto premuto

KeyUp, Generic Event

Description Eevento generato quando viene rilasciato un tasto (dopo essere stato premuto) della tastiera. L'evento restituisce le variabil,i di tipo integer, KeyCode e Shift. L'evento viene generato indipendentemente dal focus.

Parameter	Description
KeyCode As Integer	Codice VBA del tasto premuto. Il codice VBA e' un insieme di costanti che, oltre ai normali caratteri alfanumeri, senza pero' distinzione di maiuscolo o minuscolo, contempla anche gli altri tasti della tastiera come, ad esemio i tasti funzione, il Caps Lock, ecc.
Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT

KillFocus, Generic Event

Description Eevento generato quando l'oggetto in questione viene deselezionato oppure perde il focus.

Parameter	Description
None	None

MouseDown, Generic Event

Description Evento notificato sia nel codice del sinottico sia nel codice degli oggetti quando viene premuto un qualsiasi pulsante del mouse nel sinottico, indipendentemente dalla posizione o dal simbolo. L'evento restituisce le variabili, di tipo integer, Button e Shift e le variabili, di tipo single, X e Y.
Per poter gestire questo evento soltanto all'interno di un oggetto del sinottico è necessario utilizzare la funzione "IsCursorOnObject".

Parameter	Description
Button As Integer	Indice del tasto mouse premuto: 1 = Sinistro 2 = Destro 4 = Centrale
Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT
X As Single	Coordinata orizzontale riferita alla posizione del cursore al momento dell'evento
Y As Single	Coordinata verticale riferita alla posizione del cursore al momento dell'evento

MouseMove, Generic Event

Description Evento notificato sia nel codice del sinottico sia nel codice degli oggetti quando il cursore del mouse cambia la propria posizione nel sinottico, indipendentemente dalla posizione o dal simbolo. L'evento restituisce le variabili, di tipo integer, Button e Shift e le variabili, di tipo single, X e Y.
Per poter gestire questo evento soltanto all'interno di un oggetto del sinottico è necessario utilizzare la funzione "IsCursorOnObject".

Parameter	Description
Button As Integer	Indice del tasto mouse premuto: 1 = Sinistro 2 = Destro 4 = Centrale
Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT
X As Single	Coordinata orizzontale riferita alla posizione del cursore al momento dell'evento
Y As Single	Coordinata verticale riferita alla posizione del cursore al momento dell'evento

MouseUp, Generic Event

Description Evento notificato sia nel codice del sinottico sia nel codice degli oggetti quando viene rilasciato un qualsiasi pulsante del mouse nel sinottico, indipendentemente dalla posizione o dal simbolo. L'evento restituisce le variabili, di tipo integer, Button e Shift e le variabili, di tipo single, X e Y.
Per poter gestire questo evento soltanto all'interno di un oggetto del sinottico è necessario utilizzare la funzione "IsCursorOnObject".

Parameter	Description
-----------	-------------

Button As Integer	Indice del tasto mouse premuto: 1 = Sinistro 2 = Destro 4 = Centrale
Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT
X As Single	Coordinata orizzontale riferita alla posizione del cursore al momento dell'evento
Y As Single	Coordinata verticale riferita alla posizione del cursore al momento dell'evento

OnChange, Generic Event

Description Evento notificato quando l'oggetto disegno modifica il proprio stato grafico. L'evento restituisce un indice relativo a quale evento grafico è cambiato.

Gli indici sono riportati di seguito:
enum_ONCHANGE_COMPOSEDMOVE
enum_ONCHANGE_SCALE
enum_ONCHANGE_MOVEX
enum_ONCHANGE_MOVEY
enum_ONCHANGE_TITLE
enum_ONCHANGE_STARTINGX
enum_ONCHANGE_STARTINGY
enum_ONCHANGE_ENDINGX
enum_ONCHANGE_ENDINGY
enum_ONCHANGE_FILLING
enum_ONCHANGE_ROTATION
enum_ONCHANGE_VISIBLE
enum_ONCHANGE_EDGECOLOR
enum_ONCHANGE_BACKCOLOR
enum_ONCHANGE_FILLCOLOR
enum_ONCHANGE_BITMAP
enum_ONCHANGE_XROTATIONCENTER
enum_ONCHANGE_YROTATIONCENTER

Parameter	Description
ChangeType As Integer	Indice dell'evento grafico.

OnChangeExecutionCanceled, Generic Event

Description Evento notificato per un oggetto disegno che cerca di ottenere o di spostare l'esecuzione attuale della sinapsi.

Parameter	Description
None	None

OnChangeExecutionToPromoter, Generic Event

Description Evento notificato per l'oggetto disegno che ha in corso l'esecuzione della sinapsi alla quale viene notificato che un altro oggetto sta tentando di cambiare il flusso di esecuzione. Impostando il parametro bRet a False si impedisce che questo avvenga.

Parameter	Description
None	None

OnExecutionPending, Generic Event

Description Evento notificato una volta al secondo dall'oggetto disegno che ha in corso l'esecuzione della sinapsi indicando che il sistema è in attesa che questo passi l'esecuzione ad un altro oggetto tramite le funzioni SynapseValue, SynapseValueFromID o SynapsePassExecution.

Parameter	Description
None	None

OnFireExecution, Generic Event

Description Evento notificato ogni qualvolta la sinapsi dell'oggetto disegno sta per essere eseguita. Il sistema stabilisce a quale disegno attivare la sinapsi in base al flusso logico rappresentato dai collegamenti e all'ordine di tabulazione dato a questi collegamenti. Nel momento in cui all'interno del disegno, che ha in corso l'esecuzione, vengono impostate le sinapsi di uscita, il sistema provvede, in base a quanto detto prima, a passare l'esecuzione della macro ad un altro disegno scatenando l'evento OnFireExecution. Quindi in tale evento vanno inserite le funzioni logiche riguardanti la gestione delle sinapsi di ingresso e uscita del simbolo.

Parameter	Description
None	None

OnFireSynapse, Generic Event

Description Questo evento viene generato ogni qualvolta una sinapsi di input riceve il valore da una sinapsi di output ad essa collegata, o meglio un oggetto disegno ha settato tramite le proprietà SynapseValue, SynapseValueFromID o SynapsePassExecution il valore della propria sinapsi di output alla quale è collegata la sinapsi di input che riceve l'evento.
Tramite il parametro SynapseName è possibile sapere quale sinapsi di input è interessata dall'evento appena descritto.

Parameter	Description
SynapseName As String	Nome della sinapsi che ha appena ricevuto il valore.

OnPostPaint, Generic Event

Description Evento notificato ogni volta che l'oggetto disegno ha subito un 'refresh' grafico sul sinottico da parte del sistema. Questo può avvenire in diverse circostanze, ad esempio a seguito dell'apertura di un sinottico, quando l'applicazione riprende il focus, e comunque tutte le volte che viene eseguita un'animazione o ricalcolata la posizione sullo schermo del disegno.
Il parametro hdc (Handle to the device context) dà informazioni utili all'utente esperto di Windows sul tipo di rinfresco grafico adottato da Movicon.

Parameter	Description
ByVal hdc As Long	Handle to the device context.

OnPrePaint, Generic Event

Description Evento notificato ogni volta che l'oggetto disegno sta per subire un 'refresh' grafico sul sinottico da parte del sistema. Questo può avvenire in diverse circostanze, ad esempio a seguito dell'apertura di un sinottico, quando l'applicazione riprende il focus, e comunque tutte le volte che viene eseguita un'animazione o ricalcolata la posizione sullo schermo del disegno.

Impostando a false il parametro bRet viene disabilitato il refresh grafico del simbolo. Il parametro hdc (Handle to the device context) dà informazioni utili all'utente esperto di Windows sul tipo di rinfresco grafico adottato da Movicon.

Parameter	Description
ByVal hdc As Long	Handle to the device context
bRet As Boolean	Abilitazione al ridisegno

OnTextChanged, Generic Event

Description Evento notificato quando il testo del titolo dell'oggetto è stato modificato da tastiera. Restituisce la variabile stringa ChangedText contenente il nuovo testo.



L'evento "OnTextChanged" in un oggetto richiede automaticamente il login di un utente che rispetti l'impostazione di livello di password richiesto (impostato sull'oggetto stesso), nel caso in cui la gestione password del progetto sia abilitata.

Parameter	Description
ChangedText As String	Nuovo testo contenuto nel campo Titolo.

OnTextChanging, Generic Event

Description Evento notificato quando il titolo dell'oggetto viene modificato da tastiera. La variabile booleana bRet consente o meno tale modifica: se la variabile bRet viene posta a False all'interno della funzione, il cambiamento del testo contenuto nell'oggetto non ha effetto.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento del testo.

OnTimer, Generic Event

Description Evento notificato con un periodo di 1/2 di secondo circa (tempo non garantito) durante la fase di run-time. Durante il Test il periodo è proporzionale alla velocità di test impostata. Il tempo di esecuzione dell'evento può comunque essere personalizzato tramite la chiave di registro TimerEventFrequency.

Parameter	Description
None	None

OnToolTip, Generic Event

Description Evento notificato quando l'oggetto è nelle condizioni di visualizzare una stringa "pop-up" denominata ToolTip (ad esempio è puntato dal mouse). Restituisce la variabile booleana Show che è True se il mouse si trova sull'oggetto e False se il mouse è fuori dall'oggetto. Per visualizzare il ToolTip deve essere impostato il relativo metodo.

Parameter	Description
Show As Boolean	Variabile che indica se il cursore mouse è sull'oggetto.

SetFocus, Generic Event

Description Evento notificato quando l'oggetto disegno riceve il focus o viene selezionato.

Parameter	Description
None	None

SymbolLoading, Generic Event

Description Evento notificato quando l'oggetto disegno viene caricato in memoria, quindi all'apertura del sinottico di appartenenza. Questo evento è indipendente dalle condizioni di visibilità del disegno stesso.

Parameter	Description
None	None

SymbolUnloading, Generic Event

Description Evento notificato quando l'oggetto disegno viene scaricato dalla memoria alla chiusura del sinottico. Questo evento è indipendente dalle condizioni di visibilità del disegno stesso.

Parameter	Description
None	None

Functions

AddPolyPoint, DrawCmdTarget Function

Syntax AddPolyPoint(_nPos, _nX, _nY)

Description Questa funzione si utilizza nei disegni poligono per aggiungere un nuovo vertice. Il punto viene posto nel segmento indicato con nPos tenendo presente che dal primo punto che è stato disegnato al secondo il segmento assume il numero 1, dal secondo al terzo 2 e così via fino all'ultimo (dall'ultimo punto al primo) che prende valore 0. Il nuovo vertice avrà le coordinate cartesiane espresse in pixel nX e nY. L'aggiunta di un vertice comporta una nuova numerazione dei segmenti. Supponiamo di avere un poligono con 10 vertici (10 segmenti da 0 a 9) aggiungendo un punto nella posizione 7 il nuovo segmento sarà il numero 8, quello che prima era il segmento 8 diverrà il 9 e infine il 9 sarà il 10. Se la posizione non esiste il punto non viene aggiunto e la funzione ritorna con il valore False.

Parameter	Description
nPos As Integer	Posizione del segmento al cui centro inserire un nuovo vertice del poligono.
nX As Integer	Coordinata X in pixel dall'origine del sinottico in cui posizionare il nuovo punto.
nY As Integer	Coordinata Y in pixel dall'origine del sinottico in cui posizionare il nuovo punto.

Result Boolean

Example:

```
Public Sub Click()  
    MsgBox(CStr(AddPolyPoint(0, 10, 10)), vbOkOnly, GetProjectTitle)  
End Sub
```

CloseThisSynoptic, DrawCmdTarget Function

Syntax CloseThisSynoptic()

Description Chiude il sinottico contenente l'oggetto in cui la funzione viene richiamata. La funzione non ha alcun effetto se richiamata nel sinottico di avvio. Tipicamente viene utilizzata per chiudere finestre (sinottici) aperte in maniera modale o in un frame separato.

Parameter	Description
-----------	-------------

None	None
------	------

Result Long

Example:

```
Public Sub Click()
    CloseThisSynoptic
End Sub
```

ConvertAngleToPoint, DrawCmdTarget Function

Syntax ConvertAngleToPoint(_nXOffset, _nYOffset, _nAngle, _nRadius, _pnX, _pnY)

Description Questa funzione permette di identificare le coordinate X, Y del punto di origine di un angolo. I parametri passati con la funzione identificano l'angolo (nAngle), la lunghezza (nRadius) e gli offset rispetto ai punti iniziali e finali del vettore (nXOffset e nYOffset).
La funzione scrive sui parametri pnX e pnY le coordinate calcolate.

Parameter	Description
nXOffset	Offset in X.
nYOffset	Offset in Y.
nAngle	Angolo del vettore in gradi.
nRadius	Lunghezza del vettore.
pnX	Coordinata X calcolata.
pnY	Coordinata Y calcolata.

Result None

Example:

```
Public Sub Click()
    Dim nXOffset As Integer
    Dim nYOffset As Integer
    Dim nAngle As Double
    Dim nRadius As Double
    Dim pnX As Integer
    Dim pnY As Integer

    ' Vector offset (Offset dei vettori)
    nXOffset = -10
    nYOffset = 10
    ' Angle and lenght (Angolo e lunghezza)
    nAngle = 0
    nRadius = 100
    ' Convert to point (Converte nel punto)
    ConvertAngleToPoint(nXOffset, nYOffset, nAngle, nRadius, pnX, pnY)
    ' View X,Y (Visualizza X,Y)
    MsgBox "pnX=" & pnX & " pnY=" & pnY, vbOkOnly, "Test CenterRotation"
    'Return-> 90,10
End Sub
```

ConvertPointToAngle, DrawCmdTarget Function

Syntax ConvertPointToAngle(_nXOffset, _nYOffset, _nX, _nY)

Description Questa funzione permette di identificare l'angolo di un vettore definito dalle coordinate di origine nX, nY e dagli offset rispetto all'origine nXOffset, nYOffset. La funzione restituisce il valore dell'angolo del vettore rispetto all'orizzontale da 0 a 360 gradi in senso antiorario.

Parameter	Description
nXOffset	Offset in X.
nYOffset	Offset in Y.
nX	Coordinata X del punto di origine.
nY	Coordinata Y del punto di origine.

Result Long

Example:

```
Public Sub Click()  
    Dim nXOffset As Integer  
    Dim nYOffset As Integer  
    Dim nX As Integer  
    Dim nY As Integer  
    Dim nResult As Long  
  
    ' Vector offset (Offset vettore)  
    nXOffset = -10  
    nYOffset = 10  
    ' Vector origin (Origine vettore)  
    nX = 0  
    nY = 0  
    ' Calculate (Calcola)  
    nResult = ConvertPointToAngle(nXOffset, nYOffset, nX, nY)  
    MsgBox("Angle=" & nResult, , "ConvertPointToAngle")  
End Sub
```

CursorPosToObjectPos, DrawCmdTarget Function

Syntax CursorPosToObjectPos(_pnX, _pnY)

Description Questa funzione imposta e converte la posizione del cursore di puntamento nella posizione dell'oggetto. Accetta due parametri di tipo Integer per le posizioni X e Y. Questa funzione converte la posizione del cursore in funzione di un disegno facente parte di un simbolo restituendo le coordinate originali indipendentemente dalla eventuale modifica delle dimensioni del simbolo.

Parameter	Description
-----------	-------------

nX As Integer	Coordinata X.
nY As Integer	Coordinata Y.

Result None

Example:

'Se l'oggetto che richiama questo codice non è contenuto in un simbolo
'le coordinate restituite sono identiche

```
Public Sub Click()
    Dim nX As Integer
    Dim nY As Integer
    GetCursorPos(nX, nY)
    MsgBox("X=" & nX & " Y=" & nY,vbOkOnly,GetProjectTitle)
    CursorPosToObjectPos(nX, nY)
    MsgBox("X=" & nX & " Y=" & nY,vbOkOnly,GetProjectTitle)
End Sub
```

DeletePolyPoint, DrawCmdTarget Function

Syntax DeletePolyPoint(_nPos)

Description Questa funzione si utilizza nei disegni poligono per cancellare un vertice. Con nPos si indica la posizione considerando che il primo disegnato ha valore 0. Questa operazione comporta una nuova numerazione dei segmenti e dei vertici. Supponiamo di avere un poligono con 10 vertici (da 0 a 9) togliendo un punto nella posizione 7 quello che prima era il vertice 8 diverrà il 7 e il 9 sarà il punto 8. Se la posizione non esiste non viene tolto alcun punto e la funzione ritorna con il valore False.

Parameter	Description
nPos As Integer	Posizione del segmento che si vuole eliminare.

Result Boolean

Example:

```
Public Sub Click()
    MsgBox(CStr(DeletePolyPoint(0)),vbOkOnly,GetProjectTitle)
End Sub
```

EnableVariableEvent, DrawCmdTarget Function

Syntax EnableVariableEvent(_IpszVariableName, _bEnable)

Description Questa funzione abilita o disabilita la notifica dell'evento vincolato al variare di una specifica variabile (vedi On..[VARIABLE].Changed). Accetta i parametri IpszVariableName di tipo stringa contenete il nome della variabile di cui si gestisce l'evento e bEnable di tipo booleano contenente l'abilitazione alla notifica.

Parameter	Description
-----------	-------------

IpszVariableName As String	Nome variabile.
bEnable As Boolean	Valore di abilitazione notifica evento.

Result Boolean

Example:

```
'Ambito Oggetto 1
Public Sub Click()
    EnableVariableEvent("VAR00001",False)
End Sub
```

GetActiveXObject, DrawCmdTarget Function

Syntax GetActiveXObject()

Description Questa funzione restituisce un parametro di tipo Oggetto corrispondente ad un eventuale oggetto ActiveX inserito in un sinottico di Movicon. Serve per accedere alle proprietà dell'oggetto questione. L'accesso avviene tramite la definizione di un oggetto che viene impostato mediante la funzione suddetta.



Questa funzione non è supportata in Windows CE.(se utilizzata restituisce sempre 'null')

Parameter	Description
None	None

Result Object

Example:

```
'Ambito Sinottico
Public Sub Click()
    Dim obj As Object
    Dim app As Object
    Set obj =GetAbsoluteSubObject("Oggetto")
    Set app = obj .GetActiveXObject
    ' le proprietà dipendono dal tipo di oggetto
    app.Locked = True 'blocca la modifica di dati dentro la combo box Oggetto
    Set app =Nothing
    Set obj =Nothing
    ocx.Locked = True 'blocca la modifica di dati dentro la combo boxEnd Sub
'Ambito Oggetto
Public Sub Click()
    Dim app As Object
    Set app = GetActiveXObject()
    ' le proprietà dipendono dal tipo di oggetto
    app.Locked = True 'blocca la modifica di dati dentro la combo box
    Set app = Nothing
End Sub
```

GetAlias, DrawCmdTarget Function

Syntax GetAlias(_IpszAlias, _bRecursive)

Description Questa funzione restituisce il valore definito per l'Alias passato come parametro "IpszAlias". Il parametro bRecursive consente di cercare l'Alias soltanto nella tabella locale dell'oggetto oppure anche nel simbolo contenitore e nel sinottico.

Parameter	Description
IpszAlias As String	Nome dell'Alias per il quale recuperare il valore.
bRecursive As Boolean	Se impostato a True questo parametro consente di ricercare l'Alias anche nell'eventuale simbolo contenitore e nel sinottico.

Result String

Example:

```
Public Sub Click()  
    MsgBox "Alias <<TsetAlais>> = " & GetAlias("TsetAlais", False),vbInformation,  
    GetProjectTitle  
End Sub
```

GetAliasListName, DrawCmdTarget Function

Syntax GetAliasListName()

Description Questa funzione restituisce la lista degli Alias definiti nell'oggetto. Viene restituita una stringa dove i nomi degli Alias sono separati dal carattere "|" (pipe).

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    MsgBox "Alias List = " & GetAliasListName(),vbInformation, GetProjectTitle  
End Sub
```

GetAliasListValue, DrawCmdTarget Function

Syntax GetAliasListValue()

Description Questa funzione restituisce la lista dei valori associati agli Alias definiti nell'oggetto. Viene restituita una stringa dove i valori degli Alias sono separati dal carattere "|" (pipe).

Parameter	Description
-----------	-------------

None	None
------	------

Result String

Example:

```
Public Sub Click()  
    MsgBox "Alias Value List = " & GetAliasListValue(),vbInformation,  
    GetProjectTitle  
End Sub
```

GetCommandsInterfaceOnRelease, DrawCmdTarget Function

Syntax GetCommandsInterfaceOnRelease ()

Description Questa funzione consente di caricare l'interfaccia CommandsListCmdTarget relativa alla lista comandi dell'oggetto. Tramite questa interfaccia è possibile modificare la lista "Comandi su Click" dell'oggetto referenziato.

Parameter	Description
None	None

Result Object: restituisce un oggetto di tipo CommandsListCmdTarget.

Example:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim objCommandList As CommandsListCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    Set objCommandList = objRect.GetCommandsInterfaceOnRelease  
  
    Set objCommandList = Nothing  
    Set objRect = Nothing  
End Sub
```

GetConnectorObjectConnected, DrawCmdTarget Function

Syntax GetConnectorObjectConnected(_lpszSynapseName, _nConnection)

Description Questa funzione permette di accedere alle proprietà e metodi dell'oggetto connettore applicato alla sinapsi. La sinapsi sul disegno è identificata attraverso il nome con il parametro lpszSynapseName mentre il numero della connessione è stabilito dal parametro nConnection, dato che ad una sinapsi possono essere collegati più di un oggetto connettore.

Parameter	Description
-----------	-------------

lpszSynapseName As String	Nome della sinapsi definita nell'oggetto.
nConnection As Integer	Numero della connessione da considerare.

Result Object

Example:

```
Public Sub OnFireExecution()
    Set edge color of symbol connected
    ' (Imposta il colore dei simboli collegati)
    Dim obj As Object

    For i = 0 To GetNumConnectionsOnSynapse("OUT") - 1
        Set obj = GetConnectorObjectConnected("OUT", i)
        obj.EdgeColor = IColor
        Set obj = Nothing
    Next i
End Sub
```

GetContainerObject, DrawCmdTarget Function

Syntax GetContainerObject()

Description Questa funzione restituisce un parametro di tipo Oggetto corrispondente all'eventuale simbolo contenitore. Serve per accedere alle proprietà del simbolo contenente il disegno in questione. L'accesso avviene tramite la definizione di un oggetto che viene impostato mediante la funzione suddetta.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo DrawCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()
    Dim app As DrawCmdTarget

    Set app = GetContainerObject()
    If app Is Nothing Then
        MsgBox("app is Nothing", vbOkOnly, GetProjectTitle)
    Else
        MsgBox(app.Title, vbOkOnly, GetProjectTitle)
    End If
    Set app = Nothing
End Sub
```

GetCursorPos, DrawCmdTarget Function

Syntax GetCursorPos(_pnX, _pnY)

Description Questa funzione restituisce la posizione attuale del cursore attraverso la restituzione di due parametri di tipo Integer. Le coordinate sono espresse in pixels e riferite all'angolo della finestra sinottico superiore sinistro.

Parameter	Description
nX As Integer	Coordinata X.
nY As Integer	Coordinata Y.

Result Nome

Example:

```
Public Sub Click()  
    Dim nX As Integer  
    Dim nY As Integer  
    GetCursorPos (nX,nY)  
    MsgBox("X=" & nX & " Y=" & nY,vbOkOnly,GetProjectTitle)  
End Sub
```

GetCursorPosInObject, DrawCmdTarget Function

Syntax GetCursorPosInObject(_pnX, _pnY)

Description Questa funzione restituisce la posizione attuale del cursore attraverso la restituzione di due parametri di tipo Integer. Le coordinate sono espresse in pixels e riferite all'angolo superiore sinistro dell'oggetto.

Parameter	Description
nX As Integer	Coordinata X.
nY As Integer	Coordinata Y.

Result None

Example:

```
Public Sub Click()  
    Dim nX As Integer  
    Dim nY As Integer  
    GetCursorPosInObject(nX,nY)  
    MsgBox("X=" & nX & " Y=" & nY,vbOkOnly,GetProjectTitle)  
End Sub
```

GetGaugeObject, DrawCmdTarget Function

Syntax GetGaugeObject()

Description Questa funzione permette di accedere alle proprietà e metodi specifici di un oggetto gauge vettoriale inserito nel sinottico. Le proprietà e metodi menzionati sono quelli descritti nell'interfaccia GaugeCmdTarget.

L'utilizzo di questa funzione, a differenza della `GetSubGaugeObject`, non rende necessario inserire il Gauge vettoriale in un simbolo.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo `GaugeCmdTarget` se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà `Nothing`.

Example:

```
Public Sub Click()  
    Dim ObjGauge As GaugeCmdTarget  
  
    Set ObjGauge = GetSynopticObject.GetSubObject("Gauge").GetGaugeObject  
    If ObjGauge Is Nothing Then MsgBox "ObjGauge Is Nothing", vbExclamation +  
        vbOkOnly, "GetGaugeObject"  
    Set ObjGauge = Nothing  
End Sub
```

GetNumConnectionsOnSynapse, DrawCmdTarget Function

Syntax `GetNumConnectionsOnSynapse(_IpszSynapseName)`

Description Questa funzione permette di stabilire il numero di oggetti connettore collegati alla sinapsi, identificandone il nome attraverso il parametro `IpszSynapseName`.

Parameter	Description
<code>IpszSynapseName As String</code>	Nome della sinapsi definita nell'oggetto.

Result Integer

Example:

```
Public Sub OnFireExecution()  
    ' Set edge color of connector  
    ' (Imposta il colore dei collegamenti)  
    Dim obj As Object  
  
    For i = 0 To GetNumConnectionsOnSynapse("OUT") - 1  
        Set obj = GetConnectorObjectConnected("OUT", i)  
        obj.EdgeColor = IColor  
        Set obj = Nothing  
    Next i  
End Sub
```

GetNumPolyPoint, DrawCmdTarget Function

Syntax `GetNumPolyPoint()`

Description Questa funzione si utilizza nei disegni poligono per reperire il numero di segmenti (vertici) di cui si compone il poligono stesso.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    MsgBox("Numero di segmenti: " &
        GetNumPolyPoints, vbOkOnly, GetProjectTitle)
End Sub
```

GetNumSynapsis, DrawCmdTarget Function

Syntax GetNumSynapsis()

Description Questa funzione permette di verificare il numero di sinapsi che contiene l'oggetto disegno.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    MsgBox("GetNumSynapsis =" & GetNumSynapsis, vbOkOnly, GetProjectTitle)
End Sub
```

GetObjectConnectedOnSynapse, DrawCmdTarget Function

Syntax GetObjectConnectedOnSynapse(_IpszSynapseName, _nConnection)

Description Questa funzione permette di accedere alle proprietà e metodi di un oggetto disegno connesso alla sinapsi attraverso l'oggetto connettore. La sinapsi sul disegno è identificata attraverso il nome con il parametro IpszSynapseName mentre il numero della connessione è stabilito dal parametro nConnection, dato che ad una sinapsi possono essere collegati più di un oggetto connettore.

Parameter	Description
IpszSynapseName As String	Nome della sinapsi definita nell'oggetto.
nConnection As Integer	Numero della connessione da considerare.

Result Object
Viene restituito un oggetto di tipo DrawCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:


```

Public Sub OnFireExecution()
    ' Set edge color of symbol connected
    ' (Imposta il colore dei simboli collegati)
    Dim obj As DrawCmdTarget

    For i = 0 To GetNumConnectionsOnSynapse("OUT") - 1
        Set obj = GetObjectConnectedOnSynapse("OUT", i)
        obj.EdgeColor = IColor
        Set obj = Nothing
    Next i
End Sub

```

GetObjectInterface, DrawCmdTarget Function

Syntax GetObjectInterface()

Description Questa funzione permette di accedere alle proprietà e metodi specifici di alcuni oggetti inseriti nel sinottico. Le proprietà e metodi menzionati sono quelli descritti nelle relative interfacce. Gli oggetti di cui è possibile accedere alle proprietà sono:

Grafico	ChartWndCmdTarget
Trend	TrendCmdTarget
Gauge	GaugeCmdTarget
AlarmWindow	AlarmWndCmdTarget
Data LoggerWindow	DLRWndCmdTarget
HistoryLog Window	HisLogWndCmdTarget

Parameter	Description
None	None

Result Object

Example:

'Oggetto all'interno di in un sinottico

```

Public Sub Click()
    Dim objTrend As TrendCmdTarget
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
    Debug.Print objTrend.Samples
    Set objTrend = Nothing
End Sub

```

End Sub

'Oggetto all'interno di in un simbolo nel sinottico

```

Public Sub Click()
    Dim objTrend As TrendCmdTarget
    Set objTrend = GetContainerObject.GetSubObject("Trend").GetObjectInterface
    Debug.Print objTrend.Samples
    Set objTrend = Nothing
End Sub

```

End Sub

GetOnScreenPosition, DrawCmdTarget Function

Syntax GetOnScreenPosition(_pnLeft, _pnTop, _pnRight, _pnBottom)

Description Questa funzione restituisce la posizione dell'oggetto all'interno del sinottico mediante quattro parametri di tipo Long indicanti rispettivamente le coordinate, espresse in pixel, dell'estremità sinistra, superiore, destra e inferiore dell'oggetto stesso.

Parameter	Description
pnLeft As Long	Coordinata sinistra.
pnTop As Long	Coordinata superiore.
pnRight As Long	Coordinata destra.
pnBottom As Long	Coordinata inferiore.

Result None

Example:

```
Public Sub Click()
    Dim nLeft As Long
    Dim nTop As Long
    Dim nRight As Long
    Dim nBottom As Long
    GetOnScreenPosition(nLeft, nTop, nRight, nBottom)
    MsgBox("Left =" & nLeft & " Top =" & nTop & " Right =" & nRight & " Bottom ="
    & nBottom ,vbOkOnly,GetProjectTitle)
End Sub
```

GetPolyPointOnScreenX, DrawCmdTarget Function

Syntax GetPolyPointOnScreenX(_nPos)

Description Questa funzione si utilizza nei disegni poligono per rilevare la coordinata X di un vertice del poligono. Con nPos si indica la posizione considerando che il primo vertice disegnato ha valore 0. Le coordinate sono espresse in numero di pixel e sempre riferite all'origine del sinottico.

Parameter	Description
nPos As Integer	Posizione del punto.

Result Integer

Example:

```
Public Sub Click()
    MsgBox("Coordinata X: " &
    GetPolyPointOnScreenX(0),vbOkOnly,GetProjectTitle)
End Sub
```

GetPolyPointOnScreenY, DrawCmdTarget Function

Syntax GetPolyPointOnScreenY(_nPos)

Description Questa funzione si utilizza nei disegni poligono per rilevare la coordinata Y di un vertice del poligono. Con nPos si indica la posizione considerando che il primo vertice disegnato ha valore 0. Le coordinate sono espresse in numero di pixel e sempre riferite all'origine del sinottico.

Parameter	Description
nPos As Integer	Posizione del punto.

Result Integer

Example:

```
Public Sub Click()
    MsgBox("Coordinata X: " &
        GetPolyPointOnScreenY(0),vbOkOnly,GetProjectTitle)
End Sub
```

GetSubGaugeObject, DrawCmdTarget Function

Syntax GetSubGaugeObject()

Description Permette di accedere alle proprietà ed ai metodi specifici di un oggetto gauge vettoriale inserito nel simbolo. Le proprietà e i metodi menzionati sono quelli descritti nell'interfaccia GaugeCmdTarget.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo GaugeCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()
    Dim ObjGauge As GaugeCmdTarget

    Set ObjGauge = GetContainerObject.GetSubGaugeObject("Gauge")
    If ObjGauge Is Nothing Then MsgBox "ObjGauge Is Nothing", vbExclamation +
        vbOkOnly, "GetSubGaugeObject"
    Set ObjGauge = Nothing
End Sub
```

GetSubObject, DrawCmdTarget Function

Syntax GetSubObject(_IpszObjectName)

Description Questa funzione consente di referenziare un oggetto contenuto all'interno di un simbolo. In pratica consente di accedere ad un disegno facente parte di un simbolo più complesso dal simbolo stesso. Tale funzione può essere utilizzata anche da un oggetto "Sinottico Incastrato" o "Gruppo Tab" per recuperare un oggetto contenuto al loro interno.

Parameter	Description
IpszObjectName As String	Nome oggetto.

Result Object
Viene restituito un oggetto di tipo DrawCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example 1:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget

    Set objRect = GetSubObject("Rect1")
    objRect.BackColor = objRect.BackColor + 10
    Set objRect = Nothing
End Sub
```

Example 2:

```
Dim objSyn As SynopticCmdTarget
Dim objEmbeddedScreen As DrawCmdTarget
Dim objDraw As DrawCmdTarget

Public Sub Click()
    Set objSyn = GetSynopticObject
    Set objEmbeddedScreen = objSyn.GetSubObject("oEmbeddedScreen")
    Set objDraw = objEmbeddedScreen.GetSubObject("objRect1")

    If (objDraw.BackColor <> vbRed) Then
        objDraw.BackColor = vbRed
    Else
        objDraw.BackColor = vbWhite
    End If

    Set objSyn = Nothing
    Set objEmbeddedScreen = Nothing
    Set objDraw = Nothing
End Sub
```

GetSubTrendObject, DrawCmdTarget Function

Syntax GetSubTrendObject()

Description Permette di accedere alle proprietà ed ai metodi specifici di un oggetto Trend inserito nel simbolo. Le proprietà e i metodi menzionati sono quelli descritti nell'interfaccia TrendCmdTarget.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo TrendCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()
    Dim ObjTrend As TrendCmdTarget

    Set ObjTrend = GetContainerObject.GetSubTrendObject("Trend")
```

```

        If ObjTrend Is Nothing Then MsgBox "ObjTrend Is Nothing", vbExclamation +
        vbOkOnly, "GetSubTrendObject"
        Set ObjTrend = Nothing
    End Sub

```

GetSynapseName, DrawCmdTarget Function

Syntax GetSynapseName(_nID)

Description Questa funzione restituisce il nome della sinapsi passando come parametro un identificatore numerico nID che è il numero progressivo di introduzione della sinapsi

Parameter	Description
nID As Integer	Numero progressivo di introduzione della sinapsi.

Result String

Example:

```

Public Sub Click()
    Dim Result As Integer
    Result = GetNumSynapsis
    If Result > 0 Then
        MsgBox("GetSynapseName =" &
        GetSynapseName(0), vbOkOnly, GetProjectTitle)
    end if
End Sub

```

GetSynapsePoint, DrawCmdTarget Function

Syntax GetSynapsePoint(_lpszSynapseName, _pnX, _pnY)

Description Questa funzione si utilizza nei disegni poligono per rilevare la coordinata X di un vertice del poligono. Con nPos si indica la posizione considerando che il primo vertice disegnato ha valore 0. Le coordinate sono espresse in numero di pixel e sempre riferite all'origine del sinottico.

Parameter	Description
lpszSynapseName As String	Nome della sinapsi definita nell'oggetto.
nX As Integer	Posizione orizzontale dal bordo sinistro dell'oggetto.
nY As Integer	Posizione orizzontale dal bordo destro dell'oggetto.

Result Boolean

Example:

```

Public Sub Click()
    Dim nX As Integer
    Dim nY As Integer

    ' Get synpase point (Trova le posizioni)
    GetSynapsePoint("IN", nX, nY)

```

```

' View result (Visualizza risultati)
MsgBox("X=" & nX & " Y=" & nY, , "GetSynapsePoint")
End Sub

```

GetSynopticObject, DrawCmdTarget Function

Syntax GetSynopticObject()

Description Permette di accedere alle proprietà ed ai metodi specifici del sinottico contenitore del simbolo. Le proprietà e i metodi menzionati sono quelli descritti nell'interfaccia SynopticCmdTarget.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo SynopticCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```

'Ambito Oggetto 1
Public Sub Click()
    Dim app As SynopticCmdTarget
    Dim obj2 As Object 'Oggetto 2

    Set app = GetSynopticObject()
    If app Is Nothing Then
        MsgBox("app is Nothing", vbOkOnly, GetProjectTitle)
    Else
        Set obj2 = app.GetSubObject("Oggetto 2")
        MsgBox(obj2.Title, vbOkOnly, GetProjectTitle)
        Set obj2 = Nothing
    End If
    Set app = Nothing
End Sub

```

GetTrendObject, DrawCmdTarget Function

Syntax GetTrendObject()

Description Questa funzione permette di accedere alle proprietà e metodi specifici di un oggetto Trend inserito nel sinottico. Le proprietà e metodi menzionati sono quelli descritti nell'interfaccia TrendCmdTarget.
L'utilizzo di questa funzione, a differenza della GetSubTrendObject, non rende necessario inserire il Trend in un simbolo.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo TrendCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```

Public Sub Click()
    Dim ObjTrend As TrendCmdTarget

```

```

Set ObjTrend = GetSynopticObject.GetSubObject("Trend").GetTrendObject
If ObjTrend Is Nothing Then MsgBox "ObjTrend Is Nothing", vbExclamation +
vbOkOnly, "GetTrendObject"
Set ObjTrend = Nothing
End Sub

```

GetUniqueObjectID, DrawCmdTarget Function

Syntax GetUniqueObjectID()

Description Questa funzione restituisce l'identificatore numerico associato automaticamente da Movicon ad un disegno che contenga sinapsi.
Il valore di ritorno 0 indica che l'oggetto disegno non contiene sinapsi definite al suo interno.

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
    MsgBox("Numero di ID: " & GetUniqueObjectID,vbOkOnly,GetProjetTitle)
End Sub

```

GetXMLSettings, DrawCmdTarget Function

Syntax GetXMLSettings()

Description La funzione restituisce la stringa di definizione dell'oggetto nel progetto in formato XML.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    MsgBox("GetXMLSettings =" & GetXMLSettings,vbOkOnly,GetProjectTitle)
End Sub

```

HasSynapsis, DrawCmdTarget Function

Syntax HasSynapsis()

Description Questa funzione identifica se l'oggetto contiene delle sinapsi. Il valore di ritorno true indica la presenza di almeno una sinapsi definita nel simbolo, viceversa ritorna false.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    If (HasSynapsis()) Then  
        MsgBox "HasSynapsis = " & CStr(HasSynapsis()), vbOkOnly, "Test  
        HasSynapsis"  
    End If  
End Sub
```

InflateObject, DrawCmdTarget Function

Syntax InflateObject(_nCX, _nCY)

Description Modifica la dimensione dell'oggetto in funzione dei due parametri interi nCX per la proporzione su asse X e nCY per la proporzione su asse Y. La riduzione avviene per il numero di pixel indicato nei parametri in entrambi i lati del simbolo (cioè se nX = 5 la dimensione verrà aumentata di 5 pixel a destra e di 5 pixel a sinistra).

Parameter	Description
nCX As Integer	Proporzione sull'asse x.
nCY As Integer	Proporzione sull'asse y.

Result None

Example:

```
Public Sub Click()  
    InflateObject(5,5)  
    MsgBox "Width=" & width & " Height=" & height & " Xpos=" & Xpos & " Ypos=" & Ypos, vbOkOnly, GetProjectTitle  
End Sub
```

IsCursorOnObject, DrawCmdTarget Function

Syntax IsCursorOnObject()

Description Questa funzione controlla se il puntatore mouse è all'interno dell'oggetto. Se così restituisce True, diversamente restituisce False.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If (IsCursorOnObject()) Then
        MsgBox "IsCursorOnObject = " & CStr(IsCursorOnObject()), vbOkOnly, "Test
        IsCursorOnObject "
    End If
End Sub
```

IsGlobalObjectName, DrawCmdTarget Function

Syntax IsGlobalObjectName()

Description Questa funzione verifica se il nome dell'oggetto è impostato come globale.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    MsgBox("IsGlobalObjectName: " & CStr(IsGlobalObjectName),
    vbOkOnly,GetProjectTitle)
End Sub
```

IsSynapseConnected, DrawCmdTarget Function

Syntax IsSynapseConnected(_IpszSynapseName)

Description Questa funzione permette di verificare se una sinapsi è collegata ad un oggetto connettore, identificandone il nome attraverso il parametro IpszSynapseName. La funzione ritorna il valore true se la sinapsi è collegata ad un oggetto connettore.

Parameter	Description
IpszSynapseName As String	Nome della sinapsi definita nell'oggetto.

Result Boolean

Example:

```

Public Sub Click()
    Dim Result As Boolean
    Result = IsSynapseConnected("IN")
    MsgBox("IsSynapseConnected =" & Result,vbOkOnly , "GetSynapsePoint")
End Sub

```

LoadExtSettings, DrawCmdTarget Function

Syntax LoadExtSettings

Description Questa funzione permette di caricare la configurazione dell'oggetto dal file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub Click()
    Dim objSymbol As DrawCmdTarget
    Set objSymbol = GetSynopticObject.GetSubObject("TestObject")
    If objSymbol Is Nothing Then Exit Sub
    objSymbol.ExtSettingsFile = "test.sxml"
    objSymbol.LoadExtSettings
    Set objSymbol = Nothing
End Sub

```

MoveObject, DrawCmdTarget Function

Syntax MoveObject(_nLeft, _nTop, _nRight, _nBottom)

Description Questa funzione esegue lo spostamento ed il dimensionamento dell'oggetto in funzione dei parametri specificati in pixels come valori di tipo Integer. I parametri richiesti sono Left (quota lato sinistro), Top (quota lato Alto), Right (quota lato destro) e Bottom (quota lato basso). Tutti i parametri fanno sempre riferimento all'angolo superiore sinistro dello schermo.

Parameter	Description
nLeft As Integer	Valore coordinata x angolo in alto a sinistra.
nTop As Integer	Valore coordinata y angolo in alto a sinistra.
nRight As Integer	Valore coordinata x angolo in basso a destra.
nBottom As Integer	Valore coordinata y angolo in basso a destra.

Result None

Example:

```
Public Sub Click()
    MoveObject(100,200,330,450)
    MsgBox "Width=" & width & " Height=" & height & " Xpos=" & Xpos & " Ypos="
    & Ypos, vbOkOnly, GetProjectTitle
End Sub
```

OffsetObject, DrawCmdTarget Function

Syntax OffsetObject(_nX, _nY)

Description Provoca uno spostamento dell'oggetto nelle direzioni X e Y del numero di pixel specificati dai parametri interi nX per la direzione X e nY per la direzione Y. La direzione positiva per le X è verso il basso dello schermo mentre per le Y è verso la destra dello schermo. Questa proprietà lavora solo in impostazione, non restituisce cioè nessun valore.

Parameter	Description
nX As Integer	Valore spostamento lungo l'asse x.
nY As Integer	Valore spostamento lungo l'asse y.

Result None

Example:

```
Public Sub Click()
    OffsetObject(100,200)
    MsgBox "Width=" & width & " Height=" & height & " Xpos=" & Xpos & " Ypos="
    & Ypos, vbOkOnly, GetProjectTitle
End Sub
```

PolyPointX, DrawCmdTarget Function

Syntax PolyPointX(_nPos)

Description Questa proprietà si utilizza nei disegni poligono per rilevare la coordinata X di un vertice o per modificarla. Con nPos si indica la posizione considerando che il primo disegnato ha valore 0. Le coordinate sono espresse in numero di pixel e riferite all'origine del contenitore del disegno: sinottico o simbolo.

Parameter	Description
nPos As Integer	Posizione del punto (0 è il primo disegnato).

Result Integer

Example:

```
Public Sub Click()
    MsgBox "PolyPointX = " & PolyPointX(0) , vbOkOnly, GetProjectTitle
End Sub
```

PolyPointY, DrawCmdTarget Function

Syntax PolyPointY(_nPos)

Description Questa proprietà si utilizza nei disegni poligono per rilevare la coordinata Y di un vertice o per modificarla. Con nPos si indica la posizione considerando che il primo disegnato ha valore 0. Le coordinate sono espresse in numero di pixel e riferite all'origine del contenitore del disegno: sinottico o simbolo.

Parameter	Description
nPos As Integer	Posizione del punto (0 è il primo disegnato).

Result Integer

Example:

```
Public Sub Click()  
    MsgBox "PolyPointY= " & PolyPointY(0) , vbOkOnly, GetProjectTitle  
End Sub
```

PrintThisSynoptic, DrawCmdTarget Function

Syntax PrintThisSynoptic(_nMode, _bKeepPrintProportions)

Description Questa funzione effettua la stampa andando a ricaricare in background il Sinottico rieseguendo pertanto la SynopticLoading(). Per questo motivo eventuali modifiche ad oggetti dinamici, effettuate dopo il caricamento del Sinottico stesso, non vengono riportate nella stampa. Accetta il parametro nMode indicante il modo di stampa.

Il parametro può assumere i seguenti valori:

0=seleziona stampante

1=stampa diretta

2=Anteprima



Questa proprietà non è supportata in Windows CE

Parameter	Description
nMode As Integer	Modalità di stampa.
bKeepPrintProportions as boolean	Parametro Opzionale. Se impostato a "True" permette di stampare mantenendo le proporzioni che si vedono sullo schermo, mentre se impostato a "False" (valore di default) vengono adattate sia l'altezza che la larghezza all'intera dimensione della pagina.

Result None

Example:

```
'Ambito Sinottico  
Public Sub Click()  
    PrintThisSynoptic(2)  
End Sub
```

Prop, DrawCmdTarget Function

Syntax Prop(_IpszPropName)

Description Imposta o restituisce il valore di una proprietà definita sul disegno dal parametro IpszPropName. Le nuove proprietà sono persistenti, cioè il loro valore rimane anche dopo il salvataggio e la chiusura del progetto. Utilizzando l'evento OnCustomizeSymbol nella libreria dei template è possibile personalizzare l'inserimento dei simboli sul sinottico, andando a creare ed impostare delle proprietà con la funzione Prop, utilizzate poi in lettura nel restante codice del template.

Parameter	Description
IpszPropName As String	Nome della proprietà.

Result String

Example:

'Supponiamo di avere in un sinottico due simboli rappresentanti due pulsanti nominati "P1" e "P2".

'All'interno del simbolo "P1" possiamo scrivere il seguente codice:

```
Public Sub Click()  
    If Prop("Premuto") = "ON" Then  
        Prop("Premuto") = "OFF"  
    Else  
        Prop("Premuto") = "ON"  
    End If  
End Sub  
  
'Nel simbolo "P2" invece andiamo a testare la proprietà "Premuto" di "P1" nel modo seguente:  
Dim syn As Object  
Dim rett As Object  
Public Sub SymbolLoading()  
    Set syn = GetSynopticObject  
    Set rett = syn.GetSubObject("P1")  
End Sub  
Public Sub SymbolUnloading()  
    Set syn = Nothing  
    Set rett = Nothing  
End Sub  
Public Sub Click()  
    Debug.Print rett.Prop("Premuto")  
End Sub
```

RemoveAlias, DrawCmdTarget Function

Syntax RemoveAlias(_IpszAlias)

Description Questa funzione consente di eliminare l'Alias passato come parametro "IpszAlias".

Parameter	Description
IpszAlias As String	Nome dell'Alias da eliminare.

Result Boolean

Example:

```
Public Sub Click()
    Debug.Print RemoveAlias("TsetAlais")
End Sub
```

RemoveAllAliases, DrawCmdTarget Function

Syntax RemoveAllAliases()

Description Questa funzione consente di eliminare tutti gli Alias definiti nella Tabella dell'oggetto.

Parameter	Description
None	None

Result None

Example:

```
Public Sub Click()
    RemoveAllAliases()
End Sub
```

ResetColors, DrawCmdTarget Function

Syntax ResetColors()

Description Questa funzione esegue un azzeramento delle impostazioni di colore presenti sull'oggetto. Le impostazioni riguardano il contrasto, la luminosità, la scala di grigi, il colore di sfondo e il colore della linea/testo.

Parameter	Description
None	None

Result None

Example:

```
Public Sub Click()
    BackColorBrightness =200
    IRet = BackColorBrightness
    MsgBox "BackColorBrightness = " & CStr(IRet), vbOkOnly, GetProjectTitle
    ResetColors
    MsgBox "ResetColors hes been done", vbOkOnly, GetProjectTitle
End Sub
```

SaveExtSettings, DrawCmdTarget Function

Syntax SaveExtSettings

Description Questa funzione permette di salvare la configurazione dell'oggetto nel file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
Dim objSymbol As DrawCmdTarget  
Set objSymbol = GetSynopticObject.GetSubObject("TestObject")  
If objSymbol Is Nothing Then Exit Sub  
objSymbol.ExtSettingsFile = "test.sxml"  
objSymbol.SaveExtSettings  
Set objSymbol = Nothing  
End Sub
```

ScaleObject, DrawCmdTarget Function

Syntax ScaleObject(_lpar)

Description E' un metodo che ingrandisce/rimpicciolisce l'oggetto in base alla percentuale definita tramite il parametro intero che gli viene passato. Ad esempio il valore 150 ingrandirà l'oggetto una volta e mezzo le dimensioni originali (attuali), il valore 50 diminuisce le dimensioni dell'oggetto alla metà di quelle originali (attuali).

Parameter	Description
lpar As Integer	Valore di ingrandimento/rimpicciolimento.

Result None

Example:

```
Public Sub Click()  
    ScaleObject(80)  
End Sub
```

SetAlias, DrawCmdTarget Function

Syntax SetAlias(_lpszAlias, _lpszValue)

Description Questa funzione imposta il valore definito per l'Alias passato come parametro "lpszAlias". Il nuovo valore sarà quello passato nel parametro "lpszValue", e potrà essere il nome di una variabile o un valore numerico o stringa. Se l'Alias non esiste nella Tabella dell'oggetto verrà aggiunto come nuovo Alias.



Utilizzando la funzione "SetAlias" all'interno dell'evento "OnPreSymbolLoading", gli alias dinamici diventano in realtà statici. Questo significa che utilizzando di nuovo la funzione "SetAlias" dopo l'esecuzione dell'evento "OnPreSymbolLoading" il valore dell'alias non verrà più sostituito.

Utilizzando la funzione "SetAlias" all'interno dell'evento "SymbolLoading", gli alias dinamici rimarranno tali. Questo significa che utilizzando di nuovo la funzione "SetAlias" dopo l'esecuzione dell'evento "SymbolLoading" il valore dell'alias verrà sostituito nuovamente.

Parameter	Description
IpszAlias As String	Nome dell'Alias per il quale impostare il valore. Se non esiste verrà aggiunto come nuovo Alias.
IpszValue As String	Valore con cui impostare l'Alias.

Result Boolean

Example:

```
Public Sub Click()
    Debug.Print SetAlias("TsetAlais", "VAR00001")
End Sub
```

ShowPropList, DrawCmdTarget Function

Syntax ShowPropList()

Description Questa funzione mostra un elenco delle proprietà create mediante la funzione "Prop" e i relativi valori purché tali proprietà siano riconducibili a valori booleani.

Parameter	Description
None	None

Result None

Example:

```
'Ambito Sinottico
Public Sub Click()
    ShowPropList
End Sub
```

SynapsePassExecution, DrawCmdTarget Function

Syntax SynapsePassExecution(_nID)

Description Questa funzione passa l'esecuzione della macro all'oggetto disegno successivo, normalmente passata attraverso l'uso delle funzioni SynapseValue o SynapseValueFromID.
Questo metodo risulta utile per quegli oggetti ciechi, ovvero oggetti che hanno solo delle sinapsi di input e desiderano passare l'esecuzione ma non hanno nessuna sinapsi di output da settare.

Parameter	Description
-----------	-------------

None	None
------	------

Result None

Example:

```
Public Sub OnFireExecution ()
    ' Continue the logic execution (continua l'esecuzione logica)
    SynapsePassExecution
End Sub
```

ZOrderMoveBack, DrawCmdTarget Function

Syntax ZOrderMoveToBack()

Description Questa funzione modifica la posizione dell'oggetto nell'ordinamento degli oggetti nel sinottico di una posizione indietro.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    ZOrderMoveBack
End Sub
```

ZOrderMoveForward, DrawCmdTarget Function

Syntax ZOrderMoveForward()

Description Questa funzione modifica la posizione dell'oggetto nell'ordinamento degli oggetti nel sinottico di una posizione avanti.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    ZOrderMoveForward
End Sub
```

ZOrderMoveToBack, DrawCmdTarget Function

Syntax ZOrderMoveToBack()

Description Questa funzione modifica la posizione dell'oggetto nell'ordinamento dei simboli nel sinottico portandolo all'inizio dell'ordine, cioè "sotto" a tutti gli altri simboli.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    ZOrderMoveToBack  
End Sub
```

ZOrderMoveToFront, DrawCmdTarget Function

Syntax ZOrderMoveToFront()

Description Questa funzione modifica la posizione dell'oggetto nell'ordinamento dei simboli nel sinottico portandolo alla fine dell'ordine, cioè "sopra" a tutti gli altri simboli.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    ZOrderMoveToFront  
End Sub
```

Properties

AdaptFontSize, DrawCmdTarget Property

Syntax AdaptFontSize = Boolean

Description Questa proprietà imposta o restituisce l'adattamento della dimensione del font del testo alle dimensioni dell'oggetto.

Parameter	Description
-----------	-------------

None	None
------	------

Result Boolean

Example:

```
Public Sub Click()
    AdaptFontSize = Not AdaptFontSize
    MsgBox "AdaptFontSize = " & CStr(AdaptFontSize), vbOkOnly, "Test
    AdaptFontSize"
End Sub
```

AlignFont, DrawCmdTarget Property

Syntax AlignFont = Integer

Description Questa proprietà imposta o restituisce l'allineamento del testo nell'oggetto secondo le possibilità Al Centro, A Destra, A Sinistra, In Alto, in Basso, Centro-Sinistra e Centro-Destra disponibili anche nelle proprietà di stile del disegno. Inoltre è disponibile un Enumerale di tipo "_movicon.efontFormat" per effettuare la scelta del Tipo di Allineamento:

```
enum_fft_center = 0 (al centro)
enum_fft_top = 1 (in alto)
enum_fft_bottom = 2 (in basso)
enum_fft_left = 3 (a sinistra)
enum_fft_right = 4 (a destra)
enum_fft_centerleft = 5 (al centro-sinistra)
enum_fft_centreright = 6 (al centro- destra)
```

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    Dim iRet As Integer
    AlignFont = enum_fft_center
    iRet = AlignFont
    MsgBox "AlignFont = " & CStr(iRet), vbOkOnly, "Test AlignFont"
    AlignFont = enum_fft_top
    iRet = AlignFont
    MsgBox "AlignFont = " & CStr(iRet), vbOkOnly, "Test AlignFont"
    AlignFont = enum_fft_bottom
    iRet = AlignFont
    MsgBox "AlignFont = " & CStr(iRet), vbOkOnly, "Test AlignFont"
    AlignFont = enum_fft_left
    iRet = AlignFont
    MsgBox "AlignFont = " & CStr(iRet), vbOkOnly, "Test AlignFont"
    AlignFont = enum_fft_right
    iRet = AlignFont
    MsgBox "AlignFont = " & CStr(iRet), vbOkOnly, "Test AlignFont"
    AlignFont = enum_fft_centerleft
    iRet = AlignFont
    MsgBox "AlignFont = " & CStr(iRet), vbOkOnly, "Test AlignFont"
    AlignFont = enum_fft_centreright
    iRet = AlignFont
    MsgBox "AlignFont = " & CStr(iRet), vbOkOnly, "Test AlignFont"
```

End Sub

AlignFontOffsetX, DrawCmdTarget Property

Syntax AlignFontOffsetX= _Long

Description Questa proprietà imposta o restituisce il valore della proprietà "Offset X" e cioè l'Offset della posizione 'X' in pixel del testo impostato nella proprietà Titolo di un oggetto. L'offset verrà applicato alla posizione del titolo definita tramite la proprietà di sviluppo "Allineamento Testo" dell'oggetto stesso.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim sRet As Long  
  
    sRet = InputBox "Insert Offset X"  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    objRect.AlignFontOffsetX = sRet  
  
    MsgBox "AlignFontOffsetX= " & objRect.AlignFontOffsetX, vbOkOnly,  
    GetProjectTitle  
    Set objRect = Nothing  
End Sub
```

AlignFontOffsetY, DrawCmdTarget Property

Syntax AlignFontOffsetY= _Long

Description Questa proprietà imposta o restituisce il valore della proprietà "Offset Y" e cioè l'Offset della posizione 'Y' in pixel del testo impostato nella proprietà Titolo di un oggetto. L'offset verrà applicato alla posizione del titolo definita tramite la proprietà di sviluppo "Allineamento Testo" dell'oggetto stesso.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim sRet As Long
```

```

sRet = InputBox "Insert Offset y"

Set objRect = GetSynopticObject.GetSubObject("objRect")
objRect.AlignFontOffsety = sRet

MsgBox "AlignFontOffsety= " & objRect.AlignFontOffsety, vbOkOnly,
GetProjectTitle
Set objRect = Nothing
End Sub

```

AntialiasingFont, DrawCmdTarget Property

Syntax AntialiasingFont = Boolean

Description Questa proprietà imposta o restituisce il valore della proprietà "Usa Antialiasing" di un oggetto, tramite la quale verrà gestito l'Antialiasing sul Font del Titolo dell'oggetto.

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub Click()
    AntialiasingFont = Not AntialiasingFont
    MsgBox "AntialiasingFont = " & CStr(AntialiasingFont), vbOkOnly, "Test
    AntialiasingFont"
End Sub

```

AutoRepeatClick, DrawCmdTarget Property

Syntax AutoRepeatClick = _Integer

Description Questa proprietà (default a zero) consente quando impostata di ottenere con una certa frequenza l'evento click mentre l'utente mantiene premuto il tasto del mouse sul disegno. La frequenza è il multiplo della frequenza di esecuzione delle procedure basic OnTimer. Accetta un valore Integer.

Parameter	Description
None	None

Result Integer

Example:

```

Public Sub SymbolLoading()
    AutoRepeatClick = 1
End Sub
Dim nCounter As Integer
Public Sub Click()
    nCounter = nCounter + 1
    Debug.Print "Varore di conteggio = " & CStr(nCounter )
End Sub

```

BackBrushPattern, DrawCmdTarget Property

Syntax BackBrushPattern = Integer

Description Questa proprietà imposta o restituisce il motivo di sfondo dell'oggetto secondo le possibilità offerte anche dagli attributi di riempimento "Motivo di Sfondo". Sono validi i valori compresi fra 0 e 22 che raggruppano una serie di motivi di sfondo (righe verticali, punti, mattoni, tessuto, ecc...). Con il valore -1 (FFFF esadecimale) si elimina il motivo di sfondo applicato, che risulta quindi essere impostato con l'opzione "Solido".

Esistono tre motivi di sfondo di un oggetto:

1 = Trasparente
2 = Solido
3 = Pattern

Per impostare lo sfondo come solido o trasparente si deve utilizzare la funzione "BackBrushVisible". La selezione del motivo di sfondo sarà quindi possibile solo se lo sfondo non è impostato come "Trasparente".



Questa proprietà non è pienamente supportata in Windows CE (ammette soltanto il valore -1).

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim iRet As Integer  
    BackBrushPattern = 0  
    iRet = BackBrushPattern  
    MsgBox "BackBrushPattern = " & CStr(iRet), vbOkOnly, "Test BackBrushPattern"  
    BackBrushPattern = &HFFFF  
    iRet = BackBrushPattern  
    MsgBox "BackBrushPattern = " & CStr(iRet), vbOkOnly, "Test BackBrushPattern"  
    BackBrushPattern = 22  
    iRet = BackBrushPattern  
    MsgBox "BackBrushPattern = " & CStr(iRet), vbOkOnly, "Test BackBrushPattern"  
End Sub
```

BackBrushVisible, DrawCmdTarget Property

Syntax BackBrushVisible = Boolean

Description Imposta o restituisce la condizione di visibilità dello sfondo del disegno. Il valore "False" imposta lo sfondo come trasparente, il valore "True" lo imposta come "Solido". quando lo sfondo è impostato come "Solido" è poi possibile cambiare il motivo di sfondo con la funzione "BackBrushPattern".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    BackBrushVisible = Not BackBrushVisible  
    MsgBox "BackBrushVisible = " & CStr(BackBrushVisible), vbOkOnly, "Test  
    BackBrushVisible"  
End Sub
```

BackColor, DrawCmdTarget Property

Syntax BackColor = Long

Description Questa proprietà imposta o restituisce il colore di sfondo dell'oggetto. Il valore passato o restituito contiene il codice (R,G,B, su ogni byte) del colore di sfondo. Può essere utile utilizzare la funzione RGB per identificare in modo più semplice il colore.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim IRet As Long  
    BackColor = RGB(255,0,0)  
    IRet = BackColor  
    MsgBox "BackColor = " & CStr(IRet), vbOkOnly, "Test BackColor"  
    BackColor = RGB(0,255,0)  
    IRet = BackColor  
    MsgBox "BackColor = " & CStr(IRet), vbOkOnly, "Test BackColor"  
    BackColor = RGB(0,0,255)  
    IRet = BackColor  
    MsgBox "BackColor = " & CStr(IRet), vbOkOnly, "Test BackColor"  
End Sub
```

BackColorBrightness, DrawCmdTarget Property

Syntax BackColorBrightness = _Integer

Description Questa proprietà imposta e restituisce la componente di luminosità (Brightness) associata al colore di sfondo del disegno. Ogni colore è rappresentato da un valore long che può essere suddiviso in tre componenti intere: tonalità, saturazione e luminosità. I valori ammessi per questa proprietà vanno dallo 0 al 255.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim IRet As Integer  
    BackColorBrightness = 185  
    IRet = BackColorBrightness  
    MsgBox "BackColorBrightness = " & CStr(IRet), vbOkOnly, "Test  
    BackColorBrightness "  
    BackColorBrightness = 192  
    IRet = BackColorBrightness  
    MsgBox "BackColorBrightness = " & CStr(IRet), vbOkOnly, "Test  
    BackColorBrightness "  
    BackColorBrightness = 200  
    IRet = BackColorBrightness  
    MsgBox "BackColorBrightness = " & CStr(IRet), vbOkOnly, "Test  
    BackColorBrightness "  
End Sub
```

BackColorHue, DrawCmdTarget Property

Syntax BackColorHue = _Integer

Description Questa proprietà imposta e restituisce la componente di tonalità (Hue) associata al colore di sfondo del disegno. Ogni colore è rappresentato da un valore long che può essere suddiviso in tre componenti intere: tonalità, saturazione e luminosità. I valori ammessi per questa proprietà vanno dallo 0 al 255.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim IRet As Integer  
    BackColorHue = 185  
    IRet = BackColorHue  
    MsgBox "BackColor = " & CStr(IRet), vbOkOnly, "Test BackColorHue "  
    BackColorHue = 192  
    IRet = BackColorHue  
    MsgBox "BackColorHue = " & CStr(IRet), vbOkOnly, "Test BackColorHue "  
    BackColor = 200  
    IRet = BackColorHue  
    MsgBox "BackColorHue = " & CStr(IRet), vbOkOnly, "Test BackColorHue "  
End Sub
```

BackColorSaturation, DrawCmdTarget Property

Syntax BackColorSaturation = _Integer

Description Questa proprietà imposta e restituisce la componente di saturazione (Saturation) associata al colore di sfondo del disegno. Ogni colore è rappresentato da un valore long che può essere suddiviso in tre componenti intere: tonalità, saturazione e luminosità. I valori ammessi per questa proprietà vanno dallo 0 al 255.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    Dim IRet As Integer
    BackColorSaturation = 185
    IRet = BackColorSaturation
    MsgBox "BackColorSaturation = " & CStr(IRet), vbOkOnly, "Test
    BackColorSaturation "
    BackColorSaturation = 192
    IRet = BackColorSaturation
    MsgBox "BackColorSaturation = " & CStr(IRet), vbOkOnly, "Test
    BackColorSaturation "
    BackColorSaturation = 200
    IRet = BackColorSaturation
    MsgBox "BackColorSaturation = " & CStr(IRet), vbOkOnly, "Test
    BackColorSaturation "
End Sub
```

BitmapAlignment, DrawCmdTarget Property

Syntax BitmapAlignment= eImageAligns

Description Questa proprietà imposta o restituisce l'indice del tipo di allineamento dell'immagine impostata come Immagine Statica delle proprietà Sfondo dell'oggetto. Il tipo di allineamento si può specificare usando l'enumeratore eImageAligns oppure inserendo il corrispondente valore numerico:

```
enum_ima_stretch = 0 (Stratched)
enum_ima_topleft = 1 (Top-Left)
enum_ima_topcenter = 2 (Top-Center)
enum_ima_topright = 3 (Top-Right)
enum_ima_centerleft = 4 (Center-Left)
enum_ima_center = 5 (Center)
enum_ima_centreright = 6 (Center-Right)
enum_ima_bottomleft = 7 (Bottom-Left)
enum_ima_bottomcenter = 8 (Bottom-Center)
enum_ima_bottomright = 9 (Bottom-Right)
```

Impostando la proprietà al valore 0 (Stretched) verrà di conseguenza settata anche la proprietà "BitmapStretched" a True e viceversa.

Parameter	Description
None	None

Result eImageAligns

Example:

```
Public Sub Click()

    Select Case BitmapAlignment
    Case enum_ima_stretch
    MsgBox "BitmapAlignment = Stratched(" & CStr(BitmapAlignment) & ")"
```

```

Case enum_ima_topleft
MsgBox "BitmapAlignment = TopLeft(" & CStr(BitmapAlignment) & ")"
Case enum_ima_topcenter
MsgBox "BitmapAlignment = TopCenter(" & CStr(BitmapAlignment) & ")"
Case enum_ima_topright
MsgBox "BitmapAlignment = TopRight(" & CStr(BitmapAlignment) & ")"
Case enum_ima_centerleft
MsgBox "BitmapAlignment = CenterLeft(" & CStr(BitmapAlignment) & ")"
Case enum_ima_center
MsgBox "BitmapAlignment = Center(" & CStr(BitmapAlignment) & ")"
Case enum_ima_centreright
MsgBox "BitmapAlignment = CenterRight(" & CStr(BitmapAlignment) & ")"
Case enum_ima_bottomleft
MsgBox "BitmapAlignment = BottomLeft(" & CStr(BitmapAlignment) & ")"
Case enum_ima_bottomcenter
MsgBox "BitmapAlignment = BottomCenter(" & CStr(BitmapAlignment) & ")"
Case enum_ima_bottomright
MsgBox "BitmapAlignment = BottomRight(" & CStr(BitmapAlignment) & ")"
End Select

```

End Sub

BitmapID, DrawCmdTarget Property

Syntax BitmapID = _String

Description Questa proprietà permette di leggere o impostare l'immagine da rappresentare sopra all'oggetto. La stringa passata o restituita corrispondente al nome del file immagine.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Debug.Print BitmapID
End Sub

```

BitmapKeepAspectRatio, DrawCmdTarget Property

Syntax BitmapStretched = _Boolean

Description Questa proprietà di sola lettura restituisce l'adattamento proporzionale dello stretching dell'immagine impostata come Immagine Statica delle proprietà Sfondo dell'oggetto.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    objRect.BitmapKeepAspectRatio = Not(objRect.BitmapKeepAspectRatio)  
  
    Set objRect = Nothing  
End Sub
```

BitmapOffsetX, DrawCmdTarget Property

Syntax BitmapOffsetx= _Integer

Description Questa proprietà imposta o restituisce la posizione X in pixel dell'immagine visualizzata come Immagine Statica rispetto alle impostazioni fatte sulla proprietà "Allineamento Immagine" delle proprietà Sfondo dell'oggetto.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim sRet As Integer  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    sRet = objRect.BitmapOffsetX  
    MsgBox "BitmapOffsetX= " & sRet, vbOkOnly, GetProjectTitle  
    Set objRect = Nothing  
End Sub
```

BitmapOffsetY, DrawCmdTarget Property

Syntax BitmapOffsety= _Integer

Description Questa proprietà imposta o restituisce la posizione Y in pixel dell'immagine visualizzata come Immagine Statica rispetto alle impostazioni fatte sulla proprietà "Allineamento Immagine" delle proprietà Sfondo dell'oggetto.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim sRet As Integer  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    sRet = objRect.BitmapOffsetY  
  
    MsgBox "BitmapOffsetY= " & sRet, vbOkOnly, GetProjectTitle  
    Set objRect = Nothing  
End Sub
```

BitmapStretched, DrawCmdTarget Property

Syntax BitmapStretched = _Boolean

Description Questa proprietà imposta o restituisce l'abilitazione allo stretching sull'immagine impostata come Immagine Statica delle proprietà Sfondo dell'oggetto. Questa proprietà viene sempre modificata a True in conseguenza dell'impostazione della proprietà "BitmapAlignment" al valore 0 (Stretched) e viceversa

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
  
    Set objRect = GetSynopticObject.GetSubObject("objRect")  
    objRect.BitmapStretched = Not(objRect.BitmapStretched)  
  
    Set objRect = Nothing  
End Sub
```

BitmapTransparent, DrawCmdTarget Property

Syntax BitmapTransparent = Boolean

Description Questa proprietà rende attiva o disattiva la proprietà di trasparenza dell'eventuale immagine visualizzata nell'oggetto.

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub Click()
    BitmapTransparent = Not BitmapTransparent
    MsgBox "BitmapTransparent = " & CStr(BitmapTransparent), vbOkOnly, "Test
    BitmapTransparent "
End Sub

```

BitmapTransparentColor, DrawCmdTarget Property

Syntax **BitmapTransparentColor** = _Long

Description Questa proprietà imposta o restituisce il colore che si desidera rendere trasparente nell'eventuale immagine visualizzata sull'oggetto disegno.

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
    BitmapTransparentColor = RGB(255,255,255)
    MsgBox "BitmapTransparentColor = " & CStr(BitmapTransparentColor),
    vbOkOnly, GetProjectTitle 'Return-> 16777215
End Sub

```

BorderType, DrawCmdTarget Property

Syntax **BorderType** = _Integer

Description Questa proprietà imposta o restituisce il tipo di bordo impostato per l'oggetto secondo le possibilità Nessuno, A Sbalzo, Inciso, In Rilievo ed Infossato disponibili anche nelle proprietà generali del disegno.
Sono validi i seguenti valori: 0=nessuno, 1=a sbalzo, 2=inciso, 3=in rilievo, 4=infossato.

Parameter	Description
None	None

Result Integer

Example:

```

Public Sub Click()
    For i = 0 To 4 Step 1
        BorderType = i
        sRet = BorderType
        MsgBox "BorderType = " & sRet, vbOkOnly, GetProjectTitle
    Next i
End Sub

```

CenterRotation, DrawCmdTarget Property

Syntax CenterRotation = Boolean

Description Questa proprietà imposta o restituisce l'abilitazione del baricentro di rotazione dell'oggetto al centro dell'oggetto stesso.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre 'true')

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    CenterRotation = Not CenterRotation  
    MsgBox "CenterRotation = " & CStr(CenterRotation ), vbOkOnly, "Test  
    CenterRotation"  
End Sub
```

DefStructName, DrawCmdTarget Property

Syntax DefStructName = _String

Description Questa proprietà imposta o restituisce la variabile di tipo struttura definita per l'oggetto.
Per ogni disegno o simbolo è possibile associare una variabile di tipo struttura e identificare all'interno del simbolo/oggetto le variabili membro omettendo il nome della variabile struttura. Esempio: ":HighLevel" invece di "Pump:HighLevel".

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sRet As String  
    Dim IRet As Variant  
    DefStructName= "STRUCT1"  
    sRet = DefStructName  
    MsgBox "DefStructName = " & sRet, vbOkOnly, GetProjectTitle  
    sRet = sRet & ":VAR00001"  
    IRet = GetVariableValue(sRet)  
    MsgBox "VAR00001 Value = " & IRet, vbOkOnly, GetProjectTitle  
End Sub
```

DefStructNameAbsolute, DrawCmdTarget Property

Syntax DefStructNameAbsolute = _String

Description Questa proprietà imposta o restituisce la variabile di tipo struttura definita per l'oggetto contenitore (simbolo) del disegno. Quindi questa proprietà è valida per un disegno contenuto all'interno di un simbolo.
Per ogni disegno o simbolo è possibile associare una variabile di tipo struttura e identificare all'interno del simbolo le variabili membro omettendo il nome della variabile struttura. Esempio: ":HighLevel" invece di "Pump:HighLevel".

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sRet As String  
    Dim lRet As Variant  
    DefStructNameAbsolute = "STRUCT1"  
    sRet = DefStructNameAbsolute  
    MsgBox "DefStructNameAbsolute = " & sRet, vbOkOnly, GetProjectTitle  
    sRet = sRet & ":VAR00001"  
    lRet = GetVariableValue(sRet)  
    MsgBox "VAR00001 Value = " & lRet, vbOkOnly, GetProjectTitle  
End Sub
```

DrawingState, DrawCmdTarget Property

Syntax DrawingState = _Integer

Description Questa proprietà imposta o restituisce il tipo di visualizzazione associata al colore di sfondo del disegno. Sono possibili valori compresi fra 0 e 3 che corrispondono alle seguenti impostazioni: 0=Normale, 1=SemiTrasparente, 2=Disabilitato e 3=Dither.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre -1)

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim lRet as Integer  
    DrawingState =0  
    lRet = DrawingState  
    MsgBox "DrawingState = " & CStr(lRet), vbOkOnly, "Test DrawingState "  
    DrawingState =1  
    lRet = DrawingState  
    MsgBox "DrawingState = " & CStr(lRet), vbOkOnly, "Test DrawingState "
```

```

DrawingState =2
IRet = DrawingState
MsgBox "DrawingState = " & CStr(IRet), vbOkOnly, "Test DrawingState "
DrawingState =3
IRet = DrawingState
MsgBox "DrawingState = " & CStr(IRet), vbOkOnly, "Test DrawingState "
End Sub

```

DrawingStateShadow, DrawCmdTarget Property

Syntax DrawingStateShadow = _Integer

Description Questa proprietà imposta o restituisce il tipo di visualizzazione associata al colore di sfondo dell'ombra del disegno. Sono possibili valori compresi fra 0 e 3 che corrispondono alle seguenti impostazioni: 0=Normale, 1=SemiTrasparente, 2=Disabilitato e 3=Dither. Tale proprietà si può utilizzare se la proprietà Shadow è impostata a True.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre -1)

Parameter	Description
None	None

Result Integer

Example:

```

Public Sub Click()
    Dim IRet as Integer
    DrawingStateShadow =0
    IRet = DrawingStateShadow
    MsgBox "DrawingStateShadow = " & CStr(IRet), vbOkOnly, "Test DrawingStateShadow "
    DrawingStateShadow =1
    IRet = DrawingStateShadow
    MsgBox "DrawingStateShadow = " & CStr(IRet), vbOkOnly, "Test DrawingStateShadow"
    DrawingStateShadow =2
    IRet = DrawingStateShadow
    MsgBox "DrawingStateShadow = " & CStr(IRet), vbOkOnly, "Test DrawingStateShadow"
    DrawingStateShadow =3
    IRet = DrawingStateShadow
    MsgBox "DrawingStateShadow = " & CStr(IRet), vbOkOnly, "Test DrawingStateShadow"
End Sub

```

EdgeColor, DrawCmdTarget Property

Syntax EdgeColor = Long

Description Questa proprietà imposta o restituisce il colore di contorno dell'oggetto. Il valore passato o restituito contiene il codice (R,G,B, su ogni byte) del colore di contorno. Può essere utile utilizzare la funzione RGB per identificare in modo più semplice il colore.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
    Dim IRet As Long
    EdgeColor = RGB(255,0,0)
    IRet = EdgeColor
    MsgBox "EdgeColor = " & CStr(IRet), vbOkOnly, "Test EdgeColor"
    EdgeColor = RGB(0,255,0)
    IRet = EdgeColor
    MsgBox "EdgeColor = " & CStr(IRet), vbOkOnly, "Test EdgeColor"
    EdgeColor = RGB(0,0,255)
    IRet = EdgeColor
    MsgBox "EdgeColor = " & CStr(IRet), vbOkOnly, "Test EdgeColor"
End Sub
```

EmbeddedSynoptic, DrawCmdTarget Property

Syntax EmbeddedSynoptic = _String

Description Questa proprietà imposta o restituisce il nome del sinottico visualizzato all'interno dell'oggetto disegno sinottico incastrato. Quindi è una proprietà valida solamente per questo tipo di oggetto nel sinottico e l'utilizzo di questa proprietà in altri tipi di disegno genererà un errore.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    MsgBox "EmbeddedSynoptic = " & EmbeddedSynoptic, vbOkOnly,
    GetProjectTitle
End Sub
```

EnableExecution, DrawCmdTarget Property

Syntax EnableExecution = _Boolean

Description Questa proprietà imposta o restituisce l'abilitazione alla gestione delle proprietà di esecuzione. Impostando questa proprietà a False, viene inibita l'elaborazione di tutte le funzionalità grafiche configurate. Viene inibita inoltre la notifica degli eventi basic generati dal cambio valore delle variabili.

Parameter	Description
-----------	-------------

None	None
------	------

Result Boolean

Example:

```
Public Sub Click()
    EnableExecution = Not EnableExecution
    MsgBox "EnableExecution = " & EnableExecution, vbOkOnly, GetProjectTitle
End Sub
```

EnableVariable, DrawCmdTarget Property

Syntax EnableVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabili di abilitazione dell'oggetto referenziato. Inserendo la variabile, il componente sarà abilitato quando il valore della variabile sarà diverso da zero.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim objRect As DrawCmdTarget
    Dim sVarName As String

    GetVariableNameFromList(sVarName)
    Set objRect = GetSynopticObject.GetSubObject("objRect")
    objRect.EnableVariable = sVarName

    Set objRect = Nothing
End Sub
```

ExtSettingsFile, DrawCmdTarget Property

Syntax ExtSettingsFile = _String

Description Questa proprietà imposta o restituisce il file di configurazione esterna per l'oggetto referenziato. Il file può essere specificato anche in fase di design nella proprietà "File di Configurazione" dell'oggetto. L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
Dim objSymbol As DrawCmdTarget
Set objSymbol = GetSynopticObject.GetSubObject("TestObject")
If objSymbol Is Nothing Then Exit Sub
objSymbol.ExtSettingsFile = "test.sxml"
objSymbol.SaveExtSettings
Set objSymbol= Nothing
End Sub
```

FillBrushPattern, DrawCmdTarget Property

Syntax FillBrushPattern = Integer

Description Questa proprietà imposta o restituisce il motivo di sfondo (retinatura) del riempimento graduale dell'oggetto nei casi in cui sia stata abilitata la proprietà di animazione "Riempimento graduale". Sono validi i valori compresi fra 0 e 22 che raggruppano la serie di motivi di sfondo (righe verticali, punti, mattoni, tessuto, ecc...) visibili negli attributi di riempimento Retinatura. Con il valore -1 (FFFF esadecimale) si elimina il motivo di sfondo applicato.



Questa proprietà non è pienamente supportata in Windows CE (ammette soltanto il valore -1).

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
Dim iRet As Integer
FillBrushPattern = 0
iRet = FillBrushPattern
MsgBox "FillBrushPattern = " & CStr(iRet), vbOkOnly, "Test FillBrushPattern"
FillBrushPattern = &HFFFF
iRet = FillBrushPattern
MsgBox "FillBrushPattern = " & CStr(iRet), vbOkOnly, "Test FillBrushPattern"
FillBrushPattern = 22
iRet = FillBrushPattern
MsgBox "FillBrushPattern = " & CStr(iRet), vbOkOnly, "Test FillBrushPattern"
End Sub
```

FillColor, DrawCmdTarget Property

Syntax FillColor = Long

Description Questa proprietà imposta o restituisce il colore di riempimento dell'oggetto nei casi in cui sia stata abilitata la proprietà di animazione "Riempimento graduale" e non siano stati specificati altri colori tramite la "Variabile colore" o l'editazione delle soglie. Il valore passato o restituito contiene il codice (R,G,B, su ogni byte) del colore di riempimento. Può essere utile utilizzare la funzione RGB per identificare in modo più semplice il colore.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
    Dim IRet As Long
    FillColor = RGB(255,0,0)
    IRet = FillColor
    MsgBox "FillColor = " & CStr(IRet), vbOkOnly, "Test FillColor"
    FillColor = RGB(0,255,0)
    IRet = FillColor
    MsgBox "FillColor = " & CStr(IRet), vbOkOnly, "Test FillColor"
    FillColor = RGB(0,0,255)
    IRet = FillColor
    MsgBox "FillColor = " & CStr(IRet), vbOkOnly, "Test FillColor"
End Sub
```

FillingMode, DrawCmdTarget Property

Syntax FillingMode = Integer

Description Questa proprietà imposta o restituisce il tipo (direzione) di riempimento graduale del disegno. Accetta un parametro Integer contenente il valore (1, 2, 3, 4) per la direzione del riempimento.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    Dim iRet As Integer
    FillingMode = 1
    iRet = FillingMode
    MsgBox "FillingMode = " & CStr(iRet), vbOkOnly, "Test FillingMode"
End Sub
```

FillingPercent, DrawCmdTarget Property

Syntax FillingPercent = Integer

Description Questa proprietà imposta o restituisce il valore in percentuale di riempimento dell'oggetto. Accetta un parametro Integer contenente il valore di riempimento dell'oggetto. Nota: il valore del riempimento in quanto espresso in percentuale deve essere compreso tra 0 e 100.

Parameter	Description
-----------	-------------

None	None
------	------

Result Integer

Example:

```
Public Sub Click()
    Dim iRet As Integer
    FillingPercent = 30
    iRet = FillingPercent
    MsgBox "FillingPercent = " & CStr(iRet), vbOkOnly, "Test FillingPercent"
    FillingPercent = 50
    iRet = FillingPercent
    MsgBox "FillingPercent = " & CStr(iRet), vbOkOnly, "Test FillingPercent"
    FillingPercent = 80
    iRet = FillingPercent
    MsgBox "FillingPercent = " & CStr(iRet), vbOkOnly, "Test FillingPercent"
End Sub
```

Font3D, DrawCmdTarget Property

Syntax Font3D = Integer

Description Questa proprietà imposta o restituisce l'impostazione dell'effetto tridimensionale del testo nell'oggetto.

Stile dell'effetto 3D:
0=nessuno
1=sollevato
2=inserito

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    Dim iRet As Integer
    font3D = 2
    iRet = font3D
    MsgBox "Font3D = " & CStr(iRet), vbOkOnly, "Test Font3D"
    font3D = 1
    iRet = font3D
    MsgBox "Font3D = " & CStr(iRet), vbOkOnly, "Test Font3D"
    font3D = 0
    iRet = font3D
    MsgBox "Font3D = " & CStr(iRet), vbOkOnly, "Test Font3D"
End Sub
```

FontBold, DrawCmdTarget Property

Syntax FontBold = Boolean

Description Questa proprietà imposta o restituisce la selezione dello stile del carattere di tipo Grassetto.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    FontBold = Not FontBold
    MsgBox "FontBold = " & CStr(FontBold ), vbOkOnly, GetProjectTitle
End Sub
```

FontCharSet, DrawCmdTarget Property

Syntax FontCharSet = _Integer

Description Questa proprietà permette di modificare il set di caratteri del font di un disegno tramite il codice basic. I Valori assumibili ripresi dal Microsoft Platform SDK sono riportati sotto.

Il CharSet può essere:

0 = ANSI_CHARSET
1 = DEFAULT_CHARSET
2 = SYMBOL_CHARSET
128 = SHIFTJIS_CHARSET
129 = HANGEUL_CHARSET
129 = HANGUL_CHARSET
134 = GB2312_CHARSET
136 = CHINESEBIG5_CHARSET
255 = OEM_CHARSET

solo per WinNT 4.0:

77 = MAC_CHARSET
130 = JOHAB_CHARSET
177 = HEBREW_CHARSET
178 = ARABIC_CHARSET
161 = GREEK_CHARSET
162 = TURKISH_CHARSET
163 = VIETNAMESE_CHARSET
186 = BALTIC_CHARSET
222 = THAI_CHARSET
238 = EASTEUROPE_CHARSET
204 = RUSSIAN_CHARSET

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    FontCharSet = 2
    Debug.Print "Varore di charset = " & CStr(FontCharSet)
End Sub
```

FontEscapement, DrawCmdTarget Property

Syntax FontEscapement = _Integer

Description Questa proprietà imposta o restituisce la direzione del font per il testo nell'oggetto. Sono validi i valori compresi fra 0 e 359 e rappresentano i gradi di rotazione del titolo all'interno del disegno. La proprietà è disponibile solamente in ambiente Windows 32/64 bit.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    If i < 359 Then  
        FontEscapement = FontEscapement + 10  
        sRet = FontEscapement  
        MsgBox "FontEscapement = " & sRet, vbOkOnly, GetProjectTitle  
    Else  
        i = 0  
    End If  
End Sub
```

FontHeight, DrawCmdTarget Property

Syntax FontHeight = Integer

Description Questa proprietà imposta o restituisce la dimensione in pixel del font (carattere) per il testo nell'oggetto. Accetta un parametro Integer.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim iRet As Integer  
    FontHeight = 10  
    iRet = FontHeight  
    MsgBox "FontHeight = " & CStr(iRet), vbOkOnly, "Test FontHeight"  
    FontHeight = 20  
    iRet = FontHeight  
    MsgBox "FontHeight = " & CStr(iRet), vbOkOnly, "Test FontHeight"  
End Sub
```

FontItalic, DrawCmdTarget Property

Syntax FontItalic = Boolean

Description Questa proprietà imposta o restituisce la selezione dello stile del carattere di tipo Corsivo.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    FontItalic = Not FontItalic  
    MsgBox "FontItalic = " & CStr(FontItalic), vbOkOnly, GetProjectTitle  
End Sub
```

FontName, DrawCmdTarget Property

Syntax FontName = String

Description Questa proprietà imposta o restituisce il nome del font per il testo nell'oggetto. Accetta un parametro String.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sRet As String  
    FontName = "Arial"  
    sRet = FontName  
    MsgBox "FontName = " & sRet, vbOkOnly, "Test FontName"  
    FontName = "Times New Roman"  
    sRet = FontName  
    MsgBox "FontName = " & sRet, vbOkOnly, "Test FontName"  
End Sub
```

GradientColor, DrawCmdTarget Property

Syntax GradientColor = _Long

Description Questa proprietà imposta o restituisce il colore di sfumatura dell'oggetto. Accetta un parametro Long contenente il codice (R,G,B, su ogni byte) per il colore dell'ombra. Può risultare utile l'utilizzo della funzione RGB di Movicon.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    GradientColor = RGB(255,255,255)  
    sRet = GradientColor  
    MsgBox "GradientColor = " & sRet, vbOkOnly, GetProjectTitle 'Return->  
    16777215  
End Sub
```

GradientFill, DrawCmdTarget Property

Syntax GradientFill = _Integer

Description Questa proprietà imposta o restituisce, mediante un codice numerico, la direzione di sfumatura dell'oggetto. Tale proprietà può assumere i seguenti valori:

- 0 = nessuno,
- 1 = da destra a sinistra,
- 2 = dal centro verso le estremità in orizzontale
- 3 = da sinistra a destra
- 4 = dal basso verso l'alto
- 5 = dal centro verso le estremità in verticale
- 6 = dall'alto verso il basso
- 7 = dal centro verso le estremità
- 8 = dall'angolo in alto a sinistra in diagonale
- 9 = dall'angolo in alto a destra in diagonale
- 10 = dall'angolo in basso a destra in diagonale
- 11 = dall'angolo in basso a sinistra in diagonale

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    For i = 0 To 11 Step 1  
        GradientFill = i  
        sRet = GradientFill  
        MsgBox "GradientFill = " & sRet, vbOkOnly, GetProjectTitle  
    Next i  
End Sub
```

Height, DrawCmdTarget Property

Syntax Height = _Long

Description Questa proprietà restituisce o imposta il valore assegnato alla dimensione in altezza dell'oggetto.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim IRet As Long  
    Height = Height + 10  
    IRet = Height  
    MsgBox "Height = " & CStr(IRet), vbOkOnly, GetProjectTitle  
End Sub
```

Hilite, DrawCmdTarget Property

Syntax Hilite = _Boolean

Description Questa proprietà provoca l'inversione dei colori di sfondo e contorno dell'oggetto. Questa proprietà può essere utilizzata per evidenziare il passaggio di un disegno in una certa zona del sinottico. Il valore true rende attiva l'inversione dei colori.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim bRet As Boolean  
    Hilite = Not Hilite  
    bRet = Hilite  
    MsgBox "Hilite = " & CStr(bRet), vbOkOnly, "Test Hilite "  
End Sub
```

LineArrowHeight, DrawCmdTarget Property

Syntax LineArrowHeight = _Integer

Description Questa proprietà imposta o restituisce la dimensione delle frecce visualizzate sull'oggetto disegno. La proprietà è valida solamente per gli oggetti linea e connettore, gli unici che possono contenere agli estremi la visualizzazione di frecce.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    LineArrowType = 1  
    LineArrowHeight = 5  
    IRet = LineArrowHeight  
    MsgBox "LineArrowHeight = " & CStr(IRet), vbOkOnly, "Test LineArrowHeight "  
    LineArrowHeight = 12  
    IRet = LineArrowHeight
```

```

MsgBox "LineArrowHeight = " & CStr(IRet), vbOkOnly, "Test LineArrowHeight "
LineArrowHeight = 30
IRet = LineArrowHeight
MsgBox "LineArrowHeight = " & CStr(IRet), vbOkOnly, "Test LineArrowHeight "
LineArrowHeight = 25
IRet = LineArrowHeight
MsgBox "LineArrowHeight = " & CStr(IRet), vbOkOnly, "Test LineArrowHeight "
End Sub

```

LineArrowType, DrawCmdTarget Property

Syntax LineArrowType = _Integer

Description Questa proprietà imposta o restituisce il tipo di freccia visualizzata nell'oggetto disegno. La proprietà è valida solamente per gli oggetti linea e connettore, gli unici che possono contenere agli estremi la visualizzazione di frecce. I valori ammessi sono compresi fra 0 e 3 che corrispondono alle seguenti impostazioni: 0=--, 1=<-- , 2=--> e 3=<-->

Parameter	Description
None	None

Result Integer

Example:

```

Public Sub Click()
LineArrowType =0
IRet = LineArrowType
MsgBox "LineArrowType = " & CStr(IRet), vbOkOnly, "Test LineArrowType "
LineArrowType =1
IRet = LineArrowType
MsgBox "LineArrowType = " & CStr(IRet), vbOkOnly, "Test LineArrowType "
LineArrowType =2
IRet = LineArrowType
MsgBox "LineArrowType = " & CStr(IRet), vbOkOnly, "Test LineArrowType "
LineArrowType =3
IRet = LineArrowType
MsgBox "LineArrowType = " & CStr(IRet), vbOkOnly, "Test LineArrowType "
End Sub

```

LineEndingX, DrawCmdTarget Property

Syntax LineEndingX = _Long

Description Questa funzione imposta o restituisce il valore del punto finale sull'asse X di un oggetto Linea. Accetta un valore Long. Occorre tenere presente che la definizione del punto di inizio o fine linea è funzione della direzione che si è tenuta con il mouse per tracciare la linea stessa.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim IRet As Long  
    LineEndingX = LineEndingX + 1  
    IRet = LineEndingX  
    MsgBox "LineEndingX = " & CStr(IRet), vbOkOnly, GetProjectTitle  
End Sub
```

LineEndingY, DrawCmdTarget Property

Syntax LineEndingY = _Long

Description Questa funzione imposta o restituisce il valore del punto finale sull'asse Y di un oggetto Linea. Accetta un valore Long. Occorre tenere presente che la definizione del punto di inizio o fine linea è funzione della direzione che si è tenuta con il mouse per tracciare la linea stessa.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim IRet As Long  
    LineEndingY = LineEndingY + 1  
    IRet = LineEndingY  
    MsgBox "LineEndingY = " & CStr(IRet), vbOkOnly, GetProjectTitle  
End Sub
```

LineStartingX, DrawCmdTarget Property

Syntax LineStartingX = Long

Description Questa funzione imposta o restituisce il valore del punto iniziale sull'asse X di un oggetto Linea. Accetta un valore Long. Occorre tenere presente che la definizione del punto di inizio o fine linea è funzione della direzione che si è tenuta con il mouse per tracciare la linea stessa.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim IRet As Long  
    LineStartingX = LineStartingX + 1  
    IRet = LineStartingX  
    MsgBox "LineStartingX = " & CStr(IRet), vbOkOnly, "Test LineStartingX"  
End Sub
```

LineStartingY, DrawCmdTarget Property

Syntax LineStartingY = _Long

Description Questa funzione imposta o restituisce il valore del punto iniziale sull'asse Y di un oggetto Linea. Accetta un valore Long. Occorre tenere presente che la definizione del punto di inizio o fine linea è funzione della direzione che si è tenuta con il mouse per tracciare la linea stessa.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim lRet As Long  
    LineStartingX = LineStartingY + 1  
    lRet = LineStartingY  
    MsgBox "LineStartingY = " & CStr(lRet), vbOkOnly, GetProjectTitle  
End Sub
```

LinkedTextFormat, DrawCmdTarget Property

Syntax LinkedTextFormat = _String

Description La proprietà di esecuzione di un disegno o di un simbolo chiamata 'Testo' è composta da due campi di inserimento: la variabile e il formato. Quest'ultimo segue la sintassi sotto riportata.

Visualizzazione intera

x
xx
xxx
xxxx
xxxxx

Visualizzazione decimale

x.x
x.xx
x.xxx
x.xxxx
x.xxxxx

Per le variabili di tipo stringa non è necessario settare alcun tipo di formattazione.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
```

```

VariableLinkedText = InputBox("Variable name:", "Change
variable",VariableLinkedText)
LinkedTextFormat = InputBox("Format:", "Change format
string",LinkedTextFormat)
End Sub

```

LinkedTextFormatVariable, DrawCmdTarget Property

Syntax LinkedTextFormatVariable = String

Description Questa proprietà restituisce o imposta il nome della variabile il cui valore verrà utilizzato per determinare il formato di visualizzazione della variabile associata all'animazione Testo.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim objRectangle As DrawCmdTarget
    Set objRectangle = GetSynopticObject.GetSubObject("Rect1")
    MsgBox "Rect1 Format Variable = " &
objRectangle.LinkedTextFormatVariable, vbInformation, GetProjectTitle
    Set objRectangle = Nothing
End Sub

```

Look3D, DrawCmdTarget Property

Syntax Look3D = Boolean

Description Questa proprietà imposta o restituisce la visualizzazione 3D del disegno.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre false)

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub Click()
    Look3D = Not Look3D
    MsgBox "Look3D = " & CStr(Look3D ), vbOkOnly, "Test Look3D "
End Sub

```

Look3DPressed, DrawCmdTarget Property

Syntax Look3DPressed = Boolean

Description Questa proprietà imposta o restituisce la visualizzazione premuto del disegno in cui è presente un Look3D.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre false)

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Look3D = True  
    Look3DPressed = Not Look3DPressed  
    MsgBox "Look3DPressed = " & CStr(Look3DPressed), vbOkOnly, "Test  
    Look3DPressed "  
End Sub
```

MetaFile, DrawCmdTarget Property

Syntax MetaFile = _String

Description Questa proprietà permette di leggere o impostare l'immagine da rappresentare sopra all'oggetto. La stringa passata o restituita corrispondente al nome del file immagine.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre una stringa vuota)

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Debug.Print MetaFile  
End Sub
```

MouseCapture, DrawCmdTarget Property

Syntax MouseCapture = Boolean

Description Questa proprietà abilita la notifica degli eventi legati al mouse anche se il cursore esce dal sinottico. Accetta un valore Boolean. Il valore di default per questa proprietà è False.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    MouseCapture = Not MouseCapture  
    MsgBox "MouseCapture = " & MouseCapture , vbOkOnly, GetProjectTitle  
End Sub
```

ObjectName, DrawCmdTarget Property

Syntax ObjectName

Description Questa proprietà restituisce (sola lettura) il nome assegnato all'oggetto tramite le proprietà generali dello stesso.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    MsgBox "ObjectName = " & ObjectName, vbOkOnly, "Test ObjectName"  
End Sub
```

ObjectPublicName, DrawCmdTarget Property

Syntax ObjectPublicName = _String

Description Questa proprietà imposta o restituisce il nome pubblico dell'oggetto referenziato. Modificando questa proprietà nel dropping code dell'oggetto il nome pubblico verrà salvato nel codice xml dell'oggetto e di conseguenza l'oggetto verrà aggiornato secondo le impostazioni del progetto. Durante la fase di runtime sarà possibile modificare il nome pubblico soltanto nell'evento "OnPreSymbolLoading" dell'oggetto. Questo significa che il simbolo verrà aggiornato soltanto se la proprietà viene impostata all'interno dell'evento "OnPreSymbolLoading". L'impostazione della proprietà al di fuori di questo evento non produrrà alcun effetto.

Parameter	Description
None	None

Result String

Example:

```
Public Sub OnPreSymbolLoading()  
    ObjectPublicName = "ReferenceSymbol"  
End Sub
```

PenColorBrightness, DrawCmdTarget Property

Syntax PenColorBrightness = _Integer

Description Questa proprietà imposta e restituisce la componente di luminosità (Brightness) associata al colore di linea/testo del disegno. Ogni colore è rappresentato da un valore long che può essere suddiviso in tre componenti intere: tonalità, saturazione e luminosità. I valori ammessi per questa proprietà vanno dallo 0 al 255.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim IRet As Integer  
    PenColorBrightness = 185  
    IRet = PenColorBrightness  
    MsgBox "PenColorBrightness = " & CStr(IRet), vbOkOnly, "Test  
    PenColorBrightness "  
    PenColorBrightness = 192  
    IRet = PenColorBrightness  
    MsgBox "PenColorBrightness = " & CStr(IRet), vbOkOnly, "Test  
    PenColorBrightness "  
    PenColorBrightness = 200  
    IRet = PenColorBrightness  
    MsgBox "PenColorBrightness = " & CStr(IRet), vbOkOnly, "Test  
    PenColorBrightness "  
End Sub
```

PenColorHue, DrawCmdTarget Property

Syntax PenColorHue = _Integer

Description Questa proprietà imposta e restituisce la componente di tonalità (Hue) associata al colore di linea/testo del disegno. Ogni colore è rappresentato da un valore long che può essere suddiviso in tre componenti intere: tonalità, saturazione e luminosità. I valori ammessi per questa proprietà vanno dallo 0 al 255.

Parameter	Description
-----------	-------------

None	None
------	------

Result Integer

Example:

```
Public Sub Click()
    Dim IRet As Integer
    PenColorHue = 185
    IRet = PenColorHue
    MsgBox "PenColorHue = " & CStr(IRet), vbOkOnly, "Test PenColorHue "
    PenColorHue = 192
    IRet = PenColorHue
    MsgBox "PenColorHue = " & CStr(IRet), vbOkOnly, "Test PenColorHue "
    PenColorHue = 200
    IRet = PenColorHue
    MsgBox "PenColorHue = " & CStr(IRet), vbOkOnly, "Test PenColorHue "
End Sub
```

PenColorSaturation, DrawCmdTarget Property

Syntax PenColorSaturation = _Integer

Description Questa proprietà imposta e restituisce la componente di saturazione (Saturation) associata al colore di linea/testo del disegno. Ogni colore è rappresentato da un valore long che può essere suddiviso in tre componenti intere: tonalità, saturazione e luminosità. I valori ammessi per questa proprietà vanno dallo 0 al 255.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    Dim IRet As Integer
    PenColorSaturation = 185
    IRet = PenColorSaturation
    MsgBox "PenColorSaturation = " & CStr(IRet), vbOkOnly, "Test PenColorSaturation "
    PenColorSaturation = 192
    IRet = PenColorSaturation
    MsgBox "PenColorSaturation = " & CStr(IRet), vbOkOnly, "Test PenColorSaturation "
    PenColorSaturation = 200
    IRet = PenColorSaturation
    MsgBox "PenColorSaturation = " & CStr(IRet), vbOkOnly, "Test PenColorSaturation "
End Sub
```

PenStyle, DrawCmdTarget Property

Syntax PenStyle = Integer

Description Imposta o restituisce lo stile della linea del bordo del disegno (continua, tratteggio, puntini). Affinchè questa proprietà sia attiva lo spessore della linea deve essere

uguale a 1, impostato tramite gli attributi di contorno Dimensione penna o con il metodo PenWidth. I valori impostabili sono da 0 a 4 e corrispondono alla lista di tratti disponibile negli attributi di contorno Stile penna.



Questa proprietà non è pienamente supportata in Windows CE (ammette soltanto il valore 0 e 1).

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim iRet As Integer  
    PenStyle = 1  
    iRet = PenStyle  
    MsgBox "PenStyle = " & CStr(iRet), vbOkOnly, "Test PenStyle"  
    PenStyle = 0  
    iRet = PenStyle  
    MsgBox "PenStyle = " & CStr(iRet), vbOkOnly, "Test PenStyle"  
    PenStyle = 4  
    iRet = PenStyle  
    MsgBox "PenStyle = " & CStr(iRet), vbOkOnly, "Test PenStyle"  
End Sub
```

PenVisible, DrawCmdTarget Property

Syntax Visible = Boolean

Description Questa proprietà imposta o restituisce la condizione logica di visibilità del disegno. Accetta un parametro di tipo Boolean.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim bRet As Boolean  
    Visible = False  
    bRet = Visible  
    MsgBox "Visible = " & CStr(bRet), vbOkOnly, "Test Visible"  
    Visible = True  
    bRet = Visible  
    MsgBox "Visible = " & CStr(bRet), vbOkOnly, "Test Visible"  
End Sub
```

PenWidth, DrawCmdTarget Property

Syntax PenWidth = Integer

Description Imposta o restituisce la dimensione in pixel della linea del bordo di un oggetto.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim iRet As Integer  
    PenWidth = 1  
    iRet = PenWidth  
    MsgBox "PenWidth = " & CStr(iRet), vbOkOnly, "Test PenWidth"  
    PenWidth = 5  
    iRet = PenWidth  
    MsgBox "PenWidth = " & CStr(iRet), vbOkOnly, "Test PenWidth"  
    PenWidth = 10  
    iRet = PenWidth  
    MsgBox "PenWidth = " & CStr(iRet), vbOkOnly, "Test PenWidth"  
End Sub
```

Rotation, DrawCmdTarget Property

Syntax Rotation = Integer

Description Questa proprietà restituisce o imposta il valore di rotazione espresso in gradi da 0 a 360 assegnato all'oggetto, definibile anche tramite le proprietà dello stesso. Accetta un parametro di tipo Integer.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre zero)

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim iRet As Integer  
    Rotation = 30  
    iRet = Rotation  
    MsgBox "Rotation = " & CStr(iRet), vbOkOnly, "Test Rotation"  
    Rotation = 45  
    iRet = Rotation  
    MsgBox "Rotation = " & CStr(iRet), vbOkOnly, "Test Rotation"  
    Rotation = 60  
    iRet = Rotation  
    MsgBox "Rotation = " & CStr(iRet), vbOkOnly, "Test Rotation"  
End Sub
```

Shadow, DrawCmdTarget Property

Syntax Shadow = _Boolean

Description Questa proprietà abilita o disabilita la visualizzazione dell'ombra di un oggetto.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Shadow = Not Shadow  
    sRet = Shadow  
    MsgBox "Shadow = " & sRet, vbOkOnly, GetProjectTitle  
End Sub
```

ShadowColor, DrawCmdTarget Property

Syntax ShadowColor = _Long

Description Questa proprietà imposta o restituisce il colore dell'ombra dell'oggetto. Accetta un parametro Long contenente il codice (R,G,B, su ogni byte) per il colore dell'ombra. Può risultare utile l'utilizzo della funzione RGB di Movicon.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    ShadowColor = RGB(255,255,255)  
    sRet = ShadowColor  
    MsgBox "ShadowColor = " & sRet, vbOkOnly, GetProjectTitle 'Return->  
    16777215  
End Sub
```

ShadowXOffset, DrawCmdTarget Property

Syntax ShadowXOffset = _Integer

Description Questa proprietà imposta o restituisce la posizione orizzontale dell'ombra riferita all'oggetto disegno sulla quale è visualizzata. Valori positivi indicano uno spostamento verso destra dell'ombra mentre valori negativi verso sinistra. L'ombra è visualizzata nella posizione impostata solamente se è abilitata.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    Shadow = True
    ShadowXOffset = 10
    IRet = ShadowXOffset
    MsgBox "ShadowXOffset = " & CStr(IRet), vbOkOnly, "Test ShadowXOffset "
End Sub
```

ShadowYOffset, DrawCmdTarget Property

Syntax ShadowYOffset = _Integer

Description Questa proprietà imposta o restituisce la posizione verticale dell'ombra riferita all'oggetto disegno sulla quale è visualizzata. Valori positivi indicano uno spostamento verso destra dell'ombra mentre valori negativi verso sinistra. L'ombra è visualizzata nella posizione impostata solamente se è abilitata.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    Shadow = True
    ShadowYOffset = 10
    IRet = ShadowYOffset
    MsgBox "ShadowYOffset = " & CStr(IRet), vbOkOnly, "Test ShadowYOffset "
End Sub
```

ShowFocusRectangle, DrawCmdTarget Property

Syntax ShowFocusRectangle = _Boolean

Description Questa proprietà imposta o restituisce la caratteristica di visualizzazione del focus dell'oggetto. Il focus è rappresentato da un contorno tratteggiato in prossimità degli estremi del disegno che appare nel caso in cui il disegno sia sensibile alla pressione del mouse o della tastiera.
Il valore true ne abilita la visualizzazione sul disegno.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim bRet As Boolean  
    ShowFocusRectangle = Not ShowFocusRectangle  
    bRet = ShowFocusRectangle  
    MsgBox "ShowFocusRectangle = " & CStr(bRet), vbOkOnly, "Test  
    ShowFocusRectangle "  
End Sub
```

ShowHiliteRectangle, DrawCmdTarget Property

Syntax ShowHiliteRectangle = _Boolean

Description Imposta o restituisce la proprietà di visualizzazione della presenza del mouse sopra all'oggetto disegno. Tale presenza è rappresentata da un bordo rialzato intorno agli estremi del disegno che appare nel caso in cui il disegno sia sensibile alla pressione del mouse o della tastiera.
Il valore true ne abilita la visualizzazione sul disegno.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim bRet As Boolean  
    ShowHiliteRectangle = Not ShowHiliteRectangle  
    bRet = ShowHiliteRectangle  
    MsgBox "ShowHiliteRectangle = " & CStr(bRet), vbOkOnly, "Test  
    ShowHiliteRectangle "  
End Sub
```

SynapseBackColor, DrawCmdTarget Property

Syntax SynapseBackColor(_IpszSynapseName)

Description Questa proprietà imposta o restituisce il colore di sfondo per la sinapsi definita nell'oggetto disegno, identificandone il nome attraverso il parametro IpszSynapseName. Le sinapsi sono rappresentate sul disegno da un pallino colorato e la posizione è definita durante l'introduzione della sinapsi.
Il valore passato o restituito contiene il codice (R,G,B, su ogni byte) del colore di sfondo. Può essere utile utilizzare la funzione RGB per identificare in modo più semplice il colore.

Parameter	Description
IpszSynapseName As String	Nome della sinapsi.

Result Long

Example:

```
Public Sub OnFireExecution()  
    Dim IColor As Long  
    If SynapseValue("OUT") = True Then  
        IColor = RGB(0, 255, 0) ' Green color (Colore verde)  
    Else  
        IColor = RGB(255, 0, 0) ' Red color (Colore rosso)  
    End If  
  
    ' Set color (Imposta il colore)  
    SynapseBackColor("OUT") = IColor  
End Sub
```

SynapseValue, DrawCmdTarget Property

Syntax SynapseValue(_IpszSynapseName)

Description Questa proprietà imposta o restituisce il valore di una sinapsi definita nell'oggetto disegno, identificandone il nome attraverso il parametro IpszSynapseName. Il valore impostato o restituito è di tipo variant e permette ad una sinapsi di essere compatibile con tutti i tipi di dati definiti dal linguaggio basic script.

Parameter	Description
IpszSynapseName As String	Nome della sinapsi.

Result Variant

Example:

```
Public Sub OnFireExecution()  
    Dim Result As Variant  
    ' Read the old value (Legge il valore vecchio)  
    Result = GetVariableValue("Temperature")  
    ' Set value (Imposta il valore)  
    SynapseValue("OUT") = Result  
End Sub
```

SynapseValueFromID, DrawCmdTarget Property

Syntax SynapseValueFromID(_nID)

Description Questa proprietà imposta o restituisce il valore di una sinapsi definita nell'oggetto disegno, identificandone il numero di introduzione attraverso il parametro nID. Il valore impostato o restituito è di tipo variant e permette ad una sinapsi di essere compatibile con tutti i tipi di dati definiti dal linguaggio basic script.

Parameter	Description
nID As Integer	ID della sinapsi.

Result Variant

Example:

```
Public Sub OnFireExecution()  
    Dim Result As Variant  
  
    ' Read the old value (Legge il valore vecchio)  
    Result = GetVariableValue("Temperature")  
    ' Set value (Imposta il valore)  
    SynapseValueFromID(0) = Result  
End Sub
```

SynapsisVisible, DrawCmdTarget Property

Syntax SynapsisVisible = Boolean

Description Questa proprietà imposta o restituisce lo stato di visibilità delle sinapsi inserite nel disegno.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    SynapsisVisible = Not SynapsisVisible  
    MsgBox "SynapsisVisible = " & CStr(SynapsisVisible), vbOkOnly, "Test  
    SynapsisVisible "  
End Sub
```

StatusVariable, DrawCmdTarget Property

Syntax StatusVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabili di stato dell'oggetto referenziato. Inserendo la variabile in base allo stato di qualità della variabile il disegno assumerà uno stato grafico differente, rendendo visivamente immediato capire se quella variabile ha qualità buona o no.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim objRect As DrawCmdTarget  
    Dim sVarName As String  
  
    GetVariableNameFromList(sVarName)
```

```

        Set objRect = GetSynopticObject.GetSubObject("objRect")
        objRect.StatusVariable = sVarName

        Set objRect = Nothing
End Sub

```

Title, DrawCmdTarget Property

Syntax Title = String

Description Questa proprietà restituisce o imposta il titolo assegnato al controllo, definibile anche tramite le proprietà generali dello stesso. Il titolo comparirà come testo statico all'interno dell'oggetto.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim sRet As String
    Title = "MyTitle"
    sRet = Title
    MsgBox "Title = " & sRet, vbOkOnly, "Test Title"
End Sub

```

ToolTip, DrawCmdTarget Property

Syntax ToolTip = _String

Description Questa proprietà imposta o restituisce la stringa di ToolTip relativa all'oggetto.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre una stringa vuota)

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim sRet As String
    ToolTip = "OggettoX"
    sRet = ToolTip
    MsgBox "ToolTip = " & sRet, vbOkOnly, "Test ToolTip"
End Sub

```

Transparency, DrawCmdTarget Property

Syntax Transparency = _Integer

Description Questa proprietà imposta o restituisce il livello di trasparenza di un oggetto. I valori ammessi vanno da 0 a 255 (0 = completamente invisibile, 255 = completamente visibile).



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre zero)

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Transparency = 255  
End Sub
```

VariableBackColor, DrawCmdTarget Property

Syntax VariableBackColor = String

Description Questa proprietà restituisce o imposta il nome della variabile associata alla proprietà del colore di sfondo dell'oggetto.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sRet As String  
    VariableBackColor = "VAR0001"  
    sRet = VariableBackColor  
    MsgBox "VariableBackColor = " & sRet, vbOkOnly, "Test VariableBackColor"  
End Sub
```

VariableBitmapList, DrawCmdTarget Property

Syntax VariableBitmapList = String

Description Questa proprietà restituisce o imposta il nome della variabile di riferimento per la proprietà di visualizzazione bitmap dell'oggetto.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sRet As String
    VariableBitmapList = "VAR0001"
    sRet = VariableBitmapList
    MsgBox "VariableBitmapList = " & sRet, vbOkOnly, GetProjectTitle
End Sub
```

VariableComposedMovement, DrawCmdTarget Property

Syntax VariableComposedMovement = _String

Description Questa proprietà restituisce o imposta il nome della variabile di riferimento per la proprietà di movimento composto dell'oggetto.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    MsgBox "VariableComposedMovement = " & VariableComposedMovement(),
    vbOkOnly, GetProjectTitle
End Sub
```

VariableEdgeColor, DrawCmdTarget Property

Syntax VariableEdgeColor = String

Description Questa proprietà restituisce o imposta il nome della variabile associata alla proprietà del colore di contorno e testo dell'oggetto.

Parameter	Description
lpar As String	Nome della variabile.

Result String

Example:

```
Public Sub Click()
    Dim sRet As String
```

```

VariableEdgeColor = "VAR0001"
sRet = VariableEdgeColor
MsgBox "VariableEdgeColor = " & sRet, vbOkOnly, GetProjectTitle
End Sub

```

VariableEndingX, DrawCmdTarget Property

Syntax VariableEndingX = String

Description Questa proprietà restituisce o imposta il nome della variabile di riferimento per la proprietà di posizione X del punto finale dell'oggetto.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim sRet As String
    VariableEndingX = "VAR0001"
    sRet = VariableEndingX
    MsgBox "VariableEndingX = " & sRet, vbOkOnly, GetProjectTitle
End Sub

```

VariableEndingY, DrawCmdTarget Property

Syntax VariableEndingY = String

Description Questa proprietà restituisce o imposta il nome della variabile di riferimento per la proprietà di posizione Y del punto finale dell'oggetto.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim sRet As String
    VariableEndingY = "VAR0001"
    sRet = VariableEndingY
    MsgBox "VariableEndingY = " & sRet, vbOkOnly, GetProjectTitle
End Sub

```

VariableFillColor, DrawCmdTarget Property

Syntax VariableFillColor = String

Description Questa proprietà restituisce o imposta il nome della variabile di riferimento per l'animazione del colore dell'oggetto.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sRet As String  
    VariableFillColor = "VAR0001"  
    sRet = VariableFillColor  
    MsgBox "VariableFillColor = " & sRet, vbOkOnly, GetProjectTitle  
End Sub
```

VariableFilling, DrawCmdTarget Property

Syntax VariableFilling = String

Description Questa proprietà restituisce o imposta il nome della variabile di riferimento per l'animazione del riempimento graduale dell'oggetto.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sRet As String  
    VariableFilling = "VAR0001"  
    sRet = VariableFilling  
    MsgBox "VariableFilling = " & sRet, vbOkOnly, GetProjectTitle  
End Sub
```

VariableLinkedText, DrawCmdTarget Property

Syntax VariableLinkedText = String

Description Questa proprietà restituisce o imposta il nome della variabile associata alla proprietà di testo dell'oggetto.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim sRet As String
    VariableLinkedText = "VAR0001"
    sRet = VariableLinkedText
    MsgBox "VariableLinkedText = " & sRet, vbOkOnly, GetProjectTitle
End Sub

```

VariableMoveX, DrawCmdTarget Property

Syntax VariableMoveX = String

Description Questa proprietà restituisce o imposta il nome della variabile di riferimento per la proprietà di movimento orizzontale (X) dell'oggetto.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim sRet As String
    VariableMoveX = "VAR0001"
    sRet = VariableMoveX
    MsgBox "VariableMoveX = " & sRet, vbOkOnly, GetProjectTitle
End Sub

```

VariableMoveY, DrawCmdTarget Property

Syntax VariableMoveY = String

Description Questa proprietà restituisce o imposta il nome della variabile di riferimento per la proprietà di movimento verticale (Y) dell'oggetto.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim sRet As String
    VariableMoveY = "VAR0001"
    sRet = VariableMoveY
    MsgBox "VariableMoveY = " & sRet, vbOkOnly, GetProjectTitle
End Sub

```

VariableRotation, DrawCmdTarget Property

Syntax VariableRotation = String

Description Questa proprietà restituisce o imposta il nome della variabile di riferimento per la proprietà di rotazione dell'oggetto.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre una stringa vuota)

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sRet As String  
    VariableRotation = "VAR0001"  
    sRet = VariableRotation  
    MsgBox "VariableRotation = " & sRet, vbOkOnly, GetProjectTitle  
End Sub
```

VariableScaling, DrawCmdTarget Property

Syntax VariableScaling = String

Description Questa proprietà restituisce o imposta il nome della variabile di riferimento per la proprietà di riduzione in scala dell'oggetto.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sRet As String  
    VariableScaling = "VAR0001"  
    sRet = VariableScaling  
    MsgBox "VariableScaling = " & sRet, vbOkOnly, GetProjectTitle  
End Sub
```

VariableStartingX, DrawCmdTarget Property

Syntax VariableStartingX = String

Description Questa proprietà restituisce o imposta il nome della variabile di riferimento per la proprietà di posizione X del punto iniziale dell'oggetto.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sRet As String
    VariableStartingX = "VAR0001"
    sRet = VariableStartingX
    MsgBox "VariableStartingX = " & sRet, vbOkOnly, GetProjectTitle
End Sub
```

VariableStartingY, DrawCmdTarget Property

Syntax VariableStartingY = String

Description Questa proprietà restituisce o imposta il nome della variabile di riferimento per la proprietà di posizione Y del punto iniziale dell'oggetto.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sRet As String
    VariableStartingY = "VAR0001"
    sRet = VariableStartingY
    MsgBox "VariableStartingY = " & sRet, vbOkOnly, GetProjectTitle
End Sub
```

VariableVisible, DrawCmdTarget Property

Syntax VariableVisible = String

Description Questa proprietà restituisce o imposta il nome della variabile associata alla proprietà di visibilità dell'oggetto.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sRet As String
```

```

VariableVisible = "VAR0001"
sRet = VariableVisible
MsgBox "VariableVisible = " & sRet, vbOkOnly, "Test VariableVisible "
End Sub

```

VariableXRotationCenter, DrawCmdTarget Property

Syntax VariableXRotationCenter = String

Description Questa proprietà restituisce o imposta il nome della variabile di riferimento per la proprietà di posizione X del punto del baricentro di rotazione dell'oggetto.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre una stringa vuota)

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim sRet As String
    VariableXRotationCenter = "VAR0001"
    sRet = VariableXRotationCenter
    MsgBox "VariableXRotationCenter = " & sRet, vbOkOnly, GetProjectTitle
End Sub

```

VariableYRotationCenter, DrawCmdTarget Property

Syntax VariableYRotationCenter = String

Description Questa proprietà restituisce o imposta il nome della variabile di riferimento per la proprietà di posizione Y del punto del baricentro di rotazione dell'oggetto.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre una stringa vuota)

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim sRet As String
    VariableYRotationCenter = "VAR0001"

```

```

sRet = VariableYRotationCenter
MsgBox "VariableYRotationCenter = " & sRet, vbOkOnly, GetProjectTitle
End Sub

```

Visible, DrawCmdTarget Property

Syntax Visible = Boolean

Description Imposta o restituisce la condizione di visibilità dell'oggetto.

Parameter	Description
None	None

Result Boolean

Example:
Public Sub Click()
 Visible = Not **Visible**
 MsgBox "Visible = " & CStr(**Visible**), vbOkOnly, "Test Visible"
End Sub

Width, DrawCmdTarget Property

Syntax Width = _Long

Description Questa proprietà restituisce o imposta il valore assegnato alla dimensione in larghezza dell'oggetto.

Parameter	Description
None	None

Result Long

Example:
Public Sub Click()
 Dim lRet As Long
 Width = **Width** + 10
 lRet = **Width**
 MsgBox "Width = " & CStr(lRet), vbOkOnly, GetProjectTitle
End Sub

Xpos, DrawCmdTarget Property

Syntax Xpos = Long

Description Questa proprietà esprime la coordinata X dello spigolo più a sinistra del componente. Il valore è espresso in pixel ed è relativo al punto zero della finestra sinottico (spigolo

in alto a sinistra) che contiene il componente. Il valore di questa proprietà verrà modificato automaticamente ogni volta il componente viene spostato sul sinottico, viceversa, modificando questo valore il componente adeguerà la sua posizione graficamente.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
    Dim IRet As Long
    Xpos = Xpos + 1
    IRet = Xpos
    MsgBox "Xpos = " & CStr(IRet), vbOkOnly, "Test Xpos"
End Sub
```

XRotationCenter, DrawCmdTarget Property

Syntax XRotationCenter = _Integer

Description Questa proprietà imposta o restituisce la coordinata X del baricentro di rotazione dell'oggetto.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre zero)

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    Dim iRet As Integer
    XRotationCenter = 10
    iRet = XRotationCenter
    MsgBox "XRotationCenter = " & CStr(iRet), vbOkOnly, "Test XRotationCenter"
End Sub
```

Ypos, DrawCmdTarget Property

Syntax Ypos = Long

Description Questa proprietà esprime la coordinata Y dello spigolo più in alto del componente. Il valore è espresso in pixel ed è relativo al punto zero della finestra sinottico (spigolo in alto a sinistra) che contiene il componente. Il valore di questa proprietà verrà modificato automaticamente ogni volta il componente viene spostato sul sinottico, viceversa, modificando questo valore il componente adeguerà la sua posizione graficamente.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
    Dim IRet As Long
    Ypos = Ypos + 1
    IRet = Ypos
    MsgBox "Ypos = " & CStr(IRet), vbOkOnly, "Test Ypos"
End Sub
```

YRotationCenter, DrawCmdTarget Property

Syntax YRotationCenter = _Integer

Description Questa proprietà imposta o restituisce la coordinata Y del baricentro di rotazione dell'oggetto.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre zero)

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    Dim iRet As Integer
    YRotationCenter = 10
    iRet = YRotationCenter
    MsgBox "YRotationCenter = " & CStr(iRet), vbOkOnly, "Test YRotationCenter "
End Sub
```

1.15.7. EventCmdTarget

Functions

GetXMLSettings, EventCmdTarget Function

Syntax GetXMLSettings()

Description Questa funzione restituisce una stringa con il contenuto XML del file di progetto relativo all'evento referenziato.

Parameter	Description
-----------	-------------

None	None
------	------

Result String

Example:

```
Public Sub AlarmLoading()
    Dim EventObj As EventCmdTarget
    Set EventObj = GetEvent("Event1")
    If Not EventObj Is Nothing Then
        MsgBox EventObj.GetXMLSettings ,vbOkOnly,""
    End If
    Set EventObj = Nothing
End Sub
```

Properties

CommandList, EventCmdTarget Property

Syntax CommandList = _String

Description Questa proprietà restituisce la stringa XML di progetto contenente la definizione dei comandi associati all'evento referenziato.

Parameter	Description
None	None

Result String

Example:

```
Public Sub AlarmLoading()
    Dim EventObj As EventCmdTarget
    Set EventObj = GetEvent("Event1")
    If Not EventObj Is Nothing Then
        MsgBox EventObj.CommandList,vbOkOnly,""
    End If
    Set EventObj = Nothing
End Sub
```

Condition, EventCmdTarget Property

Syntax Condition = _Integer

Description Questa proprietà restituisce imposta o restituisce la condizione per cui si verifica l'evento referenziato.

Le possibili configurazioni sono:

```
enum_evc_changed = cambiamento
enum_evc_equal = uguale
enum_evc_major = maggiore
enum_evc_minor = minore
```

Parameter	Description
-----------	-------------

None	None
------	------

Result Integer

Example:

```
Public Sub AlarmLoading()
    Dim EventObj As EventCmdTarget
    Set EventObj = GetEvent("Event1")
    If Not EventObj Is Nothing Then
        EventObj.Condition = enum_evc_equal
    End If
    Set EventObj = Nothing
End Sub
```

Enable, EventCmdTarget Property

Syntax Enable = _Boolean

Description Questa proprietà abilita o disabilita l'evento referenziato.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub AlarmLoading()
    Dim EventObj As EventCmdTarget
    Set EventObj = GetEvent("Event1")
    If Not EventObj Is Nothing Then
        EventObj.Enable = True
    End If
    Set EventObj = Nothing
End Sub
```

EnableVariable, EventCmdTarget Property

Syntax EnableVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile di abilitazione per l'evento referenziato.
Durante il Runtime, se il valore di questa variabile è uguale a "zero" il "Comando" dell'Oggetto Evento non verrà mai eseguito. Quando il suo valore è diverso da "zero" il "Comando" dell'Oggetto Evento sarà eseguito in base alle condizioni impostate.



*Questa funzionalità verrà gestita da Movicon soltanto se la proprietà **"Abilita"** è stata attivata. In caso contrario infatti l'Oggetto Evento non esegue mai nessun comando indipendentemente dallo stato della **"Variabile di Abilitazione"**.*

Parameter	Description
None	None

Result String

Example:

```
Public Sub AlarmLoading()  
    Dim EventObj As EventCmdTarget  
    Set EventObj = GetEvent("Event1")  
    If Not EventObj Is Nothing Then  
        EventObj.EnableVariable = "VAR00004"  
    End If  
    Set EventObj = Nothing  
End Sub
```

Name, EventCmdTarget Property

Syntax Name = _String

Description Questa funzione restituisce una stringa con il nome dell'oggetto evento referenziato.

Parameter	Description
None	None

Result String

Example:

```
Public Sub AlarmLoading()  
    Dim EventObj As EventCmdTarget  
    Set EventObj = GetEvent("Event1")  
    If Not EventObj Is Nothing Then  
        MsgBox "Event Name is " & EventObj.Name,vbOkOnly,""  
    End If  
    Set EventObj = Nothing  
End Sub
```

Value, EventCmdTarget Property

Syntax Value = _Double

Description Questa proprietà restituisce imposta o restituisce il valore di soglia al quale la proprietà "Condizione" fa riferimento. Questa impostazione non ha significato se come "Condizione" si è selezionata l'opzione "Cambiamento".

Parameter	Description
None	None

Result Double

Example:

```
Public Sub AlarmLoading()  
    Dim EventObj As EventCmdTarget  
    Set EventObj = GetEvent("Event1")  
    If Not EventObj Is Nothing Then  
        EventObj.Value = 12  
    End If
```



```
Set EventObj = Nothing  
End Sub
```

Variable, EventCmdTarget Property

Syntax Variable = _String

Description Questa proprietà imposta o restituisce il nome della variabile da monitorare. Al cambiamento del valore della stessa, se le impostazioni della proprietà "Condizione" lo richiedono, viene eseguito il "Comando" associato.

Parameter	Description
None	None

Result String

Example:

```
Public Sub AlarmLoading()  
    Dim EventObj As EventCmdTarget  
    Set EventObj = GetEvent("Event1")  
    If Not EventObj Is Nothing Then  
        EventObj.Variable = "VAR00005"  
    End If  
    Set EventObj = Nothing  
End Sub
```

1.15.8. GaugeCmdTarget

Functions

LoadExtSettings, GaugeCmdTarget Function

Syntax LoadExtSettings

Description Questa funzione permette di caricare la configurazione dell'oggetto dal file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objSymbol As GaugeCmdTarget  
    Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface  
    If objSymbol Is Nothing Then Exit Sub  
    objSymbol.ExtSettingsFile = "test.sxml"  
    objSymbol.LoadExtSettings  
    Set objSymbol = Nothing  
End Sub
```

SaveExtSettings, GaugeCmdTarget Function

Syntax SaveExtSettings

Description Questa funzione permette di salvare la configurazione dell'oggetto nel file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".XML".

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
Dim objSymbol As GaugeCmdTarget  
Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface  
If objSymbol Is Nothing Then Exit Sub  
objSymbol.ExtSettingsFile = "test.xml"  
objSymbol.SaveExtSettings  
Set objSymbol = Nothing  
End Sub
```

Properties

BarBackColor, GaugeCmdTarget Property

Syntax BarBackColor = _Long

Description Questa proprietà imposta o restituisce il colore da associare allo Sfondo della Barra del Gauge.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit  
Public Sub Click()  
    Dim MyGauge As GaugeCmdTarget  
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface  
    If Not MyGauge Is Nothing Then  
        MyGauge.BarBackColor = RGB(234,176,89)  
        Set MyGauge = Nothing  
    End If  
End Sub
```

BarBias, GaugeCmdTarget Property

Syntax BarBias = _Boolean

Description Questa proprietà permette di visualizzare o nascondere la "Barra Bias" associata alla variabile nella finestra del Gauge

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.BarBias = Not MyGauge.BarBias
        Set MyGauge = Nothing
    End If
End Sub
```

BarBorder, GaugeCmdTarget Property

Syntax BarBorder = _Integer

Description Questa proprietà imposta o restituisce il tipo di bordo grafico della barra del Gauge. I valori possibili vanno da 0 a 7. La proprietà ha significato solo quando il tipo di Gauge è verticale od orizzontale.

I valori si riferiscono a :

- 0 none
- 1 normal
- 2 medium
- 3 thick
- 4 bump
- 5 etched
- 6 raised
- 7 sunken

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.BarVisible = True
        MyGauge.BarBorder = i
    End If
End Sub
```

```

        i = i + 1
    If i = 8 Then
        i = 0
    End If
    Set MyGauge = Nothing
End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub

```

BarFillColor, GaugeCmdTarget Property

Syntax BarFillColor = _Long

Description Questa proprietà imposta o restituisce il colore da associare al riempimento della Barra del Gauge in base al valore della variabile associata.

Parameter	Description
None	None

Result Long

Example:

```

Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.BarFillColor = RGB(234,176,89)
        Set MyGauge = Nothing
    End If
End Sub

```

BarVisible, GaugeCmdTarget Property

Syntax BarVisible = _Boolean

Description Questa proprietà permette di visualizzare o nascondere la "Barra" di sfondo e il relativo riempimento nella finestra del Gauge.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.BarVisible = Not MyGauge.BarVisible
        Set MyGauge = Nothing
    End If
End Sub

```

```
End If  
End Sub
```

CenterPos, GaugeCmdTarget Property

Syntax CenterPos = _Integer

Description Questa proprietà imposta o restituisce il centro dell'Ago del Gauge, e di conseguenza tutto il semicerchio degli elementi.
I valori possibili sono:

0	alto a sinistra
1	alto al centro
2	alto a destra
3	in centro a sinistra
4	in centro
5	in centro a destra
6	in basso a sinistra
7	in basso al centro
8	in basso a destra

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit  
Dim i As Integer  
Public Sub Click()  
    Dim MyGauge As GaugeCmdTarget  
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface  
    If Not MyGauge Is Nothing Then  
        MyGauge.BarVisible=True  
        MyGauge.CenterPos = i  
        i = i + 1  
        If i = 9 Then  
            i = 0  
        End If  
        Set MyGauge = Nothing  
    End If  
End Sub  
Public Sub SymbolLoading()  
    i = 0  
End Sub
```

ColorWarningZone, GaugeCmdTarget Property

Syntax ColorWarningZone(_nZone) = _Long

Description Questa proprietà imposta o restituisce il colore da associare alla Zona Allarme del Gauge referenziata dall'indice. Le zone sono 5 e gli indici relativi vanno da zero a 4.

Parameter	Description
-----------	-------------

nZone As Integer	Indice della zona referenziata
------------------	--------------------------------

Result Long

Example:

```
Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.ColorWarningZone(0) = RGB(234,176,89)
        Set MyGauge = Nothing
    End If
End Sub
```

EnableWarningZone, GaugeCmdTarget Property

Syntax EnableWarningZone(_nZone) = _Boolean

Description Questa proprietà imposta o restituisce la visibilità della Zona Allarme del Gauge referenziata dall'indice. Le zone sono 5 e gli indici relativi vanno da zero a 4.

Parameter	Description
nZone As Integer	Indice della zona referenziata

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.EnableWarningZone(0) = True
        Set MyGauge = Nothing
    End If
End Sub
```

EndAngle, GaugeCmdTarget Property

Syntax EndAngle = _Integer

Description Questa proprietà imposta o restituisce il valore dell'angolo relativo alla fine dell'arco di cerchio che rappresenta gli elementi del Gauge. Questa proprietà ha significato solo se il gauge è di tipo circolare.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.BarVisible = True
        MyGauge.EndAngle = i
        i = i + 1
        If i = 360 Then
            i = 0
        End If
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub
```

EndWarningZone, GaugeCmdTarget Property

Syntax EndWarningZone(_nZone) = _Integer

Description Questa proprietà imposta o restituisce il valore finale della Zona Allarme del Gauge referenziata dall'indice. Le zone sono 5 e gli indici relativi vanno da zero a 4.

Parameter	Description
nZone As Integer	Indice della zona referenziata

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.EndWarningZone(0) = 50
        Set MyGauge = Nothing
    End If
End Sub
```

ExtSettingsFile, GaugeCmdTarget Property

Syntax ExtSettingsFile = _String

Description Questa proprietà imposta o restituisce il file di configurazione esterna per l'oggetto referenziato. Il file può essere specificato anche in fase di design nella proprietà "File di Configurazione" dell'oggetto. L'estensione prevista per il file è ".SXML".

Parameter	Description
-----------	-------------

None	None
------	------

Result Long

Example:

```
Public Sub Click()
Dim objSymbol As GaugeCmdTarget
Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
If objSymbol Is Nothing Then Exit Sub
objSymbol.ExtSettingsFile = "test.sxml"
objSymbol.SaveExtSettings
Set objSymbol= Nothing
End Sub
```

FontHeightScale, GaugeCmdTarget Property

Syntax FontHeightScale = _Integer

Description Questa proprietà imposta o restituisce la dimensione del carattere utilizzato nella scala del Gauge.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
Dim MyGauge As GaugeCmdTarget
Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
If Not MyGauge Is Nothing Then
MyGauge.FontNameScale = "Arial"
MyGauge.FontHeightScale = 20
Set MyGauge = Nothing
End If
End Sub
```

FontNameScale, GaugeCmdTarget Property

Syntax FontNameScale = _String

Description Questa proprietà imposta o restituisce il carattere utilizzato nella scala del Gauge.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
```



```

Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.FontNameScale = "Arial"
        MyGauge.FontHeightScale = 20
        Set MyGauge = Nothing
    End If
End Sub

```

FormatVariable, GaugeCmdTarget Property

Syntax FormatVariable = _String

Description Questa proprietà restituisce o imposta il nome della variabile il cui valore verrà utilizzato per determinare il formato di visualizzazione della scala del potenziometro.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim objGauge As GaugeCmdTarget
    Set objGauge = GetSynopticObject.GetSubObject("Gauge1").GetObjectInterface
    MsgBox "Gauge1 Format Variable = " & objGauge.FormatVariable,
    vbInformation,GetProjectTitle
    Set objGauge = Nothing
End Sub

```

GaugeMaxVariable, GaugeCmdTarget Property

Syntax GaugeMaxVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile associata alla massima soglia rappresentata nel Gauge.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.GaugeMaxVariable = "VAR00001"
        VAR00001 = 90
        Set MyGauge = Nothing
    End If
End Sub

```

End If
End Sub

GaugeMinVariable, GaugeCmdTarget Property

Syntax GaugeMinVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile associata alla massima soglia rappresentata nel Gauge.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.GaugeMinVariable = "VAR00002"
        VAR00002 = 10
        Set MyGauge = Nothing
    End If
End Sub
```

GaugeType, GaugeCmdTarget Property

Syntax GaugeType = _Integer

Description Questa proprietà imposta o restituisce il tipo di Gauge da rappresentare.

I valori possibili sono:

- 0 verticale
- 1 orizzontale
- 2 circolare

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.GaugeType = i
    End If
End Sub
```

```

        i = i + 1
    If i = 3 Then
        i = 0
    End If
    Set MyGauge = Nothing
End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub

```

GaugeVariable, GaugeCmdTarget Property

Syntax GaugeVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile rappresentata nel Gauge.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.GaugeVariable = "VAR00004"
        Set MyGauge = Nothing
    End If
End Sub

```

GaugeWarningZoneEndVariable, GaugeCmdTarget Property

Syntax GaugeWarningZoneEndVariable(_nZone) = _String

Description Questa proprietà imposta o restituisce il nome della variabile che gestisce il valore finale della Zona Allarme del Potenzimetro. Il parametro "nZone" indica la zona di riferimento. Le zone sono 5 e gli indici relativi vanno da 0 a 4.

Parameter	Description
nZone As Integer	Valore che indica il numero di zona allarme di riferimento.

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface

```

```

        If Not MyGauge Is Nothing Then
            MyGauge.GaugeWarningZoneEndVariable(0) = "VarStartZone1"
            Set MyGauge = Nothing
        End If
    End Sub

```

GaugeWarningZoneStartVariable, GaugeCmdTarget Property

Syntax GaugeWarningZoneStartVariable(_nZone) = _String

Description Questa proprietà imposta o restituisce il nome della variabile che gestisce il valore iniziale della Zona Allarme del Potenzimetro. Il parametro "nZone" indica la zona di riferimento. Le zone sono 5 e gli indici relativi vanno da 0 a 4.

Parameter	Description
nZone As Integer	Valore che indica il numero di zona allarme di riferimento.

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.GaugeWarningZoneStartVariable(0) = "VarStartZone1"
        Set MyGauge = Nothing
    End If
End Sub

```

GeneralGap, GaugeCmdTarget Property

Syntax GeneralGap = _Integer

Description Questa proprietà imposta o restituisce il valore del "gap" tra le varie parti grafiche (bordo finestra/warning zone/bar) dell'oggetto Gauge.

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.GeneralGap = i
        i = i + 1
        If i = 5 Then

```

```

        i = 0
    End If
    Set MyGauge = Nothing
End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub

```

InvertScale, GaugeCmdTarget Property

Syntax InvertScale = _Boolean

Description Questa proprietà abilita o disabilita l'inversione dei valori di scala del Potenzimetro.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.InvertScale = Not MyGauge.InvertScale
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub

```

KnobBorder, GaugeCmdTarget Property

Syntax KnobBorder = _Integer

Description Questa proprietà imposta o restituisce il tipo di bordo grafico del cursore del Gauge. I valori possibili vanno da 0 a 7. La proprietà ha significato solo quando il tipo di Gauge è verticale od orizzontale.

```

0 none
1 normal
2 medium
3 thick
4 bump
5 etched
6 raised
7 sunken

```

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.BarVisible = True
        MyGauge.KnobBorder = i
        i = i + 1
        If i = 8 Then
            i = 0
        End If
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub
```

KnobColor, GaugeCmdTarget Property

Syntax KnobColor = _Long

Description Questa proprietà imposta o restituisce il colore del cursore del Gauge.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.BarVisible = True
        MyGauge.KnobColor = RGB(125,7,90)
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub
```

LabelEvery, GaugeCmdTarget Property

Syntax LabelEvery = _Integer

Description Questa proprietà imposta o restituisce ogni quante divisioni maggiori della "Scala" si deve visualizzare un'etichetta numerica che identifica il valore in quella posizione.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.BarVisible = True
        MyGauge.LabelEvery = 5
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub
```

MaxValue, GaugeCmdTarget Property

Syntax MaxValue = _Integer

Description Questa proprietà imposta o restituisce il valore massimo da rappresentare nel Gauge. Se la proprietà Var. Limite Max non è nulla, tale valore viene ignorato e viene considerato il valore della variabile stessa come fondo scala.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.BarVisible = True
        MyGauge.MaxValue = 120
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub
```

MinValue, GaugeCmdTarget Property

Syntax MinValue = _Integer

Description Questa proprietà imposta o restituisce il valore massimo da rappresentare nel Gauge. Se la proprietà Var. Limite Min non è nulla, tale valore viene ignorato e viene considerato il valore della variabile stessa come inizio scala.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.BarVisible = True
        MyGauge.MinValue = 10
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub
```

NeedleBorder, GaugeCmdTarget Property

Syntax NeedleBorder = _Integer

Description Questa proprietà imposta o restituisce il tipo di bordo grafico dell'ago del Gauge. I valori possibili vanno da 0 a 7. La proprietà ha significato solo quando il tipo di Gauge è circolare.

- 0 none
- 1 normal
- 2 medium
- 3 thick
- 4 bump
- 5 etched
- 6 raised
- 7 sunken

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.BarVisible = True
        MyGauge.NeedleBorder = i
        i = i + 1
    End If
End Sub
```



```

        If i = 8 Then
            i = 0
        End If
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub

```

NeedleBorderColor, GaugeCmdTarget Property

Syntax NeedleBorderColor = _Long

Description Questa proprietà imposta o restituisce il colore del bordo dell'ago del Gauge. La proprietà ha significato solo nel caso di Gauge circolare.

Parameter	Description
None	None

Result Long

Example:

```

Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.NeedleBorderColor = RGB(125,34,78)
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub

```

NeedleColor, GaugeCmdTarget Property

Syntax NeedleColor = _Long

Description Questa proprietà imposta o restituisce il colore dell'ago del Gauge. La proprietà ha significato solo nel caso di Gauge circolare.

Parameter	Description
None	None

Result Long

Example:

```

Option Explicit
Dim i As Integer

```

```

Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.NeedleColor = RGB(125,34,78)
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub

```

NeedleLength, GaugeCmdTarget Property

Syntax NeedleLength = _Integer

Description Questa proprietà imposta o restituisce la lunghezza dell'ago del Gauge. La proprietà ha significato solo nel caso di Gauge circolare.

I valori possibili sono:

```

0 small
1 medium
2 large

```

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.NeedleLength = RGB(125,34,78)
        i = i + 1
        If i = 3 Then
            i = 0
        End If
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub

```

NeedleShadow, GaugeCmdTarget Property

Syntax NeedleShadow = _Boolean

Description Questa proprietà abilita o disabilita la presenza dell'ombra dell'ago del Gauge. La proprietà ha significato solo nel caso di Gauge circolare.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.NeedleShadow = Not MyGauge.NeedleShadow
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub
```

NeedleShadowColor, GaugeCmdTarget Property

Syntax NeedleShadowColor = _Long

Description Questa proprietà imposta o restituisce il colore dell'ombra dell'ago del Gauge. La proprietà ha significato solo nel caso di Gauge circolare.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.NeedleShadowColor = RGB(125,34,78)
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub
```

NeedleVisible, GaugeCmdTarget Property

Syntax NeedleVisible = _Boolean

Description Questa proprietà abilita o disabilita la presenza della borchia dell'ago del Gauge e della sua ombra. La proprietà ha significato solo nel caso di Gauge circolare.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.NeedleVisible = Not MyGauge.NeedleVisible
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub
```

NeedleWidth, GaugeCmdTarget Property

Syntax NeedleWidth = _Integer

Description Questa proprietà imposta o restituisce la larghezza dell'ago del Gauge. La proprietà ha significato solo nel caso di Gauge circolare.

I valori possibili sono:

- 0 small
- 1 medium
- 2 large

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.NeedleWidth = RGB(125,34,78)
        i = i + 1
        If i = 3 Then
            i = 0
        End If
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
```

```
        i = 0
End Sub
```

ScaleColor, GaugeCmdTarget Property

Syntax ScaleColor = _Long

Description Questa proprietà imposta o restituisce il colore della scala del Gauge.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.ScaleColor = RGB(125,34,78)
        Set MyGauge = Nothing
    End If
End Sub
```

ScaleFormat, GaugeCmdTarget Property

Syntax ScaleFormat = _String

Description Questa proprietà imposta o restituisce il formato dei valori rappresentati sulla scala del Gauge.

I valori possibili sono:

x	Es: 1
xx	Es: 01
xxx	Es: 001
xxxx	Es: 0001
xxxxx	Es: 00001
x.x	Es: 1.0
x.xx	Es: 1.00
x.xxx	Es: 1.000
x.xxxx	Es: 1.0000
x.xxxxx	Es: 1.00000

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        Debug.Print MyGauge.ScaleFormat
        Set MyGauge = Nothing
    End If
End Sub

```

ScaleMajorDiv, GaugeCmdTarget Property

Syntax ScaleMajorDiv = _Integer

Description Questa proprietà imposta o restituisce il numero di divisioni principali da visualizzare sulla "Scala".

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.ScaleMajorDiv = 6
        Set MyGauge = Nothing
    End If
End Sub

```

ScaleMinorDiv, GaugeCmdTarget Property

Syntax ScaleMinorDiv = _Integer

Description Questa proprietà imposta o restituisce il numero di divisioni secondarie da visualizzare sulla "Scala". Le divisioni secondarie sono le divisioni comprese tra due divisioni primarie.

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.ScaleMinorDiv = 6
        Set MyGauge = Nothing
    End If
End Sub

```

End If
End Sub

ScaleRightBottom, GaugeCmdTarget Property

Syntax ScaleRightBottom = _Boolean

Description Questa proprietà, se impostata al valore booleano True, permette di visualizzare la Scala sulla Destra della finestra Gauge se questo è di tipo orizzontale o in basso alla finestra Gauge se questo è di tipo verticale.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.ScaleRightBottom = Not MyGauge.ScaleRightBottom
        Set MyGauge = Nothing
    End If
End Sub
```

ScaleUnit, GaugeCmdTarget Property

Syntax ScaleUnit = _String

Description Questa proprietà imposta o restituisce l'unità di misura sulla scala del Gauge.

I valori possibili sono:

x	Es: 1
xx	Es: 01
xxx	Es: 001
xxxx	Es: 0001
xxxxx	Es: 00001
x.x	Es: 1.0
x.xx	Es: 1.00
x.xxx	Es: 1.000
x.xxxx	Es: 1.0000
x.xxxxx	Es: 1.00000

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.ScaleUnit = "Hz"
        Set MyGauge = Nothing
    End If
End Sub

```

ScaleVisible, GaugeCmdTarget Property

Syntax ScaleVisible = _Boolean

Description Questa proprietà abilita o disabilita la presenza della scala nel Gauge.

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.ScaleVisible = Not MyGauge.ScaleVisible
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub

```

SliderBorder, GaugeCmdTarget Property

Syntax SliderBorder = _Integer

Description Questa proprietà imposta o restituisce il tipo di bordo grafico dello slider del Gauge. I valori possibili vanno da 0 a 7. La proprietà ha significato solo quando il tipo di Gauge è verticale od orizzontale.

- 0 none
- 1 normal
- 2 medium
- 3 thick
- 4 bump
- 5 etched
- 6 raised
- 7 sunken

Parameter	Description
-----------	-------------

None	None
------	------

Result Integer

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.BarVisible = True
        MyGauge.SliderBorder = i
        i = i + 1
        If i = 8 Then
            i = 0
        End If
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub
```

SliderColor, GaugeCmdTarget Property

Syntax SliderColor = _Long

Description Questa proprietà imposta o restituisce il colore dello slider del Gauge.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.BarVisible = True
        MyGauge.SliderColor = RGB(125,7,90)
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub
```

SliderVisible, GaugeCmdTarget Property

Syntax SliderVisible = _Boolean

Description Questa proprietà visualizza o nasconde lo slider del Gauge.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.BarVisible = True
        MyGauge.SliderVisible = Not MyGauge.SliderVisible
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub
```

StartAngle, GaugeCmdTarget Property

Syntax StartAngle = _Integer

Description Questa proprietà imposta o restituisce il valore dell'angolo relativo all'inizio dell'arco di cerchio che rappresenta gli elementi del Gauge. Questa proprietà ha significato solo se il gauge è di tipo circolare.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim i As Integer
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.BarVisible = True
        MyGauge.StartAngle = i
        i = i + 1
        If i = 360 Then
            i = 0
        End If
        Set MyGauge = Nothing
    End If
End Sub
Public Sub SymbolLoading()
    i = 0
End Sub
```

StartWarningZone, GaugeCmdTarget Property

Syntax StartWarningZone(_nZone) = _Integer

Description Questa proprietà imposta o restituisce il valore iniziale della Zona Allarme del Gauge referenziata dall'indice. Le zone sono 5 e gli indici relativi vanno da zero a 4.

Parameter	Description
nZone As Integer	Indice della zona referenziata

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.StartWarningZone(0) = 10
        Set MyGauge = Nothing
    End If
End Sub
```

Title, GaugeCmdTarget Property

Syntax Title = _String

Description Questa proprietà imposta o restituisce il titolo visualizzato nella finestra del Gauge.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.Title = "Temperature"
        Set MyGauge = Nothing
    End If
End Sub
```

TitleVisible, GaugeCmdTarget Property

Syntax Title = _Boolean

Description Questa proprietà imposta o restituisce la visibilità del titolo nella finestra del Gauge.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim MyGauge As GaugeCmdTarget
    Set MyGauge = GetSynopticObject.GetSubObject("MyGauge").GetObjectInterface
    If Not MyGauge Is Nothing Then
        MyGauge.TitleVisible= Not MyGauge.TitleVisible
        Set MyGauge = Nothing
    End If
End Sub
```

1.15.9. Generic Events

Click, Generic Event

Description Evento notificato quando viene premuto il tasto sinistro o destro del mouse all'interno dell'area del disegno.

Parameter	Description
Nessuno	Nessuno

DbClick, Generic Event

Description Evento generato quando viene premuto per due volte in rapida successione il tasto sinistro o destro del mouse all'interno dell'area del disegno. I tempi del doppio click sono stabiliti nelle impostazioni del sistema operativo.

Parameter	Description
Nessuno	Nessuno

KeyDown, Generic Event

Description Evento notificato quando viene premuto un tasto della tastiera. L'evento restituisce le variabili, di tipo integer, KeyCode e Shift. L'evento viene generato indipendentemente dal focus.

Parameter	Description
KeyCode As Integer	Codice VBA del tasto premuto. Il codice VBA e' un insieme di costanti che, oltre ai normali caratteri alfanumeri, senza pero' distinzione di

	maiuscolo o minuscolo, contempla anche gli altri tasti della tastiera come, ad esempio i tasti funzione, il Caps Lock, ecc.
Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT

KeyPress, Generic Event

Description Evento notificato quando viene premuto e rilasciato un tasto della tastiera. L'evento restituisce la variabile di tipo integer KeyAscii contenente il codice ASCII del tasto premuto. L'evento viene generato solo se il focus è fissato sul disegno.

Parameter	Description
Keyascii As Integer	Codice ASCII del tasto premuto

KeyUp, Generic Event

Description Evento generato quando viene rilasciato un tasto (dopo essere stato premuto) della tastiera. L'evento restituisce le variabili di tipo integer, KeyCode e Shift. L'evento viene generato indipendentemente dal focus.

Parameter	Description
KeyCode As Integer	Codice VBA del tasto premuto. Il codice VBA è un insieme di costanti che, oltre ai normali caratteri alfanumerici, senza però distinzione di maiuscolo o minuscolo, contempla anche gli altri tasti della tastiera come, ad esempio i tasti funzione, il Caps Lock, ecc.
Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT

KillFocus, Generic Event

Description Evento generato quando l'oggetto in questione viene deselezionato oppure perde il focus.

Parameter	Description
None	None

MouseDown, Generic Event

Description Evento notificato sia nel codice del sinottico sia nel codice degli oggetti quando viene premuto un qualsiasi pulsante del mouse nel sinottico, indipendentemente dalla posizione o dal simbolo. L'evento restituisce le variabili, di tipo integer, Button e Shift e le variabili, di tipo single, X e Y. Per poter gestire questo evento soltanto all'interno di un oggetto del sinottico è necessario utilizzare la funzione "IsCursorOnObject".

Parameter	Description
Button As Integer	Indice del tasto mouse premuto: 1 = Sinistro 2 = Destro 4 = Centrale
Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT
X As Single	Coordinata orizzontale riferita alla posizione del cursore al momento dell'evento
Y As Single	Coordinata verticale riferita alla posizione del cursore al momento dell'evento

MouseMove, Generic Event

Description Evento notificato sia nel codice del sinottico sia nel codice degli oggetti quando il cursore del mouse cambia la propria posizione nel sinottico, indipendentemente dalla posizione o dal simbolo. L'evento restituisce le variabili, di tipo integer, Button e Shift e le variabili, di tipo single, X e Y.
Per poter gestire questo evento soltanto all'interno di un oggetto del sinottico è necessario utilizzare la funzione "IsCursorOnObject".

Parameter	Description
Button As Integer	Indice del tasto mouse premuto: 1 = Sinistro 2 = Destro 4 = Centrale
Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT
X As Single	Coordinata orizzontale riferita alla posizione del cursore al momento dell'evento
Y As Single	Coordinata verticale riferita alla posizione del cursore al momento dell'evento

MouseUp, Generic Event

Description Evento notificato sia nel codice del sinottico sia nel codice degli oggetti quando viene rilasciato un qualsiasi pulsante del mouse nel sinottico, indipendentemente dalla posizione o dal simbolo. L'evento restituisce le variabili, di tipo integer, Button e Shift e le variabili, di tipo single, X e Y.
Per poter gestire questo evento soltanto all'interno di un oggetto del sinottico è necessario utilizzare la funzione "IsCursorOnObject".

Parameter	Description
Button As Integer	Indice del tasto mouse premuto: 1 = Sinistro 2 = Destro 4 = Centrale
Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT
X As Single	Coordinata orizzontale riferita alla posizione del cursore al momento dell'evento

Y As Single	Coordinata verticale riferita alla posizione del cursore al momento dell'evento
-------------	---

MouseWheel, Generic Events

Description Evento notificato quando si esegue lo scroll utilizzando la rotella del mouse. L'evento sarà disponibile solo per gli oggetti che supportano questa funzionalità e verrà eseguito se l'oggetto in questione ha il focus.
Per maggiori informazioni vedere anche il paragrafo "Scroll del Mouse per gli Oggetti del Sinottico".

Parameter	Description
Button As Integer	Indice del tasto mouse premuto: 1 = Sinistro 2 = Destro 4 = Centrale
Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT
Delta As Integer	Indica in che senso si sta ruotando la rotella del mouse: se il valore è maggiore di 0 lo scroll è verso l'alto mentre se il valore è minore di 0 lo scroll è verso il basso
X As Single	Coordinata orizzontale riferita alla posizione del cursore al momento dell'evento
Y As Single	Coordinata verticale riferita alla posizione del cursore al momento dell'evento

OnChange, Generic Event

Description Evento notificato quando l'oggetto disegno modifica il proprio stato grafico. L'evento restituisce un indice relativo a quale evento grafico è cambiato.

Gli indici sono riportati di seguito:
enum_ONCHANGE_COMPOSEDMOVE
enum_ONCHANGE_SCALE
enum_ONCHANGE_MOVEX
enum_ONCHANGE_MOVEY
enum_ONCHANGE_TITLE
enum_ONCHANGE_STARTINGX
enum_ONCHANGE_STARTINGY
enum_ONCHANGE_ENDINGX
enum_ONCHANGE_ENDINGY
enum_ONCHANGE_FILLING
enum_ONCHANGE_ROTATION
enum_ONCHANGE_VISIBLE
enum_ONCHANGE_EDGECOLOR
enum_ONCHANGE_BACKCOLOR
enum_ONCHANGE_FILLCOLOR
enum_ONCHANGE_BITMAP
enum_ONCHANGE_XROTATIONCENTER
enum_ONCHANGE_YROTATIONCENTER

Parameter	Description
ChangeType As Integer	Indice dell'evento grafico.

OnChangeExecutionCanceled, Generic Event

Description Evento notificato per un oggetto disegno che cerca di ottenere o di spostare l'esecuzione attuale della sinapsi.

Parameter	Description
None	None

OnChangeExecutionToPromoter, Generic Event

Description Evento notificato per l'oggetto disegno che ha in corso l'esecuzione della sinapsi alla quale viene notificato che un altro oggetto sta tentando di cambiare il flusso di esecuzione. Impostando il parametro bRet a False si impedisce che questo avvenga.

Parameter	Description
None	None

OnExecutionPending, Generic Event

Description Evento notificato una volta al secondo dall'oggetto disegno che ha in corso l'esecuzione della sinapsi indicando che il sistema è in attesa che questo passi l'esecuzione ad un altro oggetto tramite le funzioni SynapseValue, SynapseValueFromID o SynapsePassExecution.

Parameter	Description
None	None

OnFireExecution, Generic Event

Description Evento notificato ogni qualvolta la sinapsi dell'oggetto disegno sta per essere eseguita. Il sistema stabilisce a quale disegno attivare la sinapsi in base al flusso logico rappresentato dai collegamenti e all'ordine di tabulazione dato a questi collegamenti. Nel momento in cui all'interno del disegno, che ha in corso l'esecuzione, vengono impostate le sinapsi di uscita, il sistema provvede, in base a quanto detto prima, a passare l'esecuzione della macro ad un altro disegno scatenando l'evento OnFireExecution. Quindi in tale evento vanno inserite le funzioni logiche riguardanti la gestione delle sinapsi di ingresso e uscita del simbolo.

Parameter	Description
None	None

OnFireSynapse, Generic Event

Description Questo evento viene generato ogni qualvolta una sinapsi di input riceve il valore da una sinapsi di output ad essa collegata, o meglio un oggetto disegno ha settato tramite le proprietà SynapseValue, SynapseValueFromID o SynapsePassExecution il valore della propria sinapsi di output alla quale è collegata la sinapsi di input che riceve l'evento.
Tramite il parametro SynapseName è possibile sapere quale sinapsi di input è interessata dall'evento appena descritto.

Parameter	Description
SynapseName As String	Nome della sinapsi che ha appena ricevuto il valore.

OnGesture, Generic Event

Description Questo evento viene notificato quando viene eseguito un movimento di Gestualità nel progetto.

Parameter	Description
Button As Integer	Indice del tasto mouse premuto: 1 = Sinistro 2 = Destro 4 = Centrale
Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT
X1 As Integer	Coordinata orizzontale riferita alla posizione del cursore al momento dell'inizio del movimento.
Y1 As Integer	Coordinata verticale riferita alla posizione del cursore al momento dell'inizio del movimento.
X2 As Integer	Coordinata orizzontale riferita alla posizione del cursore al momento della fine del movimento.
Y2 As Integer	Coordinata verticale riferita alla posizione del cursore al momento della fine del movimento.
ElapsedTime As Single	Tempo di durata del movimento di Gestualità.
bRet as Boolean	Se tale parametro viene impostato a True all'interno dell'evento il movimento di Gestualità verrà abortito e non eseguirà quindi nessuna operazione.

OnPostPaint, Generic Event

Description Evento notificato ogni volta che l'oggetto disegno ha subito un 'refresh' grafico sul sinottico da parte del sistema. Questo può avvenire in diverse circostanze, ad esempio a seguito dell'apertura di un sinottico, quando l'applicazione riprende il focus, e comunque tutte le volte che viene eseguita un'animazione o ricalcolata la posizione sullo schermo del disegno.
Il parametro hdc (Handle to the device context) dà informazioni utili all'utente esperto di Windows sul tipo di rinfresco grafico adottato da Movicon.

Parameter	Description
ByVal hdc As Long	Handle to the device context.

OnPrePaint, Generic Event

Description Evento notificato ogni volta che l'oggetto disegno sta per subire un 'refresh' grafico sul sinottico da parte del sistema. Questo può avvenire in diverse circostanze, ad esempio a seguito dell'apertura di un sinottico, quando l'applicazione riprende il focus, e comunque tutte le volte che viene eseguita un'animazione o ricalcolata la posizione sullo schermo del disegno.

Impostando a false il parametro bRet viene disabilitato il refresh grafico del simbolo. Il parametro hdc (Handle to the device context) dà informazioni utili all'utente esperto di Windows sul tipo di rinfresco grafico adottato da Movicon.

Parameter	Description
ByVal hdc As Long	Handle to the device context
bRet As Boolean	Abilitazione al ridisegno

OnPreSymbolLoading, Generic Event

Description Questo evento viene richiamato prima che l'oggetto sia inizializzato, quindi prima ancora dell'evento "SymbolLoading" e consente di effettuare quelle operazioni che altrimenti verrebbero vanificate dal fatto che l'oggetto è stato già inizializzato. Ad esempio è possibile cambiare la struttura di default di un simbolo o di un elemento all'interno di un simbolo ed è possibile impostare il nome pubblico dell'oggetto.



Dopo l'esecuzione dell'evento "OnPreSymbolLoading" il codice script dell'oggetto viene distrutto (come se non fosse mai stato eseguito). Questo significa che ad esempio la creazione di variabili di tipo oggetto all'interno dell'evento non saranno poi valide una volta che il simbolo è stato caricato.



L'evento "OnPreSymbolLoading" NON viene mai richiamato negli oggetti contenuti all'interno di un oggetto Sinottico Incastrato o Gruppo Tab.

Parameter	Description
None	None

OnSize, Generic Event

Description Questo evento si verifica al ridimensionarsi dell'oggetto a runtime, dovuto al ridimensionamento del sinottico o al ridimensionamento dell'oggetto stesso se è abilitata l'opzione Trascinamento/Ridimensionamento.

Parameter	Description
nWidth As Integer	Parametro di uscita. Ritorna la larghezza dell'oggetto in pixel.
nHeight As Integer	Parametro di uscita. Ritorna l'altezza dell'oggetto in pixel.

Example:

```
Public Sub OnSize(ByVal nWidth As Integer, ByVal nHeight As Integer)
    MsgBox "object width = " & nWidth & ", object height = " & nHeight
End Sub
```

OnTextChanged, Generic Event

Description Evento notificato quando il testo del titolo dell'oggetto è stato modificato da tastiera. Restituisce la variabile stringa ChangedText contenente il nuovo testo.



L'evento "OnTextChanged" in un oggetto richiede automaticamente il login di un utente che rispetti l'impostazione di livello di password richiesto (impostato sull'oggetto stesso), nel caso in cui la gestione password del progetto sia abilitata.

Parameter	Description
ChangedText As String	Nuovo testo contenuto nel campo Titolo.

OnTextChanging, Generic Event

Description Evento notificato quando il titolo dell'oggetto viene modificato da tastiera. La variabile booleana bRet consente o meno tale modifica: se la variabile bRet viene posta a False all'interno della funzione, il cambiamento del testo contenuto nell'oggetto non ha effetto.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento del testo.

OnTimer, Generic Event

Description Evento notificato con un periodo di 1/2 di secondo circa (tempo non garantito) durante la fase di run-time. Durante il Test il periodo è proporzionale alla velocità di test impostata. Il tempo di esecuzione dell'evento può comunque essere personalizzato tramite la chiave di registro TimerEventFrequency.

Parameter	Description
None	None

OnToolTip, Generic Event

Description Evento notificato quando l'oggetto è nelle condizioni di visualizzare una stringa "pop-up" denominata ToolTip (ad esempio è puntato dal mouse). Restituisce la variabile booleana Show che è True se il mouse si trova sull'oggetto e False se il mouse è fuori dall'oggetto. Per visualizzare il ToolTip deve essere impostato il relativo metodo.

Parameter	Description
Show As Boolean	Variabile che indica se il cursore mouse è sull'oggetto.

SetFocus, Generic Event

Description Evento notificato quando l'oggetto disegno riceve il focus o viene selezionato.

Parameter	Description
None	None

SymbolLoading, Generic Event

Description Evento notificato quando l'oggetto disegno viene caricato in memoria, quindi all'apertura del sinottico di appartenenza. Questo evento è indipendente dalle condizioni di visibilità del disegno stesso.

Parameter	Description
None	None

SymbolUnloading, Generic Event

Description Evento notificato quando l'oggetto disegno viene scaricato dalla memoria alla chiusura del sinottico. Questo evento è indipendente dalle condizioni di visibilità del disegno stesso.

Parameter	Description
None	None

1.15.10. GridWndCmdTarget

Events

OnQueryEnd, GridWndCmdTarget Event

Description Evento notificato al termine dell'estrazione dei dati mediante query di selezione SQL.

Parameter	Description
None	None

OnQueryNext, GridWndCmdTarget Event

Description Evento notificato al termine dopo l'estrazione di ogni record dal database associato.

Parameter	Description
bRet As Boolean	Abilitazione all'esecuzione della funzione

OnQueryStart, GridWndCmdTarget Event

Description Evento notificato all'inizio dell'estrazione dell'estrazione dei dati mediante query di selezione SQL.

Parameter	Description
bRet As Boolean	Abilitazione all'esecuzione della funzione

OnSelChanged, GridWndCmdTarget Event

Description Evento notificato ogni volta che cambia la selezione delle celle nella griglia.

Parameter	Description
nRow As Integer	Numero riga della cella selezionata
nCol As Integer	Numero colonna della cella selezionata

OnSelChanging, GridWndCmdTarget Event

Description Evento notificato ogni volta che sta per cambiare la selezione delle celle nella griglia. Questo evento può essere utilizzato per intercettare il click del mouse su una qualsiasi cella. I parametri nRow e nCol variano rispettivamente fra 0,...,(numero delle righe-1) e 0,..., (numero delle colonne-1). Quando il parametro nRow vale zero identifica le intestazioni di colonna, mentre nCol le intestazioni di riga.

Parameter	Description
nRow As Integer	Numero riga della cella selezionata
nCol As Integer	Numero colonna della cella selezionata

OnSQLError, GridWndCmdTarget Event

Description Evento notificato ogni volta che si genera un errore sull'estrazione dei dati dal database.

Parameter	Description
Error As String	Stringa contenente l'errore del modulo di gestione di database.

OnUpdatingDSN, GridWndCmdTarget Event

Description Evento notificato ogni volta viene richiesta una modifica sul database collegato alla griglia mediante il comando 'Salva' (comando SQL equivalente a INSERT\DELETE\UPDATE).

Parameter	Description
SQLCommand As String	Stringa contenente il comando SQL da eseguire
bRet As Boolean	Abilitazione all'esecuzione del comando

Functions

AddColumn, GridWndCmdTarget Function

Syntax AddColumn(_IpszColumnName)

Description Questa funzione permette di inserire una nuova colonna nell'oggetto griglia. Il parametro passato contiene il nome della colonna da inserire.

Parameter	Description
IpszColumnName As String	Nome della colonna da inserire

Result None

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    objGrid.AutoLayout = True
    objGrid.AddColumn("Nuova colonna")
    objGrid.RecalcLayout
End Sub
Public Sub SymbolLoading()
Set objGrid = GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

CellEditable, GridWndCmdTarget Function

Syntax CellEditable(_nRow,_nCol)

Description Questo funzione restituisce il valore booleano True se il contenuto della cella referenziata col numero di riga e il numero di colonna è editabile.

Parameter	Description
nRow As Long	Numero di riga
nCol As Long	Numero di colonna

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    If objGrid.CellEditable(1,1) Then
        MsgBox("Is editable!!",vbOkOnly,GetProjectTitle)
    Else
        MsgBox("Is Not editable!!",vbOkOnly,GetProjectTitle)
    End If
End Sub
Public Sub SymbolLoading()
    Set objGrid = GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

DeleteColumn, GridWndCmdTarget Function

Syntax DeleteColumn(_lpszColumnName)

Description Questa funzione permette di cancellare una colonna nell'oggetto griglia. Il parametro passato contiene il nome della colonna da cancellare. La funzione restituisce il valore booleano True se la cancellazione è andata a buon fine.

Parameter	Description
lpszColumnName As String	Nome della colonna da cancellare

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    objGrid.AutoLayout = True
    Debug.Print objGrid.DeleteColumn("Colonna10")
    objGrid.RecalcLayout
End Sub
Public Sub SymbolLoading()
    Set objGrid = GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

DeleteRow, GridWndCmdTarget Function

Syntax DeleteRow()

Description Questa funzione elimina la riga selezionata dalla griglia.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    objGrid.DeleteRow
End Sub
Public Sub SymbolLoading()
Set objGrid = GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

EditCopy, GridWndCmdTarget Function

Syntax EditCopy()

Description Questa proprietà restituisce il valore booleano True se il dato contenuto nella cella referenziata dal numero di riga e dal numero di colonna passati come parametro è stato modificato.



Questa funzione non è supportata in Windows CE.

Parameter	Description
None	None

Result None

Example:

```
Option Explicit
Dim objGrid As GridWndCmdTarget
Dim X
Public Sub Click()
    objGrid.SetSelectedRange(1,1,1,3)
    objGrid.EditCopy
    Debug.Print Clipboard$()
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

EnsureVisible, GridWndCmdTarget Function

Syntax EnsureVisible()

Description Questa funzione forza lo scroll della finestra griglia in modo tale da rendere visibile la cella referenziata dal numero di riga e dal numero di colonna passati come parametri.

Parameter	Description
None	None

Result None

Example:

```
Option Explicit
Dim objGrid As GridWndCmdTarget
Dim X
Public Sub Click()
    objGrid.EnsureVisible(3,3)
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

FocusCellEditable, GridWndCmdTarget Function

Syntax FocusCellEditable()

Description Questo funzione restituisce il valore booleano True se il contenuto della cella è editabile.

Parameter	Description
None	none

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.FocusCellEditable
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

GetColCount, GridWndCmdTarget Function

Syntax GetColCount()

Description Questa funzione il numero di colonne caricate nell'oggetto griglia.

Parameter	Description
None	None

Result Long

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.GetColCount
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

GetRowCount, GridWndCmdTarget Function

Syntax GetRowCount()

Description Questa funzione restituisce il numero di righe caricate nell'oggetto griglia. Il numero di righe corrisponde al numero di record reperiti dalla query di selezione nel caso di un collegamento ad un database, o al numero di righe presenti nel file di testo associato.

Parameter	Description
None	None

Result Long

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.GetRowCount
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

GetSelectedRange, GridWndCmdTarget Function

Syntax GetSelectedRange(_nMinRow, _nMinCol, _nMaxRow, _nMaxCol)

Description Questa funzione consente di recuperare il numero di righe e colonne iniziali e finali di una selezione di più celle.

Parameter	Description
nMinRow as long	Numero di riga iniziale della selezione.
nMinCol as long	Numero di colonna iniziale della selezione.
nMaxRow as long	Numero di riga finale della selezione.

nMaxCol as long	Numero di colonna finale della selezione.
-----------------	---

Result Long

Example:

```
Public Sub Click()
    Dim objGrid As GridWndCmdTarget
    Dim nMinRow As Long
    Dim nMinCol As Long
    Dim nMaxRow As Long
    Dim nMaxCol As Long

    Set objGrid = GetSynopticObject.GetSubObject("objGrid").GetObjectInterface
    objGrid.GetSelectedRange(nMinRow, nMinCol, nMaxRow, nMaxCol)

    MsgBox "Grid Selected Range:" & vbCrLf & _
        "MinRow = " & nMinRow & vbCrLf & _
        "MinCol = " & nMinCol & vbCrLf & _
        "MaxRow = " & nMaxRow & vbCrLf & _
        "MaxCol = " & nMaxCol, vbInformation, GetProjectTitle

    Set objGrid = Nothing
End Sub
```

InsertRow, GridWndCmdTarget Function

Syntax InsertRow()

Description Questa funzione inserisce una riga in fondo alla griglia.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    objGrid.InsertRow
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

IsCellSelected, GridWndCmdTarget Function

Syntax IsCellSelected(_nRow,_nCol)

Description Questo funzione restituisce il valore booleano True se la cella referenziata dal numero di riga e dal numero di colonna è selezionata.

Parameter	Description
nRow As Long	Numero di riga
nCol As Long	Numero di colonna

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    NumericEntry("riga")
    NumericEntry("colonna")
    If objGrid.IsCellSelected(riga,colonna) Then
        MsgBox("Is selected!!",vbOkOnly,GetProjectTitle)
    Else
        MsgBox("Is NOT selected!!",vbOkOnly,GetProjectTitle)
    End If
End Sub
Public Sub SymbolLoading()
    Set                                objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface =
End Sub
```

IsCellValid, GridWndCmdTarget Function

Syntax IsCellValid(_nRow,_nCol)

Description Questo funzione restituisce il valore booleano True se la cella referenziata dal numero di riga e dal numero di colonna esiste nella finestra griglia.

Parameter	Description
nRow As Long	Numero di riga
nCol As Long	Numero di colonna

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    NumericEntry("riga")
    NumericEntry("colonna")
    If objGrid.IsCellValid(riga,colonna) Then
        MsgBox("Is valid!!",vbOkOnly,GetProjectTitle)
    Else
        MsgBox("Is NOT valid!!",vbOkOnly,GetProjectTitle)
    End If
End Sub
Public Sub SymbolLoading()
    Set                                objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface =
End Sub
```

IsCellVisible, GridWndCmdTarget Function

Syntax IsCellVisible(_nRow,_nCol)

Description Questo funzione restituisce il valore booleano True se la cella referenziata dal numero di riga e dal numero di colonna è visibile sullo schermo.

Parameter	Description
nRow As Long	Numero di riga
nCol As Long	Numero di colonna

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    NumericEntry("riga")
    NumericEntry("colonna")
    If objGrid.IsCellVisible(riga,colonna) Then
        MsgBox("Is visible!!",vbOkOnly,GetProjectTitle)
    Else
        MsgBox("Is NOT visible!!",vbOkOnly,GetProjectTitle)
    End If
End Sub
Public Sub SymbolLoading()
    Set objGrid = GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

LoadExtSettings, GridWndCmdTarget Function

Syntax LoadExtSettings

Description Questa funzione permette di caricare la configurazione dell'oggetto dal file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
Dim objSymbol As GridWndCmdTarget
Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
If objSymbol Is Nothing Then Exit Sub
objSymbol.ExtSettingsFile= "test.sxml"
objSymbol.LoadExtSettings
Set objSymbol = Nothing
End Sub
```

LoadFromTextFile, GridWndCmdTarget Function

Syntax LoadFromTextFile()

Description Questo funzione serve per caricare i dati da un file di testo all'interno della griglia. Il file di testo deve essere salvato in formato unicode.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    objGrid.TextFileName = MyProjectPath & "\data\prodotti.txt"
    objGrid.LoadFromTextFile
    objGrid.Refresh
End Sub
Public Sub SymbolLoading()
    Set objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

RecalcLayout, GridWndCmdTarget Function

Syntax RecalcLayout()

Description Questa funzione esegue un'aggiornamento grafico dell'oggetto. E' necessario eseguire tale funzione dopo che e' stata modificata una proprietà il cui impatto grafico sia particolarmente consistente, ad esempio quando viene modificata la proprietà AutoLayout dell'oggetto.

Parameter	Description
None	None

Result None

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    objGrid.AutoLayout = Not(objGrid.AutoLayout)
    objGrid.RecalcLayout
End Sub
Public Sub SymbolLoading()
    Set objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

Refresh, GridWndCmdTarget Function

Syntax Refresh()

Description Questa funzione rinfresca la visualizzazione dei dati all'interno della griglia secondo la query impostata.

Parameter	Description
None	none

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    objGrid.Refresh
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

SaveExtSettings, GridWndCmdTarget Function

Syntax SaveExtSettings

Description Questa funzione permette di salvare la configurazione dell'oggetto nel file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
Dim objSymbol As GridWndCmdTarget
Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
If objSymbol Is Nothing Then Exit Sub
objSymbol.ExtSettingsFile = "test.sxml"
objSymbol.SaveExtSettings
Set objSymbol = Nothing
End Sub
```

SaveToTextFile, GridWndCmdTarget Function

Syntax SaveToTextFile()

Description Questo funzione serve per salvare i dati all'interno della griglia in un file di testo. Il file di testo sarà in formato Unicode.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    objGrid.TextFileName = MyProjectPath & "\data\prodotti.txt"
    objGrid.LoadFromTextFile
    objGrid.CellText(1,0) = "Cambio Testo"
    objGrid.Refresh
    objGrid.SaveToTextFile
End Sub
Public Sub SymbolLoading()
    Set                                objGrid                                =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

SelectAll, GridWndCmdTarget Function

Syntax SelectAll()

Description Questa funzione permette di selezionare tutte le celle dell'oggetto griglia.

Parameter	Description
None	None

Result None

Example:

```
Option Explicit
Dim objGrid As GridWndCmdTarget
Dim X
Public Sub Click()
    objGrid.SelectAll
    objGrid.EditCopy
    MsgBox Clipboard$,vbOkOnly,GetProjectTitle
End Sub
Public Sub SymbolLoading()
    Set                                objGrid                                =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```


SetSelectedRange, GridWndCmdTarget Function

Syntax SetSelectedRange(_nMinRow, _nMinCol, _nMaxRow, _nMaxCol)

Description Questa funzione permette di selezionare tutte le celle dell'oggetto griglia comprese nel range passato come parametro.

Parameter	Description
nMinRow As Long	Numero di riga iniziale
nMinCol As Long	Numero di colonna iniziale
nMaxRow As Long	Numero di riga finale
nMaxCol As Long	Numero di colonna finale

Result None

Example:

```
Option Explicit
Dim objGrid As GridWndCmdTarget
Dim X
Public Sub Click()
    objGrid.SetSelectedRange(1,1,3,3)
    objGrid.EditCopy
    MsgBox Clipboard$, vbOkOnly, GetProjectTitle
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

UpdateDatabase, GridWndCmdTarget Function

Syntax UpdateDatabase()

Description Questa funzione salva i dati sul database relativi alle celle modificate tramite codice (con la funzione FocusCellText) o tastiera. La funzione ritorna il valore booleano True se il salvataggio è stato eseguito.

Parameter	Description
None	none

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    objGrid.FocusCellCol = 1
    objGrid.FocusCellRow = 1
    objGrid.FocusCellText = "text1"
```

```

        objGrid.RcalcLayout
        objGrid.UpdateDatabase
End Sub
Public Sub SymbolLoading()
    Set                                objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface =
End Sub

```

UpdateVariables, GridWndCmdTarget Function

Syntax UpdateVariables()

Description Questa funzione permette di aggiornare le variabili aventi lo stesso nome delle colonne dell'oggetto griglia con i dati in essa contenuti. I dati con cui vengono aggiornate le variabili sono quelli corrispondenti alla riga selezionata. Questa funzione ha effetto solo se la proprietà UpdateVariable dell'oggetto è impostata al valore booleano True.

Parameter	Description
None	none

Result Boolean

Example:

```

Dim objGrid As GridWndCmdTarget
Public Sub Click()
    If objGrid.UpdateVariable Then
        objGrid.UpdateVariables
    End If
End Sub
Public Sub SymbolLoading()
    Set                                objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface =
End Sub

```

Properties

AutoLayout, GridWndCmdTarget Property

Syntax AutoLayout = _Boolean

Description Abilitando questa proprietà, il layout della lista verrà impostato in modo automatico. Questo significa che le colonne della tabella verranno automaticamente ridimensionate in modo che risultino visibili tutte quante nell'area della finestra di visualizzazione. Disabilitando invece questa proprietà, all'apertura della finestra le colonne avranno la dimensione definita in programmazione, con la possibilità che le ultime sulla destra non siano visibili se non attivando la barra di scorrimento orizzontale.

Parameter	Description
None	none

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    objGrid.AutoLayout = Not(objGrid.AutoLayout)
    objGrid.RecalcLayout
End Sub
Public Sub SymbolLoading()
    Set objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface =
End Sub
```

ButtonPos, GridWndCmdTarget Property

Syntax ButtonPos = _Integer

Description Questa proprietà imposta o restituisce la posizione dei pulsanti nella finestra .

Le posizioni possibili sono:
enum_gba_left = sinistra
enum_gba_top = alto
enum_gba_right = destra
enum_gba_bottom = basso

Parameter	Description
None	none

Result Integer

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Dim pos
    pos = objGrid.ButtonPos
    Select Case size
        Case enum_gba_left
            Debug.Print "Pos = Left"
        Case enum_gba_top
            Debug.Print "Pos = Top"
        Case enum_gba_right
            Debug.Print "Pos = Right"
        Case enum_gba_bottom
            Debug.Print "Pos = Bottom"
        Case Else
    End Select
End Sub
Public Sub SymbolLoading()
    Set objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface =
End Sub
```

ButtonSize, GridWndCmdTarget Property

Syntax ButtonSize = _Integer

Description Questa proprietà imposta o restituisce le dimensioni dei pulsanti nella finestra .

Le dimensioni possono essere:
enum_gbz_small = piccolo
enum_gbz_medium = medio
enum_gbz_large = grande

Parameter	Description
None	none

Result Integer

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Dim size
    size = objGrid.ButtonSize
    Select Case size
        Case enum_gbz_small
            Debug.Print "Size = Small"
        Case enum_gbz_medium
            Debug.Print "Size = Medium"
        Case enum_gbz_large
            Debug.Print "Size = Large"
        Case Else
    End Select
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

CellBkColor, GridWndCmdTarget Property

Syntax CellBkColor(_nRow,_nCol)

Description Questa proprietà imposta o restituisce il colore di sfondo della cella referenziata dal numero di riga e dal numero di colonna passati come parametro.



La colorazione delle celle funzionerà correttamente soltanto se la Griglia è impostata per visualizzare le righe con un colore unico, ovvero se la chiave di registro "RowsOneColor" è impostata a "1". In caso contrario le righe di colore grigio potranno assumere dei colori diversi da quelli impostati con la proprietà "CellBkColor".

Parameter	Description
nRow As Long	Numero di riga
nCol As Long	Numero di colonna

Result Long

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    objGrid.CellBkColor(1,1) = RGB(255,0,0)
End Sub
Public Sub SymbolLoading()
    Set objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

=

CellFgColor, GridWndCmdTarget Property

Syntax CellFgColor(_nRow,_nCol)

Description Questa proprietà imposta o restituisce il colore del testo della cella referenziata dal numero di riga e dal numero di colonna passati come parametro.

Parameter	Description
nRow As Long	Numero di riga
nCol As Long	Numero di colonna

Result Long

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    objGrid.CellFgColor(1,1) = RGB(255,0,0)
End Sub
Public Sub SymbolLoading()
    Set objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

=

CellModified, GridWndCmdTarget Property

Syntax CellModified(_nRow,_nCol) = _Boolean

Description Questa proprietà restituisce il valore booleano True se il dato contenuto nella cella referenziata dal numero di riga e dal numero di colonna passati come parametro è stato modificato. Al caricamento dei dati (in apertura pagina o con il metodo refresh) tutte le celle partono con la proprietà "modified" a false. Se poi viene effettuata una modifica da tastiera o da codice, la proprietà viene messa a true. E' possibile impostare tale proprietà a false prima di effettuare un refresh per verificare se i dati sono stati caricati effettivamente.

Parameter	Description
nRow As Long	Numero di riga
nCol As Long	Numero di colonna

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    If objGrid.CellModified(1,1) Then
        MsgBox("Has been modified!!!",vbOkOnly,GetProjectTitle)
        objGrid.CellModified(1,1) = False
    End If
End Sub
Public Sub SymbolLoading()
    Set objGrid =
        GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

CellText, GridWndCmdTarget Property

Syntax CellText(_nRow,_nCol)

Description Questa proprietà imposta o restituisce il testo contenuto nella cella referenziata dal numero di riga e dal numero di colonna passati come parametro.



Per salvare o aggiornare i dati nel Data Base dopo una eventuale modifica delle celle è necessario utilizzare la proprietà FocusCellText() per modificare il contenuto delle celle e quindi la funzione UpdateDatabase() per salvare i dati nel Data Base.

Parameter	Description
nRow As Long	Numero di riga
nCol As Long	Numero di colonna

Result String

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.CellText(1,1)
End Sub
Public Sub SymbolLoading()
    Set objGrid =
        GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

Clickable, GridWndCmdTarget Property

Syntax Clickable = _Boolean

Description Impostando a False questo proprietà, il controllo non potrà essere gestito nè con il mouse nè con la tastiera. In questo caso quindi non sarà possibile, ad esempio, ordinare le colonne, vedere l'eventuale help, eseguire eventuali comandi presenti nella finestra.

Parameter	Description
None	none

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.Clickable
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

ColumnsWidth, GridWndCmdTarget Property

Syntax ColumnsWidth = _String

Description Tramite questa proprietà è possibile leggere o impostare la larghezza in pixel delle singole colonne della Griglia. La proprietà è una stringa nella quale vanno impostati i valori indicanti la dimensione delle colonne, separati dal carattere pipe (|). Ogni valore verrà associato alla colonna in base all'ordine sequenziale.

Parameter	Description
None	none

Result String

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.ColumnsWidth
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

CopyBtnText, GridWndCmdTarget Property

Syntax CopyBtnText = _String

Description Questa proprietà imposta o restituisce l'eventuale testo personalizzato da visualizzare nel tasto 'Copia'.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre una stringa vuota)

Parameter	Description
-----------	-------------

None	none
------	------

Result String

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.CopyBtnText
End Sub
Public Sub SymbolLoading()
    Set                                     objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

DeleteBtnText, GridWndCmdTarget Property

Syntax DeleteBtnText = _String

Description Questa proprietà imposta o restituisce l'eventuale testo personalizzato da visualizzare nel tasto 'Cancella'.

Parameter	Description
None	none

Result String

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.DeleteBtnText
End Sub
Public Sub SymbolLoading()
    Set                                     objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

DSN, GridWndCmdTarget Property

Syntax DSN = _String

Description Questa proprietà consente di leggere o impostare il nome del collegamento **ODBC** per l'oggetto Griglia.

Parameter	Description
None	none

Result String

Example:

```
Dim objGrid As GridWndCmdTarget
```



```

Public Sub Click()
    Debug.Print objGrid.DSN
End Sub
Public Sub SymbolLoading()
    Set objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
=

```

ExtSettingsFile, GridWndCmdTarget Property

Syntax ExtSettingsFile = _String

Description Questa proprietà imposta o restituisce il file di configurazione esterna per l'oggetto referenziato. Il file può essere specificato anche in fase di design nella proprietà "File di Configurazione" dell'oggetto. L'estensione prevista per il file è ".XML".

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
    Dim objSymbol As GridWndCmdTarget
    Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
    If objSymbol Is Nothing Then Exit Sub
    objSymbol.ExtSettingsFile = "test.xml"
    objSymbol.SaveExtSettings
    Set objSymbol= Nothing
End Sub

```

FocusCellBkColor, GridWndCmdTarget Property

Syntax FocusCellBkColor = _Long

Description Questa proprietà imposta o restituisce il colore di sfondo della cella selezionata.

Parameter	Description
None	none

Result Long

Example:

```

Dim objGrid As GridWndCmdTarget
Public Sub Click()
    objGrid.FocusCellBkColor = RGB(255,0,0)
    objGrid.RecalcLayout
End Sub
Public Sub SymbolLoading()
    Set objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
=

```

End Sub

FocusCellCol, GridWndCmdTarget Property

Syntax FocusCellCol = _Long

Description Questo proprietà imposta o restituisce il numero relativo alla colonna della cella selezionata. Restituisce il valore -1 se non è selezionata alcuna cella.

Parameter	Description
None	none

Result Long

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.FocusCellCol
End Sub
Public Sub SymbolLoading()
    Set                                objGrid                                =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

FocusCellFgColor, GridWndCmdTarget Property

Syntax FocusCellFgColor = _Long

Description Questo proprietà imposta o restituisce il colore del testo della cella selezionata.

Parameter	Description
None	none

Result Long

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    objGrid.FocusCellFgColor = RGB(255,0,0)
    objGrid.RecalcLayout
End Sub
Public Sub SymbolLoading()
    Set                                objGrid                                =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

FocusCellModified, GridWndCmdTarget Property

Syntax FocusCellModified = _Boolean

Description Questo proprietà restituisce il valore booleano True se il contenuto della cella che ha ottenuto il fuoco è cambiato. Al caricamento dei dati (in apertura pagina o con il metodo refresh) tutte le celle partono con la proprietà "modified" a false. Se poi viene effettuata una modifica da tastiera o da codice, la proprietà viene messa a true. E' possibile impostare tale proprietà a false prima di effettuare un refresh per verificare se i dati sono stati caricati effettivamente.

Parameter	Description
None	none

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.FocusCellModified
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

FocusCellRow, GridWndCmdTarget Property

Syntax FocusCellRow = _Long

Description Questo proprietà imposta o restituisce il numero relativo alla riga della cella selezionata. Restituisce il valore -1 se non è selezionata alcuna cella.

Parameter	Description
None	none

Result Long

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.FocusCellRow
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

FocusCellText, GridWndCmdTarget Property

Syntax FocusCellText = _String

Description Questa proprietà imposta o restituisce il testo della cella selezionata. Per aggiornare il database dopo un'eventuale modifica si utilizza la funzione UpdateDatabase.

Parameter	Description
None	none

Result String

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    objGrid.FocusCellCol = 1
    objGrid.FocusCellRow = 1
    objGrid.FocusCellText = "text1"
    objGrid.RcalcLayout
    objGrid.UpdateDatabase
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

GraphicButtons, GridWndCmdTarget Property

Syntax GraphicButtons = _Boolean

Description Abilitando questa proprietà i pulsanti dell'oggetto Griglia verranno disegnati usando un'icona invece del testo. Il testo opportuno verrà invece visualizzato come tooltip posizionando il mouse sopra ad un pulsante.



Il tooltip non viene gestito nella versione Windows CE.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()
    GraphicButtons = True
    RecalcLayout
End Sub
```

InsertBtnText, GridWndCmdTarget Property

Syntax InsertBtnText = _String

Description Questa proprietà imposta o restituisce l'eventuale testo personalizzato da visualizzare nel tasto 'Inserisci'.

Parameter	Description
None	none

Result String

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.InsertBtnText
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

Promptpad,GridWndCmdTarget_Property

Syntax PromptPad = _Boolean

Description Impostando la proprietà al valore "True" verrà utilizzato il Pad alfanumerico per l'editazione dei valori della griglia, impostandola a "False" i valori saranno direttamente editabili nelle celle della griglia.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim objGrid As GridWndCmdTarget
    Set objGrid = GetSynopticObject.GetSubObject("grid1").GetObjectInterface
    If objGrid Is Nothing Then Exit Sub
    If objGrid.PromptPad = False Then
        objGrid.PromptPad = True
    End If
End Sub
```

Query, GridWndCmdTarget Property

Syntax Query = _String

Description Questa proprietà imposta o restituisce la query in linguaggio standard SQL sui dati da selezionare dal Database.

Parameter	Description
None	none

Result String

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.Query
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

SaveBtnText, GridWndCmdTarget Property

Syntax SaveBtnText = _String

Description Questa proprietà imposta o restituisce l'eventuale testo personalizzato da visualizzare nel tasto 'Salva'.

Parameter	Description
None	none

Result String

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.SaveBtnText
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

SelectAllBtnText, GridWndCmdTarget Property

Syntax SelectAllBtnText = _String

Description Questa proprietà imposta o restituisce l'eventuale testo personalizzato da visualizzare nel tasto 'Seleziona Tutto'.

Parameter	Description
None	none

Result String

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.SelectAllBtnText
End Sub
Public Sub SymbolLoading()
    Set                                     objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

=

ShowCopyBtn, GridWndCmdTarget Property

Syntax ShowCopyBtn = _Boolean

Description Questa proprietà imposta o restituisce la possibilità di visualizzare il tasto 'Copia' nella finestra griglia. Se impostata al valore booleano true sarà visibile il tasto nella griglia.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre false)

Parameter	Description
None	none

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.ShowCopyBtn
End Sub
Public Sub SymbolLoading()
    Set                                     objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

=

ShowDeleteBtn, GridWndCmdTarget Property

Syntax ShowDeleteBtn = _Boolean

Description Questa proprietà imposta o restituisce la possibilità di visualizzare il tasto 'Cancella' per la cancellazione di una riga nella finestra griglia. Se impostata al valore booleano true sarà visibile il tasto nella griglia.

Parameter	Description
None	none

Result Boolean

Example:

```

Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.ShowDeleteBtn
End Sub
Public Sub SymbolLoading()
    Set objGrid =
        GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub

```

ShowInsertBtn, GridWndCmdTarget Property

Syntax ShowInsertBtn = _Boolean

Description Questa proprietà imposta o restituisce la possibilità di visualizzare il tasto 'Inserisci' nella finestra griglia per l'inserimento di una riga. Se impostata al valore booleano true sarà visibile il tasto nella griglia.

Parameter	Description
None	none

Result Boolean

Example:

```

Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.ShowInsertBtn
End Sub
Public Sub SymbolLoading()
    Set objGrid =
        GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub

```

ShowSaveBtn, GridWndCmdTarget Property

Syntax ShowSaveBtn = _Boolean

Description Questa proprietà imposta o restituisce la possibilità di visualizzare il tasto 'Salva' nella finestra griglia. Se impostata al valore booleano true sarà visibile il tasto nella griglia.

Parameter	Description
None	none

Result Boolean

Example:

```

Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.ShowSaveBtn
End Sub
Public Sub SymbolLoading()

```



```

        Set objGrid = GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
    End Sub

```

ShowSelectAllBtn, GridWndCmdTarget Property

Syntax ShowSelectAllBtn = _Boolean

Description Questa proprietà imposta o restituisce la possibilità di visualizzare il tasto 'Seleziona Tutto' nella finestra griglia. Se impostata al valore booleano true sarà visibile il tasto per selezionare tutti i dati nella griglia.

Parameter	Description
None	none

Result Boolean

Example:

```

Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.ShowSelectAllBtn
End Sub
Public Sub SymbolLoading()
    Set objGrid = GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub

```

ShowUpdateBtn, GridWndCmdTarget Property

Syntax ShowUpdateBtn = _Boolean

Description Questa proprietà imposta o restituisce la possibilità di visualizzare il tasto 'Aggiorna' nella finestra griglia. Se impostata al valore booleano true sarà visibile il tasto di aggiornamento dei dati nella griglia.

Parameter	Description
None	none

Result Boolean

Example:

```

Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.ShowUpdateBtn
End Sub
Public Sub SymbolLoading()
    Set objGrid = GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub

```

TextFileName, GridWndCmdTarget Property

Syntax TextFileName = _String

Description Questa proprietà imposta o restituisce il nome del file di testo (salvato informato Unicode) contenente i dati necessari per riempire la griglia.
ATTENZIONE: quando questa proprietà contiene un nome di file, l'oggetto Griglia preleverà i dati dal file e non più dall'eventuale collegamento ODBC definito nell'oggetto. Per far sì che l'oggetto Griglia lavori di nuovo con il collegamento ODBC è necessario reimpostare la proprietà "TextFileName" con un valore nullo (TextFileName = "").

Parameter	Description
None	none

Result String

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.TextFileName
End Sub
Public Sub SymbolLoading()
    Set                                     objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

TextSeparator, GridWndCmdTarget Property

Syntax TextSeparator = _String

Description Questa proprietà imposta o restituisce il testo di separazione fra un dato e il successivo all'interno del file di testo (lasciare questo campo vuoto per utilizzare il separatore di default ','). Il file di testo deve essere salvato in formato unicode.

Parameter	Description
None	none

Result String

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.TextSeparator
End Sub
Public Sub SymbolLoading()
    Set                                     objGrid
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

UpdateBtnText, GridWndCmdTarget Property

Syntax UpdateButtonText = _String

Description Questa proprietà imposta o restituisce l'eventuale testo personalizzato da visualizzare nel tasto 'Aggiorna'.

Parameter	Description
None	none

Result String

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.UpdateButtonText
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

UpdateVariable, GridWndCmdTarget Property

Syntax UpdateVariable = _Boolean

Description Impostando a True questa proprietà, viene abilitato l'aggiornamento delle variabili con lo stesso delle colonne. L'aggiornamento effettivo avviene richiamando la funzione UpdateVariables. I dati con cui vengono aggiornate le variabili sono quelli corrispondenti alla riga selezionata.

Parameter	Description
None	none

Result Boolean

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    objGrid.UpdateVariable = True
    objGrid.UpdateVariables
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

UserName, GridWndCmdTarget Property

Syntax UserName = _String

Description Questa proprietà specifica il nome dell'utente utilizzato per la connessione **ODBC**.

Parameter	Description
None	none

Result String

Example:

```
Dim objGrid As GridWndCmdTarget
Public Sub Click()
    Debug.Print objGrid.UserName
End Sub
Public Sub SymbolLoading()
    Set objGrid =
    GetSynopticObject.GetSubObject("GridWindow").GetObjectInterface
End Sub
```

1.15.11. HisLogWndCmdTarget

Events

OnFilter, HisLogWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di applicare un filtro per l'estrazione dei dati dal Log storico.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

OnPrint, HisLogWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di effettuare la stampa dei dati caricati nella finestra di visualizzazione.



Questo evento non è supportato in Windows CE.

Parameter	Description
bRet As Boolean	Abilitazione all'invio della stampa.

OnRefresh, HisLogWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di effettuare un rinfresco dei dati caricati nella finestra di visualizzazione.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

Functions

EditCopy, HisLogWndCmdTarget Function

Syntax EditCopy()

Description Questa proprietà esegue una copia negli appunti del contenuto della riga selezionata.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.EditCopy
    End If
    Set HisWindow = Nothing
End Sub
```

EditLayout, HisLogWndCmdTarget Function

Syntax EditLayout()

Description Questa funzione apre la finestra di configurazione dei campi da visualizzare nella finestra di visualizzazione del Log Storico.



La funzione verrà eseguita soltanto se nell'oggetto Finestra è stata abilitata la proprietà "Mostra Finestra di Controllo". In caso contrario la "Finestra Scelta Campo" non verrà aperta e la funzione ritornerà il valore "False"

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
```

```

Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
If Not HisWindow Is Nothing Then
    HisWindow.EditLayout
End If
Set HisWindow = Nothing
End Sub

```

LoadExtSettings, HisLogWndCmdTarget Function

Syntax LoadExtSettings

Description Questa funzione permette di caricare la configurazione dell'oggetto dal file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".XML".

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub Click()
Dim objSymbol As HisLogWndCmdTarget
Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
If objSymbol Is Nothing Then Exit Sub
objSymbol.ExtSettingsFile = "test.xml"
objSymbol.LoadExtSettings
Set objSymbol = Nothing
End Sub

```

RecalcLayout, HisLogWndCmdTarget Function

Syntax RecalcLayout()

Description Questa funzione esegue un'aggiornamento grafico dell'oggetto. E' necessario eseguire tale funzione dopo che e' stata modificata una proprietà il cui impatto grafico sia particolarmente consistente, come ad esempio la modifica della dimensione di una colonna.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Public Sub Click()
Dim HisWindow As HisLogWndCmdTarget
Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
If Not HisWindow Is Nothing Then
    HisWindow.AutoLayout = Not HisWindow.AutoLayout
End If

```

```

        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub

```

Refresh, HisLogWndCmdTarget Function

Syntax Refresh()

Description Questa funzione esegue un'aggiornamento dei dati nell'oggetto, utile ad esempio quando viene modificata la query di estrazione dei dati dal database del Log.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.Query = "SELECT * FROM SysMsgs ORDER BY SysMsgs.TimeCol DESC"
        HisWindow.Refresh
    End If
    Set HisWindow = Nothing
End Sub

```

SaveExtSettings, HisLogWndCmdTarget Function

Syntax SaveExtSettings

Description Questa funzione permette di salvare la configurazione dell'oggetto nel file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
    Dim objSymbol As HisLogWndCmdTarget
    Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
    If objSymbol Is Nothing Then Exit Sub
    objSymbol.ExtSettingsFile = "test.sxml"
    objSymbol.SaveExtSettings
    Set objSymbol = Nothing
End Sub

```

End Sub

Properties

AutoLayout, HisLogWndCmdTarget Property

Syntax AutoLayout = _Boolean

Description Abilitando questa proprietà il layout verrà impostato in modo automatico. Questo significa che le colonne verranno automaticamente ridimensionate in modo che risultino visibili tutte quante nell'area della "Finestra Log Storico". Disabilitando invece questa proprietà all'apertura della finestra le colonne avranno la dimensione definita in programmazione, con la possibilità che le ultime sulla destra non siano visibili se non attivando la barra di scorrimento orizzontale.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.AutoLayout = Not HisWindow.AutoLayout
        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub
```

ButtonPos, HisLogWndCmdTarget Property

Syntax ButtonPos

Description Questa impostazione restituisce la posizione in cui dovranno apparire i pulsanti nella "Finestra Log Storico".

Le possibilità sono:
0 = sinistra
1 = sopra
2 = destra
3 = sotto

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim objHisLogWnd As HisLogWndCmdTarget
```



```

Set objHisLogWnd =
GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
If Not objHisLogWnd Is Nothing Then
    MsgBox "objHisLogWnd's ButtonPos is " & objHisLogWnd.ButtonPos
    ,vbInformation,GetProjectTitle
    objHisLogWnd.ButtonPos = 2
    objHisLogWnd.RecalcLayout
Else
    MsgBox "objHisLogWnd is nothing",vbInformation,GetProjectTitle
End If
End Sub

```

ButtonSize, HisLogWndCmdTarget Property

Syntax ButtonSize

Description Questa impostazione restituisce la dimensione dei pulsanti che verranno poi visualizzati nella "Finestra Log Storico".

Le possibilità sono:

0 = piccolo

1 = medio

2 = grande

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Public Sub Click()
    Dim objHisLogWnd As HisLogWndCmdTarget
    Set objHisLogWnd =
GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
If Not objHisLogWnd Is Nothing Then
    MsgBox "objHisLogWnd's ButtonSize is " &
objHisLogWnd.ButtonSize,vbInformation,GetProjectTitle
    objHisLogWnd.ButtonSize= 2
    objHisLogWnd.RecalcLayout
Else
    MsgBox "objHisLogWnd is nothing",vbInformation,GetProjectTitle
End If
End Sub

```

Clickable, HisLogWndCmdTarget Property

Syntax Clickable = _Boolean

Description Questa proprietà permette di definire se l'operatore potrà interagire con la finestra di visualizzazione del Log Storico. Disabilitando questa proprietà il controllo non potrà infatti essere gestito nè con il mouse nè con la tastiera.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.Clickable = Not HisWindow.Clickable
        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub
```

EventType, HisLogWndCmdTarget Property

Syntax EventType = _Integer

Description Questa proprietà imposta o restituisce la tipologia di messaggi da visualizzare nella "Finestra Log Storico".

I valori possibili sono:

- 0 = Messaggi di Sistema
- 1 = Messaggi di Allarme
- 2 = Messaggi dei Driver
- 3 = Tutti i Messaggi

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.EventType = 1
    End If
    Set HisWindow = Nothing
End Sub
```

ExtSettingsFile, HisLogWndCmdTarget Property

Syntax ExtSettingsFile = _String

Description Questa proprietà imposta o restituisce il file di configurazione esterna per l'oggetto referenziato. Il file può essere specificato anche in fase di design nella proprietà "File di Configurazione" dell'oggetto. L'estensione prevista per il file è ".XML".

Parameter	Description
-----------	-------------

None	None
------	------

Result Long

Example:

```
Public Sub Click()
Dim objSymbol As HisLogWndCmdTarget
Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
If objSymbol Is Nothing Then Exit Sub
objSymbol.ExtSettingsFile = "test.sxml"
objSymbol.SaveExtSettings
Set objSymbol= Nothing
End Sub
```

FilterBtnText, HisLogWndCmdTarget Property

Syntax FilterBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando di stampa report dei dati visualizzati nella finestra del Log Storico. Se non specificato nulla, Movicon utilizza il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Sub Click()
Dim objHisLogWnd As HisLogWndCmdTarget
Set objHisLogWnd = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
If Not objHisLogWnd Is Nothing Then
objHisLogWnd.RefreshBtnText = "Riefresh all"
objHisLogWnd.PrintBtnText = "Print report"
objHisLogWnd.FilterBtnText = "Sort by..."
objHisLogWnd.RecalcLayout
MsgBox "objHisLogWnd's RefreshBtnText is " & objHisLogWnd.RefreshBtnText ,vbInformation,GetProjectTitle
MsgBox "objHisLogWnd's PrintBtnText is " & objHisLogWnd.PrintBtnText ,vbInformation,GetProjectTitle
MsgBox "objHisLogWnd's FilterBtnText is " & objHisLogWnd.FilterBtnText ,vbInformation,GetProjectTitle
Else
MsgBox "objHisLogWnd is nothing",vbInformation,GetProjectTitle
End If
End Sub
```

FilterEvent, HisLogWndCmdTarget Property

Syntax FilterEvent = _String

Description Questa proprietà imposta o restituisce il filtro 'Tipo Evento' per la visualizzazione dei messaggi nella finestra di Log Storico di Movicon.

Il filtro può assumere i seguenti valori:

ALARM ACK"ALARM OFF
 ALARM ON
 ALARM RESET
 Com. Driver
 System
 Trace

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim objHisLogWnd As HisLogWndCmdTarget
    Set objHisLogWnd =
    GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not objHisLogWnd Is Nothing Then
        MsgBox "objHisLogWnd's FilterEvent is " & objHisLogWnd.FilterEvent
        ,vbInformation,GetProjectTitle
        objHisLogWnd.FilterEvent = "System"
        objHisLogWnd.Refresh
    Else
        MsgBox "objHisLogWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub
```

FilterEventTypeCol, HisLogWndCmdTarget Property

Syntax FilterEventTypeCol = _Long

Description Questa proprietà imposta o restituisce il filtro 'Priorità' per la visualizzazione dei messaggi nella finestra di Log Storico di Movicon. La priorità si riferisce all'ID identificativo dell'evento registrato nella colonna EvNumCol. Questa impostazione ha senso solamente se si è specificato un valore diverso da "nessuno" nel campo "Condizione di Priorità".

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Public Sub Click()
    Dim objHisLogWnd As HisLogWndCmdTarget
    Set objHisLogWnd =
    GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not objHisLogWnd Is Nothing Then
        MsgBox "objHisLogWnd's FilterEventTypeCol is " &
        objHisLogWnd.FilterEventTypeCol,vbInformation,GetProjectTitle
        objHisLogWnd.FilterEventTypeCol = 0
        objHisLogWnd.Refresh
    Else
```

```

MsgBox "objHisLogWnd is nothing",vbInformation,GetProjectTitle
End If
End Sub

```

FilterEventTypeColCondition, HisLogWndCmdTarget Property

Syntax FilterEventTypeColCondition = _Integer

Description Questa proprietà imposta o restituisce il filtro 'Condizione Priorità' per la visualizzazione dei messaggi nella finestra di Log Storico di Movicon.

I valori possibili sono::

```

0 -> 'nessuno'
1 -> 'uguale'
2 -> 'maggiore'
3 -> 'minore'

```

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Public Sub Click()
    Dim objHisLogWnd As HisLogWndCmdTarget
    Set objHisLogWnd =
    GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not objHisLogWnd Is Nothing Then
        MsgBox "FilterEventTypeColCondition is " &
        objHisLogWnd.FilterEventTypeColCondition,vbInformation,GetProjectTitle
        objHisLogWnd.FilterEventTypeColCondition = 0
        objHisLogWnd.Refresh
    Else
        MsgBox "objHisLogWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub

```

FilterFromDate, HisLogWndCmdTarget Property

Syntax FilterFromDate = _Date

Description Questa proprietà imposta o restituisce il filtro 'Data iniziale' per la visualizzazione dei messaggi nella finestra di Log Storico di Movicon.

Parameter	Description
None	None

Result Date

Example:

```
Option Explicit
Public Sub Click()
    Dim objHisLogWnd As HisLogWndCmdTarget
    Set objHisLogWnd = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not objHisLogWnd Is Nothing Then
        MsgBox "objHisLogWnd's FilterFromDate is " & objHisLogWnd.FilterFromDate, vbInformation, GetProjectTitle
        objHisLogWnd.FilterFromDate = Now()
        objHisLogWnd.Refresh
    Else
        MsgBox "objHisLogWnd is nothing", vbInformation, GetProjectTitle
    End If
End Sub
```

FilterToDate, HisLogWndCmdTarget Property

Syntax FilterToDate = _Date

Description Questa proprietà imposta o restituisce il filtro 'Data finale' per la visualizzazione dei messaggi nella finestra di Log Storico di Movicon.

Parameter	Description
None	None

Result Date

Example:

```
Option Explicit
Public Sub Click()
    Dim objHisLogWnd As HisLogWndCmdTarget
    Set objHisLogWnd = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not objHisLogWnd Is Nothing Then
        MsgBox "objHisLogWnd's FilterToDate is " & objHisLogWnd.FilterToDate, vbInformation, GetProjectTitle
        objHisLogWnd.FilterToDate = Now()
        objHisLogWnd.Refresh
    Else
        MsgBox "objHisLogWnd is nothing", vbInformation, GetProjectTitle
    End If
End Sub
```

FilterUniqueID, HisLogWndCmdTarget Property

Syntax FilterFromDate = _Long

Description Questa proprietà permette di filtrare i record della Finestra Log Storico secondo il valore del campo "UniID" (UniqueID) registrato univoco per ogni allarme nella tabella Allarmi.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Dim objHisLogAlarm As HisLogWndCmdTarget
Dim objAlarm As AlarmCmdTarget
Dim objThreshold As AlarmThresholdCmdTarget

Public Sub Click()
    Set objAlarm = GetAlarm("Alarm VAR00001")
    If Not objAlarm Is Nothing Then
        Set objThreshold = objAlarm.GetAlarmThreshold("DigitalThreshold")
        If Not objThreshold Is Nothing Then
            Set objHisLogAlarm =
                GetSynopticObject.GetSubObject("AlarmHisLog").GetObjectInterface
            If Not objHisLogAlarm Is Nothing Then
                objHisLogAlarm.FilterUniqueID =
                    objThreshold.GetUniqueID
                objHisLogAlarm.Refresh
            End If
        End If
    End If
    Set objAlarm = Nothing
    Set objThreshold = Nothing
    Set objHisLogAlarm = Nothing
End Sub
```

FilterUser, HisLogWndCmdTarget Property

Syntax FilterUser = _String

Description Questa proprietà imposta o restituisce il filtro 'Utente' per la visualizzazione dei messaggi nella finestra di Log Storico di Movicon.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim objHisLogWnd As HisLogWndCmdTarget
    Set objHisLogWnd =
        GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not objHisLogWnd Is Nothing Then
        MsgBox "objHisLogWnd's FilterUser is " &
            objHisLogWnd.FilterUser, vbInformation, GetProjectTitle
        objHisLogWnd.FilterUser = "User00001"
        objHisLogWnd.Refresh
    Else
        MsgBox "objHisLogWnd is nothing", vbInformation, GetProjectTitle
    End If
End Sub
```

FormatDateTime, HisLogWndCmdTarget Property

Syntax FormatDateTime = _String

Description Questa proprietà permette di inserire il formato della data e dell'ora con cui verrà visualizzato l'orario nella colonna "Tempo Evento" della Finestra Log Storico. Tutti i codici di formattazione utilizzabili in questa proprietà sono elencati nella sezione Proprietà Stile dei Disegni e Controlli. Dopo avere modificato la proprietà è necessario eseguire un "Refresh" della finestra.

Parameter	Description
None	None

Result String

Example:

```
Dim WndHisLog As HisLogWndCmdTarget
Public Sub Click()
    Set WndHisLog = GetSynopticObject.GetSubObject("objHisLog").GetObjectInterface
    WndHisLog.FormatDateTime = "%d:%m:%Y %H:%M:%S"
    WndHisLog.Refresh
    Set WndHisLog = Nothing
End Sub
```

FormatDuration, HisLogWndCmdTarget Property

Syntax FormatDuration = _String

Description Questa proprietà permette di inserire il formato della durata riportato nella colonna "Durata" della Finestra Log Storico. Tutti i codici di formattazione utilizzabili in questa proprietà sono elencati nella sezione Proprietà Stile dei Disegni e Controlli. Dopo avere modificato la proprietà è necessario eseguire un "Refresh" della finestra.

Parameter	Description
None	None

Result String

Example:

```
Dim WndHisLog As HisLogWndCmdTarget
Public Sub Click()
    Set WndHisLog = GetSynopticObject.GetSubObject("objHisLog").GetObjectInterface
    WndHisLog.FormatDuration = "%D,%H:%M:%S"
    WndHisLog.Refresh
    Set WndHisLog = Nothing
End Sub
```


GraphicButtons, HisLogWndCmdTarget Property

Syntax GraphicButtons = _Boolean

Description Abilitando questa proprietà i pulsanti della Finestra Log Storico verranno disegnati usando un'icona invece del testo. Il testo opportuno verrà invece visualizzato come tooltip posizionando il mouse sopra ad un pulsante.



Il tooltip non viene gestito nella versione Windows CE.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()  
    GraphicButtons = True  
    RecalcLayout  
End Sub
```

IncludeMilliseconds, HisLogWndCmdTarget Property

Syntax IncludeMilliseconds = _Boolean

Description Questa proprietà permette di definire se visualizzare o meno i millisecondi nella colonna tempo della finestra.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit  
Public Sub Click()  
    Dim HisWindow As HisLogWndCmdTarget  
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface  
    If Not HisWindow Is Nothing Then  
        HisWindow.IncludeMilliseconds = Not HisWindow.IncludeMilliseconds  
        HisWindow.Refresh  
    End If  
    Set HisWindow = Nothing  
End Sub
```

MaxCount, HisLogWndCmdTarget Property

Syntax MaxCount = _Integer

Description Questa proprietà permette di definire il numero massimo di righe che la "Finestra Log Storico" potrà visualizzare.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.MaxCount = 150
        HisWindow.Refresh
    End If
    Set HisWindow = Nothing
End Sub
```

NetworkBackupServerName, HisLogWndCmdTarget Property

Syntax NetworkBackupServerName = _String

Description Questa proprietà imposta o restituisce il nome dell'eventuale Server di Networking di Backup usato per prelevare i dati da visualizzare nella Finestra Log Storico quando il Server Primario, ovvero quello impostato nella proprietà "NetowrkServerName", è in timeout.

Parameter	Description
None	None

Result String

Example:

```
Dim objHisLogWnd As HisLogWndCmdTarget
Public Sub Click()
    Debug.Print objHisLogWnd.NetworkBackupServerName
End Sub
Public Sub SymbolLoading()
    Set objHisLogWnd =
    GetSynopticObject.GetSubObject("HisLogWindow").GetObjectInterface
End Sub
```

NetworkServerName, HisLogWndCmdTarget Property

Syntax NetworkServerName = _String

Description Questa proprietà restituisce il nome dell'eventuale Server di Networking dal quale dovranno essere prelevati i dati da visualizzare nella Finestra Log Storico.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.NetworkServerName = "PERSONAL11"
        HisWindow.Refresh
    End If
    Set HisWindow = Nothing
End Sub
```

PrintBtnText, HisLogWndCmdTarget Property

Syntax PrintBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando di stampa report dei dati visualizzati nella finestra del Log Storico. Se non specificato nulla, Movicon utilizza il testo di default.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre una stringa vuota)

Parameter	Description
None	None

Result String

Example:

```
Sub Click()
    Dim objHisLogWnd As HisLogWndCmdTarget
    Set objHisLogWnd = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not objHisLogWnd Is Nothing Then
        objHisLogWnd.RefreshBtnText = "Riefresh all"
        objHisLogWnd.PrintBtnText = "Print report"
        objHisLogWnd.FilterBtnText = "Sort by..."
        objHisLogWnd.RecalcLayout
        MsgBox "objHisLogWnd's RefreshBtnText is " & objHisLogWnd.RefreshBtnText ,vbInformation,GetProjectTitle
    End If
End Sub
```

```

        MsgBox "objHisLogWnd's PrintBtnText is " &
objHisLogWnd.PrintBtnText ,vbInformation,GetProjectTitle
        MsgBox "objHisLogWnd's FilterBtnText is " &
objHisLogWnd.FilterBtnText ,vbInformation,GetProjectTitle
    Else
        MsgBox "objHisLogWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub

```

Project, HisLogWndCmdTarget Property

Syntax Project = _String

Description Questa proprietà permette di definire il nome del progetto figlio dal quale si vogliono recuperare i dati da visualizzare. Se il campo viene lasciato vuoto verrà utilizzato il progetto corrente.



In questa proprietà va inserito esclusivamente il nome di un eventuale progetto figlio del progetto corrente.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        Debug.Print HisWindow.Project
    End If
    Set HisWindow = Nothing
End Sub

```

Query, HisLogWndCmdTarget Property

Syntax Query = _String

Description Questa proprietà permette di definire una query di selezione in linguaggio SQL per l'estrazione dei dati contenuti nel database del Log. Questa query verrà eseguita per default ad ogni rinfresco dei dati nella finestra, sia esso eseguito automaticamente che su comando da operatore.

Le tabelle nel database del Log sono tre:

- SysMsgs Messaggi di Sistema
- Driver Messaggi relativi ai Driver
- Alarms Messaggi relativi agli Allarmi

sono:

- EventCol Event ID
- EvDescCol Event Text
- TimeCol Event Time
- UserCol User

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.Query = "SELECT * FROM SysMsgs ORDER BY SysMsgs.TimeCol DESC"
        HisWindow.Refresh
    End If
    Set HisWindow = Nothing
End Sub
```

RefreshBtnText, HisLogWndCmdTarget Property

Syntax RefreshBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando di rinfresco dei dati nella finestra di visualizzazione del Log Storico. Se non specificato nulla, Movicon utilizza il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Sub Click()
    Dim objHisLogWnd As HisLogWndCmdTarget
    Set objHisLogWnd = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not objHisLogWnd Is Nothing Then
        objHisLogWnd.RefreshBtnText = "Riefresh all"
        objHisLogWnd.PrintBtnText = "Print report"
        objHisLogWnd.FilterBtnText = "Sort by..."
        objHisLogWnd.RecalcLayout
        MsgBox "objHisLogWnd's RefreshBtnText is " & objHisLogWnd.RefreshBtnText ,vbInformation,GetProjectTitle
        MsgBox "objHisLogWnd's PrintBtnText is " & objHisLogWnd.PrintBtnText ,vbInformation,GetProjectTitle
        MsgBox "objHisLogWnd's FilterBtnText is " & objHisLogWnd.FilterBtnText ,vbInformation,GetProjectTitle
    Else
        MsgBox "objHisLogWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub
```

ReportFile, HisLogWndCmdTarget Property

Syntax ReportFile = _String

Description Questa proprietà imposta o restituisce il nome del file di report da utilizzare per la stampa dei dati visualizzati nella finestra di Log Storico. Il file deve essere creato con Report Designer o Crystal Report® (.rpt). Se il campo viene lasciato vuoto Movicon utilizza il file di report di default creato da Progea in formato Report Designer.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre una stringa vuota)

Parameter	Description
None	None

Result String

Example:

```
Sub Click()  
    Dim objHisLogWnd As HisLogWndCmdTarget  
    Set objHisLogWnd = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface  
    If Not objHisLogWnd Is Nothing Then  
        objHisLogWnd.ReportFile = "C:\Report1.rpt"  
    Else  
        MsgBox "objHisLogWnd is nothing",vbInformation,GetProjectTitle  
    End If  
End Sub
```

ShowFilterBtn, HisLogWndCmdTarget Property

Syntax ShowFilterBtn = _Boolean

Description Questa proprietà permette di visualizzare il pulsante di comando per l'operazione di filtro nella finestra di visualizzazione del Log Storico.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit  
Public Sub Click()  
    Dim HisWindow As HisLogWndCmdTarget  
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface  
    If Not HisWindow Is Nothing Then  
        HisWindow.ShowFilterBtn = Not HisWindow.ShowFilterBtn  
        HisWindow.RecalcLayout  
    End If  
    Set HisWindow = Nothing  
End Sub
```

ShowFlatGrid, HisLogWndCmdTarget Property

Syntax _ShowFlatGrid

Description Imposta la modalità di 'Visualizzazione Piatta' sulla Finestra Log Storico quando è impostata per la visualizzare gli eventi 'Messaggi di Allarmi' nelle proprietà Stile

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
Dim bSFG As Boolean
Dim oHLW As HisLogWndCmdTarget

Set oHLW = GetSynopticObject.GetSubObject("HisLogWnd").GetObjectInterface

bSFG = oHLW.ShowFlatGrid
oHLW.ShowFlatGrid = Not bSFG
oHLW.Refresh

Debug.Print "ShowFlatGrid: " & oHLW.ShowFlatGrid

End Sub
```

ShowPrintBtn, HisLogWndCmdTarget Property

Syntax ShowPrintBtn = _Boolean

Description Questa proprietà permette di visualizzare il pulsante il pulsante di comando per l'operazione di stampa dei dati visualizzati finestra di visualizzazione del Log Storico. La stampa viene eseguita utilizzando il file di report che dovrà essere specificato nella proprietà "File di Report". Movicon "passa" al report le stesse impostazioni di filtro per la stampa dei dati, che coincidono con ciò che è visualizzato nella finestra di Visualizzazione in questione.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre false)

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.ShowPrintBtn = Not HisWindow.ShowPrintBtn
        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub
```

ShowRefreshBtn, HisLogWndCmdTarget Property

Syntax ShowRefreshBtn = _Boolean

Description Questa proprietà permette di visualizzare il pulsante di comando per l'operazione di rinfresco dei dati nella finestra di visualizzazione del Log Storico.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.ShowRefreshBtn = Not HisWindow.ShowRefreshBtn
        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub
```

SortBy, HisLogWndCmdTarget Property

Syntax SortBy = _String

Description Questa proprietà imposta o restituisce il filtro 'Ordina Per' per la visualizzazione dei messaggi nella finestra di Log Storico di Movicon.

I campi possibili sono:

CommCol
DescCol
DurCol
EvDescCol
EvCol
EvNumCol
LocalCol
MSecCol

TimeCol
UserCol

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim objHisLogWnd As HisLogWndCmdTarget
    Set objHisLogWnd = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not objHisLogWnd Is Nothing Then
        MsgBox "objHisLogWnd's SortBy is " & objHisLogWnd.SortBy
        ,vbInformation,GetProjectTitle
        objHisLogWnd.SortBy = "EvCol"
        objHisLogWnd.Refresh
    Else
        MsgBox "objHisLogWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub
```

SubItemComment, HisLogWndCmdTarget Property

Syntax SubItemComment = _String

Description Questa proprietà permette di definire il testo che dovrà apparire come nome della colonna "Commento". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objHisLogWnd As HisLogWndCmdTarget
Public Sub Click()
    Debug.Print objHisLogWnd.SubItemEventNum
End Sub
Public Sub SymbolLoading()
    Set objHisLogWnd = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
End Sub
```

SubItemCommentPos, HisLogWndCmdTarget Property

Syntax SubItemCommentPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Commento" all'interno della Finestra Log Storico. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        Debug.Print HisWindow.SubItemCommentPos
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemCommentWidth, HisLogWndCmdTarget Property

Syntax SubItemCommentWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione del Log Storico. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.SubItemCommentWidth = 20
        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemDesc, HisLogWndCmdTarget Property

Syntax SubItemDesc = _String

Description Questa proprietà permette di definire il testo che dovrà apparire come nome della colonna "Descrizione". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objHisLogWnd As HisLogWndCmdTarget
Public Sub Click()
    Debug.Print objHisLogWnd.SubItemDesc
End Sub
Public Sub SymbolLoading()
    Set objHisLogWnd =
    GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
End Sub
```

SubItemDescPos, HisLogWndCmdTarget Property

Syntax SubItemDescPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Descrizione" all'interno della Finestra Log Storico. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        Debug.Print HisWindow.SubItemDescPos
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemDescWidth, HisLogWndCmdTarget Property

Syntax SubItemDescWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione del Log Storico. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.SubItemDescWidth = 20
        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemDuration, HisLogWndCmdTarget Property

Syntax SubItemDuration = _String

Description Questa proprietà permette di definire il testo che dovrà apparire come nome della colonna "Durata". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objHisLogWnd As HisLogWndCmdTarget
Public Sub Click()
    Debug.Print objHisLogWnd.SubItemDuration
End Sub
Public Sub SymbolLoading()
    Set objHisLogWnd =
    GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
End Sub
```

SubItemDurationPos, HisLogWndCmdTarget Property

Syntax SubItemDurationPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Durata" all'interno della Finestra Log Storico. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        Debug.Print HisWindow.SubItemDurationPos
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemDurationWidth, HisLogWndCmdTarget Property

Syntax SubItemDurationWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione del Log Storico. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.SubItemDurationWidth = 20
        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemEventId, HisLogWndCmdTarget Property

Syntax SubItemEventId = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna " Event Id". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.SubItemEventId = "Message Type"
        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemEventIdPos, HisLogWndCmdTarget Property

Syntax SubItemEventIdPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "ID Evento" all'interno della Finestra Log Storico. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        Debug.Print HisWindow.SubItemEventIdPos
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemEventIdWidth, HisLogWndCmdTarget Property

Syntax SubItemEventIdWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione del Log Storico. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.SubItemEventIdWidth = 20
        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemEventNum, HisLogWndCmdTarget Property

Syntax SubItemEventNum = _String

Description Questa proprietà permette di definire il testo che dovrà apparire come nome della colonna "Numero Eventi". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objHisLogWnd As HisLogWndCmdTarget
Public Sub Click()
    Debug.Print objHisLogWnd.SubItemEventNum
End Sub
Public Sub SymbolLoading()
    Set objHisLogWnd =
    GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
End Sub
```

SubItemEventNumPos, HisLogWndCmdTarget Property

Syntax SubItemEventNumPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Numero Evento" all'interno della Finestra Log Storico. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        Debug.Print HisWindow.SubItemEventNumPos
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemEventNumWidth, HisLogWndCmdTarget Property

Syntax SubItemEventNumWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione del Log Storico. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.SubItemEventNumWidth = 20
        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub
```


SubItemText, HisLogWndCmdTarget Property

Syntax SubItemText = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna " Event Text". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.SubItemText = "Description"
        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemTextPos, HisLogWndCmdTarget Property

Syntax SubItemTextPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Testo Evento" all'interno della Finestra Log Storico. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        Debug.Print HisWindow.SubItemTextPos
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemTextWidth, HisLogWndCmdTarget Property

Syntax SubItemTextWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione del Log Storico. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.SubItemTextWidth = 20
        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemTime, HisLogWndCmdTarget Property

Syntax SubItemTime = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna " Event Time". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.SubItemTime = "Date/Time"
        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemTimePos, HisLogWndCmdTarget Property

Syntax SubItemTimePos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Tempo Evento" all'interno della Finestra Log Storico. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        Debug.Print HisWindow.SubItemTimePos
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemTimeWidth, HisLogWndCmdTarget Property

Syntax SubItemTimeWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione del Log Storico. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.SubItemTimeWidth = 20
        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemUser, HisLogWndCmdTarget Property

Syntax SubItemUser = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "User". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.SubItemUser = "Logon Users"
        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemUserPos, HisLogWndCmdTarget Property

Syntax SubItemUserPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Utente" all'interno della Finestra Log Storico. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        Debug.Print HisWindow.SubItemUserPos
    End If
    Set HisWindow = Nothing
End Sub
```

SubItemUserWidth, HisLogWndCmdTarget Property

Syntax SubItemUserWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione del Log Storico. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim HisWindow As HisLogWndCmdTarget
    Set HisWindow = GetSynopticObject.GetSubObject("HisLog").GetObjectInterface
    If Not HisWindow Is Nothing Then
        HisWindow.SubItemUserWidth = 20
        HisWindow.RecalcLayout
    End If
    Set HisWindow = Nothing
End Sub
```

1.15.12. HourSelectorCmdTarget

Events

OnAddScheduler, HourSelectorCmdTarget Event

Description Evento notificato sul comando usato per aggiungere un nuovo schedatore in runtime.

Parameter	Description
bRet As Boolean	Impostato a "false" consente di annullare l'operazione.

Example:

```
Public Sub OnAddScheduler(ByRef bRet As Boolean)
    If MsgBox ("Do you want add a new scheduler objects ?", vbYesNo +
        vbQuestion, GetProjectTitle) = vbYes Then
        Debug.Print "Adding Scheduler..."
    Else
        Debug.Print "Deleting Add Scheduler..."
        bRet = False
    End If
End Sub
```

OnCancel, HourSelectorCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di cancellare le modifiche effettuate sui dati di pianificazione dello Schedulatore.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

OnRemoveScheduler, HourSelectorCmdTarget Event

Description Evento notificato sul comando usato per rimuovere uno schedulatore aggiunto in runtime.

Parameter	Description
bRet As Boolean	Impostato a "false" consente di annullare l'operazione.

Example:

```
Public Sub OnAddScheduler(ByRef bRet As Boolean)
    If MsgBox ("Do you want remove the scheduler objects?", vbYesNo +
        vbQuestion, GetProjectTitle) = vbYes Then
        Debug.Print "Removing Scheduler..."
    Else
        Debug.Print "Deleting Remove Scheduler..."
        bRet = False
    End If
End Sub
```

OnSave, HourSelectorCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di salvare le modifiche effettuate sui dati di pianificazione dello Schedulatore.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

OnSchedulerChanged, HourSelectorCmdTarget Event

Description Evento notificato tutte le volte che l'oggetto schedulatore associato alla finestra cambia di valore, ad esempio perchè ne viene selezionato un altro usando la combo box di selezione degli schedulatori.

Parameter	Description
-----------	-------------

Nessuno	
---------	--

Example:

```
Public Sub OnSchedulerChanged()
    Dim objScheduler As SchedulerCmdTarget
    If Scheduler = "" Then Exit Sub
    Set objScheduler = GetScheduler(Scheduler)
    If Not objScheduler Is Nothing Then
        MsgBox "New Scheduler Name: " & objScheduler.Name
    End If
    Set objScheduler = Nothing
End Sub
```

OnSwitchGridMode, HourSelectorCmdTarget Event

Description Evento notificato sul comando usato per passare dal modo tabella tempi al modo griglia, e viceversa.

Parameter	Description
bRet As Boolean	Impostato a "false" consente di annullare l'operazione.

Example:

```
Public Sub OnAddScheduler(ByRef bRet As Boolean)
    If MsgBox ("Do you want switch to grid mode ?", vbYesNo + vbQuestion,
        GetProjectTitle) = vbYes Then
        Debug.Print "Swtiching to grid mode..."
    Else
        Debug.Print "Deleting Swtich to grid mode..."
        bRet = False
    End If
End Sub
```

OnSwitchHolidays, HourSelectorCmdTarget Event

Description Evento notificato sul comando usato per passare dal piano ore normale a quello festivo, e viceversa.

Parameter	Description
bRet As Boolean	Impostato a "false" consente di annullare l'operazione.

Example:

```
Public Sub OnAddScheduler(ByRef bRet As Boolean)
    If MsgBox ("Do you want switch to holiday ?", vbYesNo + vbQuestion,
        GetProjectTitle) = vbYes Then
        Debug.Print "Swtiching to holiday..."
    Else
        Debug.Print "Deleting Swtich to holiday..."
        bRet = False
    End If
End Sub
```

```
End If
End Sub
```

Functions

Cancel, HourSelectorCmdTarget Function

Syntax Cancel()

Description Questa funzione annulla ogni modifica effettuata sui dati di pianificazione e ricarica la pianificazione dal file dello schedatore. La proprietà restituisce True se l'operazione è andata a buon fine.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objSelector As HourSelectorCmdTarget
Public Sub Click()
    Debug.Print objSelector.Cancel
End Sub
Public Sub SymbolLoading()
    Set objSelector =
    GetSynopticObject.GetSubObject("HourSelector").GetObjectInterface
End Sub
```

LoadExtSettings, HourSelectorCmdTarget Function

Syntax LoadExtSettings

Description Questa funzione permette di caricare la configurazione dell'oggetto dal file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".XML".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
Dim objSymbol As HourSelectorCmdTarget
Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
If objSymbol Is Nothing Then Exit Sub
objSymbol.ExtSettingsFile = "test.xml"
objSymbol.LoadExtSettings
Set objSymbol = Nothing
End Sub
```


RecalcLayout, HourSelectorCmdTarget Function

Syntax RecalcLayout()

Description Questa funzione esegue un'aggiornamento grafico dell'oggetto. E' necessario eseguire tale funzione dopo che e' stata modificata una proprietà il cui impatto grafico sia particolarmente consistente.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()  
    Dim bResult As Boolean  
    bResult = RecalcLayout  
    Debug.Print bResult  
End Sub
```

Save, HourSelectorCmdTarget Function

Syntax Save()

Description Questa funzione salva ogni modifica effettuata sui dati di pianificazione dello schedatore e restituisce True se l'operazione è andata a buon fine.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objSelector As HourSelectorCmdTarget  
Public Sub Click()  
    Debug.Print objSelector.Save  
End Sub  
Public Sub SymbolLoading()  
    Set objSelector =  
        GetSynopticObject.GetSubObject("HourSelector").GetObjectInterface  
End Sub
```

SaveExtSettings, HourSelectorCmdTarget Function

Syntax SaveExtSettings

Description Questa funzione permette di salvare la configurazione dell'oggetto nel file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".XML".

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
Dim objSymbol As HourSelectorCmdTarget  
Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface  
If objSymbol Is Nothing Then Exit Sub  
objSymbol.ExtSettingsFile = "test.xml"  
objSymbol.SaveExtSettings  
Set objSymbol = Nothing  
End Sub
```

Properties

AddBtnText, HourSelectorCmdTarget Property

Syntax AddBtnText = _String

Description Questa proprietà imposta o restituisce il testo visualizzato nel pulsante usato per aggiungere nuovi schedulatori in runtime. Impostando la proprietà con una stringa vuota verrà visualizzato il testo predefinito. La modifica diventerà effettiva soltanto dopo aver richiamato il metodo "RecalcLayout".

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
Dim objHourSelector As HourSelectorCmdTarget  
  
If objHourSelector Is Nothing Then Set objHourSelector =  
GetSynopticObject.GetSubObject("HourSelector1").GetObjectInterface  
If objHourSelector Is Nothing Then Exit Sub  
If objHourSelector.AddBtnText = "" Then  
objHourSelector.AddBtnText = "ADD (INS)"  
Else  
objHourSelector.AddBtnText = ""  
End If
```

```
objHourSelector.RecalcLayout  
End Sub
```

Border, HourSelectorCmdTarget Property

Syntax Border = _Integer

Description Questa proprietà imposta o restituisce il tipo di bordo impostato per l'oggetto secondo le possibilità Nessuno, A Sbalzo, Inciso, In Rilievo ed Infossato disponibili anche nelle proprietà generali del disegno.
Sono validi i seguenti valori: 0=nessuno, 1=a sbalzo, 2=inciso, 3=in rilievo, 4=infossato.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    For i = 0 To 4 Step 1  
        Border= i  
        sRet = Border  
        MsgBox "Border= " & sRet, vbOkOnly, GetProjectTitle  
    Next i  
End Sub
```

ButtonPos, HourSelectorCmdTarget Property

Syntax ButtonPos = _Integer

Description Questa proprietà imposta o restituisce la posizione per i pulsanti "Salva" e "Cancella" secondo le possibilità Sinistra, Sopra, Destra, Sotto disponibili anche nelle proprietà generali del disegno.
Sono validi i seguenti valori: 0=Sinistra, 1=Sopra, 2=Destra, 3=Sotto.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    For i = 0 To 3 Step 1  
        ButtonPos= i  
        sRet = ButtonPos  
        MsgBox "ButtonPos= " & sRet, vbOkOnly, GetProjectTitle  
    Next i  
End Sub
```

ButtonSize, HourSelectorCmdTarget Property

Syntax ButtonSize = _Integer

Description Questa proprietà imposta o restituisce la dimensione dei pulsanti "Salva" e "Cancella" secondo le possibilità Piccolo, Medio, Grande, disponibili anche nelle proprietà generali del disegno.
Sono validi i seguenti valori: 0=Piccolo, 1=Medio, 2=Grande.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    For i = 0 To 2 Step 1  
        ButtonSize = i  
        sRet = ButtonSize  
        MsgBox "ButtonSize= " & sRet, vbOkOnly, GetProjectTitle  
    Next i  
End Sub
```

CancelBtnText, HourSelectorCmdTarget Property

Syntax CancelBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando di cancellazione delle modifiche dei dati di pianificazione dello Schedulatore. Se non specificato nulla, Movicon utilizza il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objSelector As HourSelectorCmdTarget  
Public Sub Click()  
    Debug.Print objSelector.CancelBtnText  
End Sub  
Public Sub SymbolLoading()  
    Set objSelector =  
        GetSynopticObject.GetSubObject("HourSelector").GetObjectInterface  
    objSelector.CancelBtnText = "Cancella"  
End Sub
```

ColorSelCell, HourSelectorCmdTarget Property

Syntax ColorSelCell = _Long

Description Questa proprietà imposta o restituisce il colore di selezione per la Finestra Schedulatore

Parameter	Description
None	None

Result Long

Example:

```
Dim objSelector As HourSelectorCmdTarget
Public Sub Click()
    Debug.Print objSelector.ColorSelCell
End Sub
Public Sub SymbolLoading()
    Set objSelector =
        GetSynopticObject.GetSubObject("HourSelector").GetObjectInterface
End Sub
```

DaysText, HourSelectorCmdTarget Property

Syntax DaysText(_nIndex) = _String

Description Questa proprietà imposta o restituisce il testo visualizzato per le caselle orizzontali poste sulla sinistra. L'indice può assumere i valori da 0 a 7. Se non specificato nulla, Movicon utilizza il testo di default.

Parameter	Description
nIndex As Integer	Indice della casella referenziata.

Result String

Example:

```
Dim objSelector As HourSelectorCmdTarget
Public Sub Click()
    Debug.Print objSelector.DaysText(1)
End Sub
Public Sub SymbolLoading()
    Set objSelector =
        GetSynopticObject.GetSubObject("HourSelector").GetObjectInterface
    objSelector.DaysText(1) = "Domenica"
End Sub
```

EditMode, HourSelectorCmdTarget Property

Syntax EditMode = _Integer

Description Questa proprietà imposta o restituisce la modalità con cui si eseguirà l'input del piano ore. La modifica diventerà effettiva soltanto dopo aver richiamato il metodo "RecalcLayout". Sono validi i seguenti valori:

0 = Tabella Tempi: La visualizzazione classica, con la possibilità di definire gli intervalli di attivazione/disattivazione con una precisione di 15 min.
1 = Griglia: La visualizzazione a griglia con la possibilità di definire degli intervalli di attivazione/disattivazione con precisione al minuto.
2 = Entrambi: Un pulsante si attiverà per passare dalla visualizzazione a Griglia (quella predefinita in apertura), alla visualizzazione Tabella Tempi.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim objHourSelector As HourSelectorCmdTarget  
    If objHourSelector Is Nothing Then Set objHourSelector =  
        GetSynopticObject.GetSubObject("HourSelector1").GetObjectInterface  
    If objHourSelector Is Nothing Then Exit Sub  
  
    Dim sEditMode(0 To 2) As String  
    Dim nSelectItem As Integer  
  
    sEditMode(0) = "Time Table"  
    sEditMode(1) = "Grid"  
    sEditMode(2) = "Both"  
    nSelectItem = ShowPopupMenu(sEditMode, , , )  
    objHourSelector.EditMode = nSelectItem  
    objHourSelector.RecalcLayout  
End Sub
```

EndTimeColText, HourSelectorCmdTarget Property

Syntax EndTimeColText = _String

Description Questa proprietà imposta o restituisce il testo visualizzato nell'intestazione della seconda colonna della griglia. Impostando la proprietà con una stringa vuota verrà visualizzato il testo predefinito. La modifica diventerà effettiva soltanto dopo aver richiamato il metodo "RecalcLayout".

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim objHourSelector As HourSelectorCmdTarget  
    If objHourSelector Is Nothing Then Set objHourSelector =  
        GetSynopticObject.GetSubObject("HourSelector1").GetObjectInterface  
    If objHourSelector Is Nothing Then Exit Sub  
    If objHourSelector.EndTimeColText = "" Then  
        objHourSelector.EndTimeColText = "END"
```

```

Else
    objHourSelector.EndTimeColText = ""
End If
objHourSelector.RecalcLayout
End Sub

```

ErrorString, HourSelectorCmdTarget Property

Syntax ErrorString = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato come stringa di errore durante l'impostazione del piano ore. Durante l'editazione in modo Griglia, se l'utente inserisce un intervallo non corretto, verrà visualizzato come messaggio di errore il testo corrispondente al valore di questa proprietà. Impostando la proprietà con una stringa vuota verrà visualizzato il testo predefinito.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim objHourSelector As HourSelectorCmdTarget
    If objHourSelector Is Nothing Then Set objHourSelector =
        GetSynopticObject.GetSubObject("HourSelector1").GetObjectInterface
    If objHourSelector Is Nothing Then Exit Sub
    If objHourSelector.ErrorString = "" Then
        objHourSelector.ErrorString = "INVALID DATA TIME"
    Else
        objHourSelector.ErrorString = ""
    End If
End Sub

```

ExtSettingsFile, HourSelectorCmdTarget Property

Syntax ExtSettingsFile = _String

Description Questa proprietà imposta o restituisce il file di configurazione esterna per l'oggetto referenziato. Il file può essere specificato anche in fase di design nella proprietà "File di Configurazione" dell'oggetto. L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
    Dim objSymbol As HourSelectorCmdTarget
    Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
    If objSymbol Is Nothing Then Exit Sub

```

```
objSymbol.ExtSettingsFile = "test.xml"
objSymbol.SaveExtSettings
Set objSymbol= Nothing
End Sub
```

GraphicButtons, HourSelectorCmdTarget Property

Syntax GraphicButtons = _Boolean

Description Abilitando questa proprietà i pulsanti della Finestra Scheduler verranno disegnati usando un'icona invece del testo. Il testo opportuno verrà invece visualizzato come tooltip posizionando il mouse sopra ad un pulsante.



Il tooltip non viene gestito nella versione Windows CE.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()
    GraphicButtons = True
    RecalcLayout
End Sub
```

GridModeBtnText, HourSelectorCmdTarget Property

Syntax GridModeBtnText = _String

Description Questa proprietà imposta o restituisce il testo visualizzato nel pulsante "Modalità Griglia". Impostando la proprietà con una stringa vuota verrà visualizzato il testo predefinito. La modifica diventerà effettiva soltanto dopo aver richiamato il metodo "RecalcLayout".

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim objHourSelector As HourSelectorCmdTarget
    If objHourSelector Is Nothing Then Set objHourSelector =
        GetSynopticObject.GetSubObject("HourSelector1").GetObjectInterface
    If objHourSelector Is Nothing Then Exit Sub
```



```

        If objHourSelector.GridModeBtnText = "" Then
            objHourSelector.GridModeBtnText = "GRID MODE (F6)"
        Else
            objHourSelector.GridModeBtnText = ""
        End If
        objHourSelector.RecalcLayout
    End Sub

```

HolidaysBtnText, HourSelectorCmdTarget Property

Syntax HolidaysBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante Festività dello Schedulatore. Se non specificato nulla, Movicon utilizza il testo di default.

Parameter	Description
None	None

Result String

Example:

```

Dim objSelector As HourSelectorCmdTarget
Public Sub Click()
    Debug.Print objSelector.HolidaysBtnText
End Sub
Public Sub SymbolLoading()
    Set objSelector =
        GetSynopticObject.GetSubObject("HourSelector").GetObjectInterface
    objSelector.HolidaysBtnText = "Festivi"
End Sub

```

MaxRow, HourSelectorCmdTarget Property

Syntax MaxRow = _Integer

Description Questa proprietà imposta o restituisce il numero di righe della griglia, corrispondente al numero di fasce orarie programmabili per ogni giorno della settimana. La modifica diventerà effettiva soltanto dopo aver richiamato il metodo "RecalcLayout".

Parameter	Description
None	None

Result Integer

Example:

```

Public Sub Click()
    Dim objHourSelector As HourSelectorCmdTarget
    If objHourSelector Is Nothing Then Set objHourSelector =
        GetSynopticObject.GetSubObject("HourSelector1").GetObjectInterface
    If objHourSelector Is Nothing Then Exit Sub

```

```

Dim sMenuItems(0 To 2) As String
Dim nSelectItem As Integer

sMenuItems(0) = "Reset"
sMenuItems(1) = "Increase"
sMenuItems(2) = "Decrease"
nSelectItem = ShowPopupMenu(sMenuItems, , , )

Select Case nSelectItem
    Case 0 ' Reset
        objHourSelector.MaxRow = 0
    Case 1 ' Increase
        objHourSelector.MaxRow = objHourSelector.MaxRow + 1
    Case 2 ' Decrease
        objHourSelector.MaxRow = objHourSelector.MaxRow - 1
End Select
objHourSelector.RecalcLayout
End Sub

```

NetworkBackupServerName, HourSelectorCmdTarget Property

Syntax NetworkBackupServerName = _String

Description Questa proprietà imposta o restituisce il nome dell'eventuale Server di Networking di Backup usato per prelevare i dati da visualizzare nella Finestra Scheduler quando il Server Primario, ovvero quello impostato nella proprietà "NetowrkServerName", è in timeout.

Parameter	Description
None	None

Result String

Example:

```

Dim objHourSelector As HourSelectorCmdTarget
Public Sub Click()
    Debug.Print objHourSelector.NetworkBackupServerName
End Sub
Public Sub SymbolLoading()
    Set objHourSelector =
    GetSynopticObject.GetSubObject("HourSelector").GetObjectInterface
End Sub

```

NetworkServerName, HourSelectorCmdTarget Property

Syntax NetworkServerName = _String

Description Questa proprietà restituisce il nome dell'eventuale Server di Networking dal quale dovranno essere prelevati i dati da visualizzare nella Finestra Scheduler.

Parameter	Description
None	None

Result String

Example:

```
Dim objSelector As HourSelectorCmdTarget
Public Sub Click()
    Debug.Print objSelector.NetworkServerName
End Sub
Public Sub SymbolLoading()
    Set objSelector =
    GetSynopticObject.GetSubObject("HourSelector").GetObjectInterface
End Sub
```

PromptPad, HourSelectorCmdTarget Property

Syntax PromptPad = _Boolean

Description Se l'HourSelector è in modalità "Grid", impostando la proprietà al valore "True" verrà utilizzato il Pad numerico o alfanumerico per l'editazione dei valori della griglia, impostandola a "False" i valori saranno direttamente editabili nelle celle della griglia.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim objHourSel As HourSelectorCmdTarget
    Set objHourSel =
    GetSynopticObject.GetSubObject("HourSel").GetObjectInterface
    If objHourSel Is Nothing Then Exit Sub
    If objHourSel .PromptPad = False Then
        objHourSel .PromptPad = True
    End If
End Sub
```

RemoveBtnText, HourSelectorCmdTarget Property

Syntax RemoveBtnText = _String

Description Questa proprietà imposta o restituisce il testo visualizzato nel pulsante usato per rimuovere gli schedulatori aggiunti in runtime. Impostando la proprietà con una stringa vuota verrà visualizzato il testo predefinito. La modifica diventerà effettiva soltanto dopo aver richiamato il metodo "RecalcLayout".

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim objHourSelector As HourSelectorCmdTarget  
    If objHourSelector Is Nothing Then Set objHourSelector =  
        GetSynopticObject.GetSubObject("HourSelector1").GetObjectInterface  
    If objHourSelector Is Nothing Then Exit Sub  
    If objHourSelector.RemoveBtnText = "" Then  
        objHourSelector.RemoveBtnText = "REMOVE (CANC)"  
    Else  
        objHourSelector.RemoveBtnText = ""  
    End If  
    objHourSelector.RecalcLayout  
End Sub
```

SaveBtnText, HourSelectorCmdTarget Property

Syntax SaveBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando di salvataggio delle modifiche dei dati di pianificazione dello Schedulatore. Se non specificato nulla, Movicon utilizza il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objSelector As HourSelectorCmdTarget  
Public Sub Click()  
    Debug.Print objSelector.SaveBtnText  
End Sub  
Public Sub SymbolLoading()  
    Set objSelector =  
        GetSynopticObject.GetSubObject("HourSelector").GetObjectInterface  
    objSelector.SaveBtnText= "Salva"  
End Sub
```

Scheduler, HourSelectorCmdTarget Property

Syntax Scheduler = _String

Description Questa proprietà imposta o restituisce il nome dello Schedulatore associato all'oggetto.

Parameter	Description
None	None

Result String

Example:

```
Dim objSelector As HourSelectorCmdTarget
Public Sub Click()
    Debug.Print objSelector.Scheduler
End Sub
Public Sub SymbolLoading()
    Set objSelector =
        GetSynopticObject.GetSubObject("HourSelector").GetObjectInterface
End Sub
```

ShowAddBtn, HourSelectorCmdTarget Property

Syntax ShowAddBtn = _Boolean

Description Questa proprietà imposta o restituisce lo stato di visibilità del pulsante usato per aggiungere gli schedulatori in runtime. La modifica diventerà effettiva soltanto dopo aver richiamato il metodo "RecalcLayout".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim objHourSelector As HourSelectorCmdTarget
    If objHourSelector Is Nothing Then Set objHourSelector =
        GetSynopticObject.GetSubObject("HourSelector1").GetObjectInterface
    If objHourSelector Is Nothing Then Exit Sub
    objHourSelector.ShowAddBtn = Not objHourSelector.ShowAddBtn
    objHourSelector.RecalcLayout
End Sub
```

ShowCancelBtn, HourSelectorCmdTarget Property

Syntax ShowCancelBtn = _Boolean

Description Questa proprietà permette di visualizzare il pulsante di comando per l'operazione di cancellazione delle modifiche effettuate ai dati di pianificazione dello Schedulatore.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objSelector As HourSelectorCmdTarget
```

```

Public Sub Click()
    If Not objSelector Is Nothing Then
        MsgBox "objSelector 's ShowCancelBtn is " &
            objSelector.ShowCancelBtn,vbInformation,GetProjectTitle
        objSelector.ShowCancelBtn= Not objSelector.ShowCancelBtn
        objSelector.RecalcLayout
    Else
        MsgBox "objSelector is nothing",vbInformation,GetProjectTitle
    End If
End Sub
Public Sub SymbolLoading()
    Set objSelector =
        GetSynopticObject.GetSubObject("HourSelector").GetObjectInterface
End Sub

```

ShowColumValue, HourSelectorCmdTarget Property

Syntax ShowColumValue = _Boolean

Description Questa proprietà imposta o restituisce lo stato di visibilità della colonna della griglia usata per impostare il valore di set nel comando usato dallo schedatore. La modifica diventerà effettiva soltanto dopo aver richiamato il metodo "RecalcLayout".

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub Click()
    Dim objHourSelector As HourSelectorCmdTarget
    If objHourSelector Is Nothing Then Set objHourSelector =
        GetSynopticObject.GetSubObject("HourSelector1").GetObjectInterface
    If objHourSelector Is Nothing Then Exit Sub
    objHourSelector.ShowColumValue = Not objHourSelector.ShowColumValue
    objHourSelector.RecalcLayout
End Sub

```

ShowColumVariable, HourSelectorCmdTarget Property

Syntax ShowColumVariable = _Boolean

Description Questa proprietà imposta o restituisce lo stato di visibilità della colonna della griglia usata per impostare il nome della variabile nel comando usato dallo schedatore. La modifica diventerà effettiva soltanto dopo aver richiamato il metodo "RecalcLayout".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objHourSelector As HourSelectorCmdTarget  
    If objHourSelector Is Nothing Then Set objHourSelector =  
        GetSynopticObject.GetSubObject("HourSelector1").GetObjectInterface  
    If objHourSelector Is Nothing Then Exit Sub  
    objHourSelector.ShowColumVariable = Not  
    objHourSelector.ShowColumVariable  
    objHourSelector.RecalcLayout  
End Sub
```

ShowComboScheduler, HourSelectorCmdTarget Property

Syntax ShowComboScheduler = _Boolean

Description Questa proprietà imposta o restituisce lo stato di visibilità della combo box usata per selezionare gli schedulatori in runtime. La modifica diventerà effettiva soltanto dopo aver richiamato il metodo "RecalcLayout".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objHourSelector As HourSelectorCmdTarget  
    If objHourSelector Is Nothing Then Set objHourSelector =  
        GetSynopticObject.GetSubObject("HourSelector1").GetObjectInterface  
    If objHourSelector Is Nothing Then Exit Sub  
    objHourSelector.ShowComboScheduler = Not  
    objHourSelector.ShowComboScheduler  
    objHourSelector.RecalcLayout  
End Sub
```

ShowHolidaysBtn, HourSelectorCmdTarget Property

Syntax ShowHolidaysBtn = _Boolean

Description Questa proprietà permette di visualizzare il pulsante di comando per il passaggio della visualizzazione del piano ore normale a quello festivo e viceversa dello Schedulatore.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objSelector As HourSelectorCmdTarget
Public Sub Click()
    If Not objSelector Is Nothing Then
        MsgBox "objSelector 's ShowHolidaysBtn is " &
            objSelector.ShowHolidaysBtn,vbInformation,GetProjectTitle
        objSelector.ShowHolidaysBtn= Not objSelector.ShowHolidaysBtn
        objSelector.RecalcLayout
    Else
        MsgBox "objSelector is nothing",vbInformation,GetProjectTitle
    End If
End Sub
Public Sub SymbolLoading()
    Set objSelector =
    GetSynopticObject.GetSubObject("HourSelector").GetObjectInterface
End Sub
```

ShowRemoveBtn, HourSelectorCmdTarget Property

Syntax ShowRemoveBtn = _Boolean

Description Questa proprietà imposta o restituisce lo stato di visibilità del pulsante usato per rimuovere gli schedulatori aggiunti in runtime. La modifica diventerà effettiva soltanto dopo aver richiamato il metodo "RecalcLayout".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim objHourSelector As HourSelectorCmdTarget
    If objHourSelector Is Nothing Then Set objHourSelector =
        GetSynopticObject.GetSubObject("HourSelector1").GetObjectInterface
    If objHourSelector Is Nothing Then Exit Sub
    objHourSelector.ShowRemoveBtn = Not objHourSelector.ShowRemoveBtn
    objHourSelector.RecalcLayout
End Sub
```

ShowSaveBtn, HourSelectorCmdTarget Property

Syntax ShowSaveBtn = _Boolean

Description Questa proprietà permette di visualizzare il pulsante di comando per l'operazione di salvataggio delle modifiche effettuate ai dati di pianificazione dello Schedulatore.

Parameter	Description
-----------	-------------

None	None
------	------

Result Boolean

Example:

```
Dim objSelector As HourSelectorCmdTarget
Public Sub Click()
    If Not objSelector Is Nothing Then
        MsgBox "objSelector 's ShowSaveBtn is " &
            objSelector.ShowSaveBtn,vbInformation,GetProjectTitle
        objSelector.ShowSaveBtn= Not objSelector.ShowSaveBtn
        objSelector.RecalcLayout
    Else
        MsgBox "objSelector is nothing",vbInformation,GetProjectTitle
    End If
End Sub
Public Sub SymbolLoading()
    Set objSelector =
    GetSynopticObject.GetSubObject("HourSelector").GetObjectInterface
End Sub
```

StartTimeColText, HourSelectorCmdTarget Property

Syntax StartTimeColText = _String

Description Questa proprietà imposta o restituisce il testo visualizzato nell'intestazione della prima colonna della griglia. Impostando la proprietà con una stringa vuota verrà visualizzato il testo predefinito. La modifica diventerà effettiva soltanto dopo aver richiamato il metodo "RecalcLayout".

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim objHourSelector As HourSelectorCmdTarget
    If objHourSelector Is Nothing Then Set objHourSelector =
    GetSynopticObject.GetSubObject("HourSelector1").GetObjectInterface
    If objHourSelector Is Nothing Then Exit Sub
    If objHourSelector.StartTimeColText = "" Then
        objHourSelector.StartTimeColText = "START"
    Else
        objHourSelector.StartTimeColText = ""
    End If
    objHourSelector.RecalcLayout
End Sub
```

ValueColText, HourSelectorCmdTarget Property

Syntax ValueColText = _String

Description Questa proprietà imposta o restituisce il testo visualizzato nell'intestazione della quarta colonna della griglia. Impostando la proprietà con una stringa vuota verrà visualizzato il testo predefinito. La modifica diventerà effettiva soltanto dopo aver richiamato il metodo "RecalcLayout".

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim objHourSelector As HourSelectorCmdTarget  
    If objHourSelector Is Nothing Then Set objHourSelector =  
        GetSynopticObject.GetSubObject("HourSelector1").GetObjectInterface  
    If objHourSelector Is Nothing Then Exit Sub  
    If objHourSelector.ValueColText = "" Then  
        objHourSelector.ValueColText = "VALUE"  
    Else  
        objHourSelector.ValueColText = ""  
    End If  
    objHourSelector.RecalcLayout  
End Sub
```

VariableColText, HourSelectorCmdTarget Property

Syntax VariableColText = _String

Description Questa proprietà imposta o restituisce il testo visualizzato nell'intestazione della terza colonna della griglia. Impostando la proprietà con una stringa vuota verrà visualizzato il testo predefinito. La modifica diventerà effettiva soltanto dopo aver richiamato il metodo "RecalcLayout".

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim objHourSelector As HourSelectorCmdTarget  
    If objHourSelector Is Nothing Then Set objHourSelector =  
        GetSynopticObject.GetSubObject("HourSelector1").GetObjectInterface  
    If objHourSelector Is Nothing Then Exit Sub  
    If objHourSelector.VariableColText = "" Then  
        objHourSelector.VariableColText = "TAG"  
    Else  
        objHourSelector.VariableColText = ""  
    End If
```

```

objHourSelector.RecalcLayout
End Sub

```

ValueErrorString, HourSelectorCmdTarget Property

Syntax ValueErrorString= _String

Description Questa proprietà imposta o restituisce il messaggio d'errore impostato su un oggetto HourSelector quando si eccedono i limiti min max impostati su una variabile della quale si modifica il valore tramite la cella value dell'HourSelector

Parameter	Description
None	None

Result String

Example:

```

'per leggere il contenuto
Public Sub Click()
    Dim objSelector As HourSelectorCmdTarget
    Set objSelector = GetSynopticObject.GetSubObject("hs").GetObjectInterface
    If Not objSelector Is Nothing Then
        MsgBox "objSelector 's ValueErrorString is " & objSelector.valueerrorstring
    Else
        MsgBox "objSelector is nothing",vbInformation,GetProjectTitle
    End If
End Sub

'per scrivere il contenuto
Public Sub Click()
    Dim objSelector As HourSelectorCmdTarget
    Set objSelector = GetSynopticObject.GetSubObject("hs").GetObjectInterface
    If Not objSelector Is Nothing Then
        objSelector.valueerrorstring = "nuovo valore impostato"
    End If
End Sub

```

1.15.13. IOPortInterface

1.16. Utilizzo dell'IOPortInterface

L'utilizzo dell'interfaccia di programmazione "IOPortInterface" è diretto, cioè non è necessario istanziare un oggetto di tipo "IOPortInterface" ma è possibile utilizzare le sue proprietà e metodi direttamente nel codice VB Script. In pratica tutte le proprietà e funzioni della "IOPortInterface" sono disponibili direttamente dall'Intellisense indipendentemente dal contesto VB Script in cui ci si trova. Quando Movicon utilizza una licenza di tipo "Standard" è possibile comunque utilizzare anche la sintassi che prevede la creazione di un oggetto istanza di tipo "IOPortInterface", al fine di garantire la compatibilità con versioni precedenti del prodotto. Nel caso invece Movicon venga utilizzato con una Licenza di tipo "Basic" l'interfaccia "IOPortInterface" è utilizzabile solo in modo diretto. Ad esempio la seguente sintassi sarà supportata sia utilizzando una licenza "Standard" che una licenza "Basic" di Movicon:

```

Public Sub Click()
    Dim ID As Long

```

```

ID = IOPortOpen("COM1:9600,n,8,1")
MsgBox "PortOpen = " & ID, vbInformation, GetProjectTitle
End Sub

```

mentre la seguente sintassi sarà supportata soltanto utilizzando una licenza "Standard" di Movicon:

```

Public Sub Click()
    Dim objIOPort As IOPortInterface
    Dim ID As Long

    Set objIOPort = GetIOPortInterface
    ID = objIOPort.IOPortOpen("COM1:9600,n,8,1")
    Set objIOPort = Nothing

    MsgBox "PortOpen = " & ID, vbInformation, GetProjectTitle
End Sub

```

Functions

IOClosePort, IOPortInterface Function

Syntax IOClosePort(_IPortID)

Description Esegue la chiusura della porta di comunicazione identificata dal parametro "IPortID".

Il valore di ritorno della funzione può assumere i seguenti significati:

-0: porta di comunicazione chiusa correttamente

-1: identificativo "IPortID" sconosciuto

Parameter	Description
IPortID As Long	Identificativo della porta seriale aperta

Result Long

Example:

```

Sub Main
    Dim bResult As Boolean
    Debug.Print IOClosePort(IPortID)
End Sub

```

IOGetLastError, IOPortInterface Function

Syntax IOGetLastError(_IPortID)

Description Restituisce l'errore di comunicazione più recente. La proprietà non è disponibile in fase di progettazione ed è di sola lettura in fase di esecuzione. Questo metodo non ritorna l'ultimo errore di ritorno da una funzione (come ad esempio -6). Questa ritorna un eventuale errore aggiuntivo che potrebbe aiutare nell'analisi del problema: ad esempio se usando la IOInput non è tornato nessun valore, allora si potrebbe chiamare la IOGetLastError per sapere se c'è stato un errore.

Il valore di ritorno della funzione può assumere i seguenti significati:

0x0001 = Overflow del buffer di ricezione. Spazio esaurito nel buffer di ricezione
 0x0002 = Overrun della porta. L'hardware non ha letto un carattere prima dell'arrivo del successivo e il carattere è andato perduto
 0x0004 = Errore di parità in ricezione. È stato rilevato un errore di parità
 0x0008 = Errore del frame in ricezione. È stato rilevato un errore nel frame di ricezione
 0x0010 = Rilevato lo stato di BREAK
 0x0100 = Buffer di trasmissione pieno. Spazio esaurito nel buffer di trasmissione durante il tentativo di inserimento di un carattere
 0x0200 = Errore di TimeOut Stampante. È stato rilevato un errore di TimeOut con la Stampante
 0x0400 = Errore di I/O Stampante. È stato rilevato un errore della Stampante
 0x0800 = Errore Stampante. Dispositivo non selezionato
 0x1000 = Errore di fine carta Stampante
 0x8000 = Errore modalità richiesta non supportata

Parameter	Description
lPortID As Long	Identificativo della porta seriale aperta.

Result Long

Example:

```

Sub Main
    Dim sVariant As String
    Dim PortID As Long
    PortID = IOPortOpen("COM1:9600,n,8,1")
    MsgBox "IOGetLastError = " & IOGetLastError(PortID), vbInformation,
    GetProjectTitle
End Sub
  
```

IOInBufferCount, IOPortInterface Function

Syntax IOInBufferCount(_lPortID)

Description Restituisce il numero di caratteri in attesa nel buffer di ricezione. Non disponibile in fase di progettazione.
 IOInBufferCount indica il numero di caratteri ricevuti dal modem e memorizzati temporaneamente nel buffer di ricezione. È possibile svuotare il buffer di ricezione impostando la proprietà IOInBufferCount su 0.

Parameter	Description
lPortID As Long	Identificativo della porta seriale aperta

Result Integer

Example:

```

Sub Main
    Dim ID As Long
    ID = IOPortOpen("COM1:9600,n,8,1")
    MsgBox "IOInBufferCount = " & IOInBufferCount(ID), vbInformation,
    GetProjectTitle
    IOInBufferCount(ID) = 0
End Sub
  
```

IOInput, IOPortInterface Function

Syntax IOInput(_IPortID, _bMode)

Description Restituisce e rimuove un flusso di dati dal buffer di ricezione. La proprietà non è disponibile in fase di progettazione ed è di sola lettura in fase di esecuzione. Il parametro bMode determina il modo in cui i dati saranno recuperati:
False = proprietà Input restituisce dati di testo in una variabile Variant
True = proprietà Input restituisce dati binari in una matrice di byte in una variabile Variant

Nota:

a) IOInputLen = 0: IOInput legge il contenuto del buffer di ricezione della seriale per un numero massimo di 255 caratteri. Quindi se il buffer contiene più di 255 caratteri, occorre ripetere la funzione più volte. 'IOInBufferCount' si aggiorna ogni volta che vengono letti dei caratteri dal buffer della seriale con la funzione IOInput
b) IOInputLen <> 0 (max. 255): IOInput preleva il numero di caratteri impostato con la proprietà 'IOInputLen' soltanto se il buffer della seriale contiene un numero uguale o maggiore dei caratteri richiesti; altrimenti ritorna con un variant vuoto (VT_EMPTY).

Parameter	Description
IPortID As Long	Identificativo della porta seriale aperta.
bMode As Boolean	Modalità di recupero dei dati.

Result Long

Example:

```
Sub Main
    Dim i As Integer
    Dim bByte As Variant
    Dim sDebug As String

    bByte = IOInput(PortID, True)
    For i = 0 To UBound(bByte)
        sDebug = sDebug & bByte(i) & ", "
    Next i
    Debug.Print sDebug
End Sub
```

IOOutput, IOPortInterface Function

Syntax IOOutput(_IPortID, _vData)

Description Scrive un flusso di dati nel buffer di trasmissione. La proprietà non è disponibile in fase di progettazione ed è di sola lettura in fase di esecuzione.

La proprietà IOOutput consente di trasmettere dati di testo o dati binari. Per inviare dati di testo con la proprietà IOOutput è necessario specificare una variabile Variant che includa una stringa (la variabile dovrà quindi essere gestita come variabile di tipo stringa). Per inviare dati binari è necessario passare alla proprietà IOOutput una variabile Variant contenente una matrice di byte (la variabile dovrà quindi essere gestita come variabile di tipo array di byte).

Le stringhe ANSI vengono in genere inviate come dati di testo. I dati che includono caratteri di controllo incorporati, quali i caratteri Null, vengono inviati come dati binari, quindi possono essere inviati solo come array di byte.

Parameter	Description
-----------	-------------

IPortID As Long	Identificativo della porta seriale aperta.
vData As Variant	Dato da inviare.

Result Long
0: Funzione eseguita con successo.
-1: Type Mismatch in Parameter. Pass a one-dimensional array
-2: Failed getting the lowest index of the array
-3: Failed getting the highest index of the array
-4: Failed getting array pointer
-5: Generic Error.

Example:

```
Sub Main
    Dim sVariant As String
    sVariant = "ABC"
    Debug.Print IOOutput(PortID, sVariant)
End Sub
```

IOPortOpen, IOPortInterface Function

Syntax IOPortOpen(_IpszPortSettings)

Description Esegue l'apertura della porta di comunicazione impostandola con i parametri contenuti in "IpszPortSettings".

Il valore di ritorno della funzione rappresenta l'identificatore della porta aperta. Questo valore di tipo long è necessario per poter poi utilizzare gli altri metodi e proprietà della IOPortInterface. Può però assumere alcuni significati particolari in caso di errori:

- 1: il numero di caratteri del parametro IpszPortSettings è minore di 4
- 2: non è riuscita l'apertura della porta seriale
- 3: non è stato possibile interpretare le impostazioni della porta seriale (ad esempio baud-rate sbagliato, etc)
- 4: ci sono stati degli errori impostando i valori sulla porta seriale (ad esempio baud-rate, etc)

Es: COM1:9600,n,8,1 dove:
COM1: -> Porta seriale da utilizzare
9600 -> BaudRate
n -> Parità
8 -> Byte Size
1 -> Stop Bits
BaudRate

Rappresenta il baud rate per la comunicazione supportato della seriale. I valori standard sono:

110, 300, 600, 1200, 2400, 4800, 9600, 14400, 19200, 38400, 56000, 57600, 115200, 128000, 256000

Parità

Specifica il sistema di parità da utilizzare. Questo membro può essere uno dei seguenti valori:

- e -> EVENPARITY (Even)
- m -> MARKPARITY (Mark)
- n -> NOPARITY (No parity)
- o -> ODDPARITY (Odd)
- s -> SPACEPARITY (Space)

Byte Size

Rappresenta il numero di bits nei byte trasmessi e ricevuti. 5,6,7 e 8 sono i valori standard.
Stop Bits

 Specifica il numero di bit di stop da utilizzare. Questo membro può essere uno dei seguenti valori:
 0 -> ONESTOPBIT (1 stop bit)
 1 -> ONE5STOPBITS (1.5 stop bits)
 2 -> TWOSTOPBITS (2 stop bits)
 Se si specifica solo "COM1", come impostazioni della seriale verranno prese quelle di default di Windows.

Parameter	Description
lpszPortSettings As String	Settaggi per l'impostazione della porta seriale.

Result Long

Example:

```
Sub Main
    Dim ID As Long
    ID = IOPortOpen("COM1:9600,n,8,1")
    MsgBox "PortOpen = " & ID, vbInformation, GetProjectTitle
End Sub
```

Properties

IOBreak, IOPortInterface Property

Syntax IOBreak(_IPortID)

Description Alza o abbassa il segnale BREAK sulla linea seriale aperta. La proprietà non è disponibile in fase di progettazione.
 Il valore True abilita lo stato di Break, il valore False disabilita lo stato di Break. Per maggiori informazioni sul segnale di BREAK consultare la guida UART a partire dalla 8250 in poi.

Parameter	Description
IPortID As Long	Identificativo della porta seriale aperta

Result Boolean

Example:

```
Sub Main
    objIOPort.IOBreak(PortID) = True
End Sub
```

IOCDHolding, IOPortInterface Property

Syntax IOCDHolding(_IPortID)

Description Determina se è presente una portante, verificando lo stato della linea CD (Carrier Detect). Il segnale CD viene inviato da un modem al computer connesso per indicare che il modem è in linea. La proprietà non è disponibile in fase di progettazione ed è di sola lettura in fase di esecuzione.
 Il valore di ritorno può assumere i valori:

True = la linea CD è attiva
False = la linea CD non è attiva

Nota: Il rilevamento della perdita della portante è particolarmente importante nel caso di un'applicazione host, quale una BBS, in quanto il chiamante può interrompere la comunicazione (perdere la portante) in qualsiasi momento. La condizione CD è anche nota come RLSD (Receive Line Signal Detect).

Parameter	Description
IPortID As Long	Identificativo della porta seriale aperta

Result Boolean

Example:

```
Sub Main
    IOCDHolding(PortID) = True
End Sub
```

IOCTSHolding, IOPortInterface Property

Syntax IOCTSHolding(_IPortID)

Description Determina se è possibile inviare i dati, verificando lo stato della linea CTS (Clear To Send). Il segnale CTS viene, in genere, inviato da un modem al computer connesso, per indicare che è possibile procedere con la trasmissione. La proprietà non è disponibile in fase di progettazione ed è di sola lettura in fase di esecuzione. Il valore di ritorno può assumere i valori:
True = la linea CTS è attiva
False = la linea CTS non è attiva

Nota: La linea CTS viene utilizzata per la sincronia hardware RTS/CTS (Request To Send/Clear To Send). La proprietà IOCTSHolding consente di eseguire manualmente il polling della linea CTS per determinarne lo stato.

Parameter	Description
IPortID As Long	Identificativo della porta seriale aperta

Result Boolean

Example:

```
Sub Main
    Dim bResult As Boolean
    bResult = IOCTSHolding(PortID)
    MsgBox "CTSHolding = " & bResult
End Sub
```

IODSRHolding, IOPortInterface Property

Syntax IODSRHolding(_IPortID)

Description Determina lo stato di una linea DSR (Data Set Ready). Un modem, in genere, invia il segnale DSR al computer al quale è connesso per indicare che è pronto per

l'esecuzione. La proprietà non è disponibile in fase di progettazione ed è di sola lettura in fase di esecuzione.

Il valore di ritorno può assumere i valori:

True = la linea DSR è attiva

False = la linea DSR non è attiva

Nota: Questa proprietà è utile per la scrittura di una routine di sincronia Data Set Ready/Data Terminal Ready per un computer DTE (Data Terminal Equipment)

Parameter	Description
IPortID As Long	Identificativo della porta seriale aperta

Result Boolean

Example:

```
Sub Main
    Dim bResult As Boolean
    bResult = IODSRHolding(PortID)
    MsgBox "DSRHolding=" & bResult
End Sub
```

IODTREnable, IOPortInterface Property

Syntax IODTREnable(_IPortID)

Description Determina se attivare o meno la linea DTR (Data Terminal Ready) durante le comunicazioni. Il segnale DTR è in genere inviato da un computer al rispettivo modem, a indicare che il computer è pronto per ricevere i dati in arrivo. Il valore True abilita la linea DTR, il valore False disabilita la linea DTR.

Nota: Quando DTREnable è impostata su True, la linea DTR è attivata (on) se la porta è aperta e disattivata (off) se la porta è chiusa. Quando IODTREnable è impostata su False, la linea DTR è sempre disattivata. Nella maggior parte dei casi, la disattivazione della linea DTR equivale alla chiusura della comunicazione telefonica.

Parameter	Description
IPortID As Long	Identificativo della porta seriale aperta

Result Boolean

Example:

```
Sub Main
    Dim bResult As Boolean
    IODTREnable(PortID) = True
End Sub
```

IOInputLen, IOPortInterface Property

Syntax IOInputLen(_IPortID)

Description Imposta e restituisce il numero di caratteri che la proprietà Input legge dal buffer di ricezione.

Nota: a) `IOInputLen = 0`: `IOInput` legge il contenuto del buffer di ricezione della seriale per un numero massimo di 255 caratteri. Quindi se il buffer contiene più di 255 caratteri, occorre ripetere la funzione più volte. `IOInBufferCount` si aggiorna ogni volta che vengono letti dei caratteri dal buffer della seriale con la funzione `IOInput`.
b) `IOInputLen <> 0` (max. 255): `IOInput` preleva il numero di caratteri impostato con la proprietà `'IOInputLen'` soltanto se il buffer della seriale contiene un numero uguale o maggiore dei caratteri richiesti; altrimenti ritorna con un variant vuoto (`VT_EMPTY`).

Parameter	Description
IPortID As Long	Identificativo della porta seriale aperta

Result Integer

Example:

```
Sub Main
    Dim nValue As Integer
    Dim PortID As Long
    PortID = IOPortOpen("COM1:9600,n,8,1")
    nValue = Val(InputBox("0 = Max.", "InputLen", "0", 200, 200))
    IOInputLen(PortID) = nValue
End Sub
```

IOOutBufferCount, IOPortInterface Property

Syntax `IOOutBufferCount(_IPortID)`

Description Restituisce il numero di caratteri in attesa nel buffer di trasmissione. Può essere inoltre utilizzata per svuotare il buffer di trasmissione. La proprietà non è disponibile in fase di progettazione.
Per svuotare il buffer di trasmissione, impostare la proprietà `IOOutBufferCount` su 0.

Parameter	Description
IPortID As Long	Identificativo della porta seriale aperta

Result Integer

Example:

```
Sub Main
    MsgBox "IOOutBufferCount = " & IOOutBufferCount(PortID), vbInformation,
    GetProjectTitle
End Sub
```

IORTSEnable, IOPortInterface Property

Syntax `IORTSEnable(_IPortID)`

Description Determina se attivare la linea RTS (Request To Send). Il segnale RTS che richiede l'autorizzazione alla trasmissione di dati è in genere inviato da un computer al

modem collegato. Il valore True attiva la linea RTS, il valore False disattiva la linea RTS.

Nota: Se la proprietà IORTSEnable è impostata su True, la linea RTS è attivata (on) quando la porta è aperta, e disattivata (off) quando la porta è chiusa. La linea RTS viene utilizzata per la sincronia hardware RTS/CTS. La proprietà IORTSEnable consente il polling manuale della linea RTS quando è necessario determinarne lo stato.

Parameter	Description
IPortID As Long	Identificativo della porta seriale aperta

Result Boolean

Example:

Sub Main

IORTSEnable(PortID) = True

End Sub

1.16.1. ListBoxCmdTarget

Events

OnSelected, ListBoxCmdTarget Event

Description Evento notificato ogni volta che è stato selezionato un elemento dalla lista.

Parameter	Description
nSel As Long	Indice dell'elemento selezionato

OnSelecting, ListBoxCmdTarget Event

Description Evento notificato ogni volta che sta per essere selezionato un elemento dalla lista.

Parameter	Description
nSel As Long	Indice dell'elemento selezionato
bRet As boolean	Abilitazione alla selezione

Functions

AddString, ListBoxCmdTarget Function

Syntax AddString(_lpszItem)

Description Questa funzione aggiunge la stringa passata col parametro lpszItem alla lista. La funzione restituisce la posizione della stringa all'interno della lista.

Parameter	Description
lpszItem As String	Stringa da aggiungere nella lista

Result Long

Example:

```
Public Sub Click()  
    Debug.Print GetObjectInterface.AddString("stringa1")  
End Sub
```

GetCount, ListBoxCmdTarget Function

Syntax GetCount()

Description Questa funzione restituisce il numero di stringhe inserite nella lista.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Debug.Print GetObjectInterface.GetCount  
End Sub
```

GetSelectedIndex, ListBoxCmdTarget Function

Syntax GetSelectedIndex()

Description Questa funzione restituisce l'indice della stringa selezionata all'interno della lista.

Parameter	Description
-----------	-------------

None	None
------	------

Result Long

Example:

```
Public Sub Click()
    Debug.Print GetObjectInterface.GetSelectedIndex
End Sub
```

GetText, ListBoxCmdTarget Function

Syntax GetText(_nIndex)

Description Questa funzione restituisce il testo relativo alla posizione all'interno della lista specificata dal parametro nIndex..

Parameter	Description
nIndex As Long	Indice all'interno della lista del testo da reperire

Result String

Example:

```
Public Sub Click()
    Debug.Print GetObjectInterface.GetText(GetObjectInterface.GetCount-1)
End Sub
```

LoadExtSettings, ListBoxCmdTarget Function

Syntax LoadExtSettings

Description Questa funzione permette di caricare la configurazione dell'oggetto dal file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim objSymbol As ListBoxCmdTarget
    Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
    If objSymbol Is Nothing Then Exit Sub
    objSymbol.ExtSettingsFile = "test.sxml"
    objSymbol.LoadExtSettings
    Set objSymbol = Nothing
End Sub
```

RefillList, ListBoxCmdTarget Function

Syntax RefillList()

Description Questa funzione aggiorna il contenuto della lista.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Debug.Print GetObjectInterface.RefillList  
End Sub
```

RemoveString, ListBoxCmdTarget Function

Syntax RemoveString(_IpszItem)

Description Questa funzione cancella la stringa passata col parametro IpszItem dalla lista. La funzione restituisce il valore booleano true se l'operazione è andata a buon fine.

Parameter	Description
IpszItem As String	Stringa da eliminare dalla lista

Result Boolean

Example:

```
Public Sub Click()  
    Debug.Print GetObjectInterface.RemoveString("stringa1")  
End Sub
```

SaveExtSettings, ListBoxCmdTarget Function

Syntax SaveExtSettings

Description Questa funzione permette di salvare la configurazione dell'oggetto nel file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
Dim objSymbol As ListBoxCmdTarget  
Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface  
If objSymbol Is Nothing Then Exit Sub  
objSymbol.ExtSettingsFile = "test.xml"  
objSymbol.SaveExtSettings  
Set objSymbol = Nothing  
End Sub
```

Properties

ExtSettingsFile, ListBoxCmdTarget Property

Syntax ExtSettingsFile = _String

Description Questa proprietà imposta o restituisce il file di configurazione esterna per l'oggetto referenziato. Il file può essere specificato anche in fase di design nella proprietà "File di Configurazione" dell'oggetto. L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
Dim objSymbol As ListBoxCmdTarget  
Set objSymbol =  
GetSynopticObject.GetSubObject("TestObject").GetObjectInterface  
If objSymbol Is Nothing Then Exit Sub  
objSymbol.ExtSettingsFile = "test.xml"  
objSymbol.SaveExtSettings  
Set objSymbol= Nothing  
End Sub
```

ListData, ListBoxCmdTarget Property

Syntax ListData = _String

Description Questa proprietà imposta o restituisce il contenuto del ListBox, ovvero la stringa impostata nella proprietà "Item ListBox" di un oggetto lista o combo box. Se il valore della proprietà "ListData" viene modificato occorre usare il metodo "RefillList" per aggiornare la lista sull'oggetto.

ATTENZIONE: Quando la lista è dinamica, quindi recuperata dalla variabile stringa collegata al campo "Var. Lista ListBox" dell'oggetto, la proprietà "ListData" è in sola lettura e restituisce sempre il contenuto della variabile stringa.

Parameter	Description
-----------	-------------

None	None
------	------

Result String

Example:

```
Public Sub Click()
    Debug.Print GetObjectInterface.ListData
End Sub
```

ListVariable, ListBoxCmdTarget Property

Syntax ListVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile collegata alla lista dei valori inseriti nel ListBox di Movicon.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Debug.Print GetObjectInterface.ListVariable
End Sub
```

SortItems,ListBoxCmdTarget Property

Syntax SortItems = _String

Description Consente di impostare o leggere la proprietà che determina l'ordinamento degli elementi presenti nell'oggetto.

Parameter	Description
None	None

Result String

Example1:

'Utilizzo di un oggetto Combo-Box:

```
Dim objCombo As DrawCmdTarget
Dim objDisplay As DisplayEditCmdTarget
Dim objList As ListBoxCmdTarget
Public Sub Click()
    Set objCombo=GetSynopticObject.GetSubObject("combobox")
    Set objDisplay=objCombo.GetObjectInterface()
    Set objList=objDisplay.GetComboListInterface
    objList.SortItems=True
    objList.RefillList
```

```

        Set objCombo=Nothing
        Set objDisplay=Nothing
        Set objList=Nothing
    End Sub

```

Example2:

'Utilizzo di un oggetto List-Box (scrittura):

```

Dim objListBox As ListBoxCmdTarget
Public Sub Click()
    Set objListBox=GetSynopticObject.GetSubObject("listbox").GetObjectInterface
    objListBox.SortItems=True
    objListBox.RefillList
    Set objListBox=Nothing
End Sub

```

Example3:

'Utilizzo di un oggetto List-Box (lettura):

```

Dim objListBox As ListBoxCmdTarget
Public Sub Click()
    Set objListBox=GetSynopticObject.GetSubObject("listbox").GetObjectInterface
    Dim sorted As Boolean
    sorted=objListBox.SortItems
    debug.print "La proprietà SortItems è " & sorted
    Set objListBox=Nothing
End Sub

```

Variable, ListBoxCmdTarget Property

Syntax Variable = _String

Description Questa proprietà imposta o restituisce il nome della variabile collegata all'item selezionato del ListBox di Movicon.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Debug.Print GetObjectInterface.Variable
End Sub

```

1.16.2. NetworkClientCmd

Functions

ConnectVariable, NetworkClientCmd Function

Syntax ConnectVariable(_IpszVarName, _IpszServerName, _IpszServerVarName, _nMode,_bSynchronous)

Description Questa funzione permette di collegare una variabile lato client ad una variabile sul server.

Le modalità di connessione sono:

- 0 input
- 1 output
- 2 input/output

Parameter	Description
IpszVarName As String	Nome della variabile da collegare
IpszServerName As String	Nome del Server a cui collegarsi
IpszServerVarName As String	Nome della variabile lato Server
nMode As Integer	Modalità di connessione
bSynchronous as boolean	Consente di decidere se connettere la variabile in modo sincrono a asincrono (default = True) . bSynchronous = True : la funzione ritorna quando la variabile è stata connessa al server ed è stato letto il valore. Quindi per ogni variabile da collegare in input/output o input, occorre attendere un tempo pari al timeout impostato sul client. bSynchronous = False : La funzione connette la variabile al server e ritorna subito senza aspettare che la variabile contenga un valore valido.

Result Boolean

Example:

```
Public Sub Click()  
    Dim NetwObj As NetworkClientCmd  
    Set NetwObj = GetNetworkClient  
    If Not NetwObj Is Nothing Then  
        NetwObj.ConnectVariable("VAR00001", "Server2", "VAR00005", 0,  
        True)  
        Set NetwObj = Nothing  
    End If  
End Sub
```

DisconnectVariable, NetworkClientCmd Function

Syntax DisconnectVariable(_IpszVarName)

Description Questa funzione permette di scollegare una variabile lato client dal server.

Parameter	Description
IpszVarName As String	Nome della variabile da collegare

Result Boolean

Example:

```
Public Sub Click()  
    Dim NetwObj As NetworkClientCmd
```

```

Set NetwObj = GetNetworkClient
If Not NetwObj Is Nothing Then
    NetwObj.DisconnectVariable("VAR00001")
Set NetwObj = Nothing
End If
End Sub

```

GetClientRules, NetworkClientCmd Function

Syntax GetClientRules(_IpszClientRulesName)

Description Questa funzione permette di reperire l'oggetto ClientRules, facente parte della ClientRulesInterface, referenziato dal nome passato come parametro. Se come parametro viene passata la stringa vuota, l'oggetto farà riferimento alle proprietà generali del Network Client. Se invece contiene il nome di un Profilo Client valido, l'oggetto farà riferimento alle proprietà generali dello stesso.

Parameter	Description
IpszClientRulesName As String	Nome dell'oggetto ClientRules da reperire

Result Object
Viene restituito un oggetto di tipo ClientRulesInterface se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```

Public Sub Click()
    Dim NetwObj As NetworkClientCmd
    Dim ClientObj As ClientRulesInterface
    Set NetwObj = GetNetworkClient
    If Not NetwObj Is Nothing Then
        Set ClientObj = NetwObj.GetClientRules("Server2")
        If Not ClientObj Is Nothing Then
            Debug.Print ClientObj.ClientTimeout
            Set ClientObj = Nothing
        End If
        Set NetwObj = Nothing
    End If
End Sub

```

GetRASStation, NetworkClientCmd Function

Syntax GetRASStation(_IpszRASStation)

Description Questa funzione permette di reperire l'oggetto RASStation, facente parte della RASStationInterface, referenziato dal nome passato come parametro.

Parameter	Description
IpszRASStation As String	Nome dell'oggetto RASStation da reperire

Result Object
Viene restituito un oggetto di tipo RASStationInterface se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```

Public Sub Click()
    Dim NetwObj As NetworkClientCmd
    Dim RASObj As RASStationInterface
    Set NetwObj = GetNetworkClient

```

```

        If Not NetwObj Is Nothing Then
            Set RASSObj = NetwObj.GetRASStation("RAS1")
            If Not RASSObj Is Nothing Then
                Debug.Print RASStation.IsConnected
                Set RASSObj = Nothing
            End If
            Set NetwObj = Nothing
        End If
    End Sub

```

IsServerAvailable, NetworkClientCmd Function

Syntax IsServerAvailable(_IpszServerName)

Description Questa funzione verifica se il server, il cui nome viene passato come parametro, è disponibile per la condivisione delle variabili.

Parameter	Description
IpszServerName As String	Nome del server da testare

Result Boolean

Example:

```

Public Sub Click()
    Dim NetwObj As NetworkClientCmd
    Set NetwObj = GetNetworkClient
    If Not NetwObj Is Nothing Then
        L$ = InputBox$("Enter server name:", "", "Server2")
        MsgBox "NetwObj.IsAvailable = " &
            NetwObj.IsServerAvailable(L), vbOkOnly, ""
        Set NetwObj = Nothing
    End If
End Sub

```

1.16.3. NetworkRedudancyCmd

Functions

ActNumRetries, NetworkRedudancyCmd Function

Syntax ActNumRetries()

Description Questa funzione restituisce l'attuale numero di tentativi di connessione da parte del Secondario verso il primario. Lo script è da eseguire sul Secondario: quando il primario cade, il Secondario effettua un n° di tentativi pari al valore inserito nella Proprietà "# Retries", e questo metodo restituisce il n° del tentativo di connessione in corso.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim ObjRed As NetworkRedudancyCmd  
    Set ObjRed = GetNetworkRedudancy  
    If ObjRed is Nothing Then Exit Sub  
    MsgBox "ActNumRetries -> " & ObjRed.ActNumRetries  
End Sub
```

CallBackServer, NetworkRedudancyCmd Function

Syntax CallBackServer()

Description Questa funzione restituisce ulteriori informazioni riguardo al server al quale è connesso il Secondario (Ip , numero porta, etc..).

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim ObjRed As NetworkRedudancyCmd  
    Set ObjRed = GetNetworkRedudancy  
    If ObjRed is Nothing Then Exit Sub  
    MsgBox "CallBackServer -> " & ObjRed.CallBackServer  
End Sub
```

ConnectToServer, NetworkRedudancyCmd Function

Syntax ConnectToServer()

Description Questa funzione restituisce l'Indirizzo IP del Server Primario al quale è connesso il Secondario.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim ObjRed As NetworkRedudancyCmd  
    Set ObjRed = GetNetworkRedudancy  
    If ObjRed is Nothing Then Exit Sub  
    MsgBox "PrimaryServer IP -> " & ObjRed.ConnectToServer  
End Sub
```

GetTotalPendingMessage, NetworkRedudancyCmd Function

Syntax GetTotalPendingMessage()

Description Questa funzione restituisce l'attuale numero di messaggi in attesa di essere inviati tra i 2 server. Lo script si può eseguire su entrambi i Server, ognuno dei quali restituirà il valore dei propri messaggi in attesa di spedizione.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim ObjRed As NetworkRedudancyCmd  
    Set ObjRed = GetNetworkRedudancy  
    If ObjRed is Nothing Then Exit Sub  
    MsgBox "GetTotalPendingMessage -> " & ObjRed.GetTotalPendingMessages  
End Sub
```

IsActiveServer, NetworkRedudancyCmd Function

Syntax IsActiveServer()

Description Questa funzione restituisce il valore booleano True o False a seconda del fatto che il Server su cui viene eseguito lo script, sia o meno il Server Attivo.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim ObjRed As NetworkRedudancyCmd  
    Set ObjRed = GetNetworkRedudancy  
    If ObjRed is Nothing Then Exit Sub  
    MsgBox "IsActiveServer -> " & ObjRed.IsActiveServer  
End Sub
```

LastInteractionTime, NetworkRedudancyCmd Function

Syntax LastInteractionTime()

Description Questa funzione restituisce la Data e l'ora dell'ultima interazione avvenuta tra i 2 Server; lo script può essere eseguito su entrambi i Server.

Parameter	Description
None	None

Result Date

Example:

```
Public Sub Click()  
    Dim ObjRed As NetworkRedudancyCmd  
    Set ObjRed = GetNetworkRedudancy  
    If ObjRed is Nothing Then Exit Sub  
    MsgBox "LastInteractionTime -> " & ObjRed.LastInteractionTime  
End Sub
```

PendingStartedDriverOnSecondary, NetworkRedudancyCmd Function

Syntax PendingStartedDriverOnSecondary()

Description Questa funzione restituisce il valore booleano True se i Driver di Comunicazione sono in attesa di essere attivati sul Server Secondario.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim ObjRed As NetworkRedudancyCmd  
    Set ObjRed = GetNetworkRedudancy  
    If ObjRed is Nothing Then Exit Sub  
    MsgBox "PendingStartedDriverOnSecondary -> " &  
        ObjRed.PendingStartedDriverOnSecondary  
End Sub
```

SecondaryServerConnected, NetworkRedudancyCmd Function

Syntax SecondaryServerConnected()

Description Questa funzione restituisce il valore booleano True se il server secondario è connesso al primario.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim ObjRed As NetworkRedudancyCmd  
    Set ObjRed = GetNetworkRedudancy  
    If ObjRed is Nothing Then Exit Sub  
    MsgBox "SecondaryServerConnected -> " &  
        ObjRed.SecondaryServerConnected  
End Sub
```

StartedDriverOnSecondary, NetworkRedudancyCmd Function

Syntax StartedDriverOnSecondary()

Description Questa funzione restituisce il valore booleano True se i Driver di comunicazione sono attivi sul secondario.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim ObjRed As NetworkRedudancyCmd  
    Set ObjRed = GetNetworkRedudancy  
    If ObjRed is Nothing Then Exit Sub  
    MsgBox "StartedDriverOnSecondary -> " &  
        ObjRed.StartedDriverOnSecondary  
End Sub
```

StatusVariable, NetworkRedudancyCmd Function

Syntax StatusVariable()

Description Questa funzione restituisce il nome della variabile impostata come variabile di stato della ridondanza.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim ObjRed As NetworkRedudancyCmd  
    Set ObjRed = GetNetworkRedudancy  
    If ObjRed is Nothing Then Exit Sub  
    MsgBox "StatusVariable Name -> " & ObjRed.StatusVariable
```

End Sub

Synchronizing, NetworkRedudancyCmd Function

Syntax Synchronizing()

Description Questa funzione restituisce il valore booleano True se i server sono in fase di sincronizzazione. Se i server non sono sincronizzati o hanno già effettuato la sincronizzazione, il risultato è False.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim ObjRed As NetworkRedudancyCmd  
    Set ObjRed = GetNetworkRedudancy  
    If ObjRed Is Nothing Then Exit Sub  
    MsgBox "Synchronizing -> " & ObjRed.Synchronizing  
End Sub
```

Type, NetworkRedudancyCmd Function

Syntax Type()

Description Questa funzione restituisce un numero che indica il tipo di server.

I valori possibili sono:

0 none
1 primary
2 secondary

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim ObjRed As NetworkRedudancyCmd  
    Dim StrType As String  
    Set ObjRed = GetNetworkRedudancy  
    If ObjRed Is Nothing Then Exit Sub  
    Select Case ObjRed.Type  
        Case 0  
            StrType= "None"  
        Case 1  
            StrType= "Primary"  
        Case 2  
            StrType= "Secondary"  
        Case Else
```

```

End Select
MsgBox "ServerType-> " & StrType
End Sub

```

Properties

DriverErrorTimeout, NetworkRedudancyCmd Property

Syntax DriverErrorTimeout = _Long

Description Questa proprietà permette di impostare un valore di timeout di errore (in ms) dei Driver di comunicazione.

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
    Dim ObjRed As NetworkRedudancyCmd
    Dim StrValue As String
    Set ObjRed = GetNetworkRedudancy
    If ObjRed Is Nothing Then Exit Sub
    StrValue = InputBox "Insert DriverErrorTimeout", "Input Value"
    ObjRed.DriverErrorTimeout = Val(StrValue)
    MsgBox "DriverErrorTimeout -> " & ObjRed.DriverErrorTimeout
End Sub

```

MaxHisCacheHits, NetworkRedudancyCmd Property

Syntax MaxHisCacheHits = _Long

Description Questa proprietà imposta o restituisce il numero massimo di registrazioni dopo le quali vengono eliminati i record.

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
    Dim ObjRed As NetworkRedudancyCmd
    Dim StrValue As String
    Set ObjRed = GetNetworkRedudancy
    If ObjRed Is Nothing Then Exit Sub
    StrValue = InputBox "Insert MaxHisCacheHit", "Input Value"
    ObjRed.MaxHisCacheHits = Val(StrValue)

```

```

MsgBox "MaxHisCacheHit -> " & ObjRed.MaxHisCacheHits
End Sub

```

Retries, NetworkRedudancyCmd Property

Syntax Retries = _Long

Description Questa proprietà imposta o restituisce il numero di tentativi di riconnessione del secondario in caso di "caduta" del primario.

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
    Dim ObjRed As NetworkRedudancyCmd
    Dim StrValue As String
    Set ObjRed = GetNetworkRedudancy
    If ObjRed Is Nothing Then Exit Sub
    StrValue = InputBox "Insert Retries", "Input Value"
    ObjRed.Retries = Val(StrValue)
    MsgBox "Retries -> " & ObjRed.Retries
End Sub

```

SwitchServerOnDriverError, NetworkRedudancyCmd Property

Syntax SwitchServerOnDriverError = _Boolean

Description Questa proprietà permettere al secondario (se impostata al valore booleano True) di attivare i driver di comunicazione in caso di fail da parte del primario.

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub Click()
    Dim ObjRed As NetworkRedudancyCmd
    Dim StrValue As String
    Set ObjRed = GetNetworkRedudancy
    If ObjRed Is Nothing Then Exit Sub
    ObjRed.SwitchServerOnDriverError = Not
    ObjRed.SwitchServerOnDriverError
    MsgBox "SwitchServerOnDriverError -> " &
    ObjRed.SwitchServerOnDriverError
End Sub

```

SyncTimeFreq, NetworkRedudancyCmd Property

Syntax SyncTimeFreq = _Long

Description Questa proprietà imposta o restituisce il numero di sincronizzazioni necessarie fra i due server nell'arco delle 24 ore.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim ObjRed As NetworkRedudancyCmd  
    Dim StrValue As String  
    Set ObjRed = GetNetworkRedudancy  
    If ObjRed Is Nothing Then Exit Sub  
    StrValue = InputBox "Insert SyncTimeFreq ", "Input Value"  
    ObjRed.SyncTimeFreq = Val(StrValue)  
    MsgBox "Retries -> " & ObjRed.SyncTimeFreq  
End Sub
```

TimeOut, NetworkRedudancyCmd Property

Syntax TimeOut = _Long

Description Questa proprietà imposta o restituisce il tempo di timeout (in ms) che il Secondario deve attendere prima di diventare attivo in caso di "caduta" del primario.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim ObjRed As NetworkRedudancyCmd  
    Dim StrValue As String  
    Set ObjRed = GetNetworkRedudancy  
    If ObjRed Is Nothing Then Exit Sub  
    StrValue = InputBox "Insert TimeOut", "Input Value"  
    ObjRed.TimeOut = Val(StrValue)  
    MsgBox "Retries -> " & ObjRed.TimeOut  
End Sub
```

1.16.4. OPCAECmdTarget

Functions

GetEventInQueueAt, OPCAECmdTarget Function

Syntax GetEventInQueueAt(_nIndex)

Description Funzione al momento non supportata.

Parameter	Description
nIndex As Integer	Indice dell'evento i-esimo

Result String

GetNumEventInQueue, OPCAECmdTarget Function

Syntax GetNumEventInQueue()

Description Funzione al momento non supportata.

Parameter	Description
None	None

Result Long

GetNumObjectsInHeap, OPCAECmdTarget Function

Syntax GetNumObjectsInHeap()

Description Funzione al momento non supportata.

Parameter	Description
None	None

Result Long

GetServerStatus, OPCAECCmdTarget Function

Syntax GetServerStatus()

Description Funzione al momento non supportata.

Parameter	Description
None	None

Result Long

GetServerVendorInfo, OPCAECCmdTarget Function

Syntax GetServerVendorInfo()

Description Funzione al momento non supportata.

Parameter	Description
None	None

Result String

GetXMLSettings, OPCAECCmdTarget Function

Syntax GetXMLSettings()

Description Funzione al momento non supportata.

Parameter	Description
None	None

Result String

Reconnect, OPCAECCmdTarget Function

Syntax Reconnect()

Description Funzione al momento non supportata.

Parameter	Description
None	None

Result Boolean

Refresh, OPCAECmdTarget Function

Syntax Refresh()

Description Funzione al momento non supportata.

Parameter	Description
None	None

Result Boolean

Properties

MaxEventQueue, OPCAECmdTarget Property

Syntax MaxEventQueue = _Long

Description Proprietà al momento non supportata.

Parameter	Description
None	None

Result Long

Node, OPCAECmdTarget Property

Syntax Node = _Long

Description Proprietà al momento non supportata.

Parameter	Description

None	None
------	------

Result String

ReconnectTime, OPCAECCmdTarget Property

Syntax ReconnectTime = _Long

Description Proprietà al momento non supportata.

Parameter	Description
None	None

Result Long

Server, OPCAECCmdTarget Property

Syntax Server = _Long

Description Proprietà al momento non supportata.

Parameter	Description
None	None

Result String

1.16.5. OPCClientCmdTarget

Functions

ClearDynOPCItemTypeCache, OPCClientCmdTarget Function

Syntax ClearDynOPCItemTypeCache

Description Questa funzione consente di pulire la cache di memoria per la gestione delle variabili dinamiche in uso.

Parameter	Description
-----------	-------------

None	None
------	------

Result Long

Example:

```
Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim vResult As Long
    Set objOPC = GetOPCClient
    vResult = objOPC.ClearDynOPCItemTypeCache
    MsgBox "ClearDynOPCItemTypeCache = " &
    vResult,vbInformation,GetProjectTitle
    Set objOPC = Nothing
End Sub
```

DisableDynOPCGroup,OPCClientCmdTarget Function

Syntax DisableDynOPCGroup (IpszNodeName, IpszServerName, IpszGroupName, bWait)

Description Disabilita il Gruppo OPC associato ad una Variabile collegata attraverso la proprietà "Indirizzo Fisico I/O" (Link Dinamico) ad un Item OPC e sulla quale è impostata la proprietà "Nome Gruppo OPC" col nome del Gruppo OPC presente nella risorsa "OPC Client DA"



se la variabile è associata ad un Item OPC il metodo "DisableDynOPCGroup" non ha significato, deve essere utilizzato invece il metodo EnableGroup

Parameter	Description
IpszNodeName As String	Nome del PC sul quale è attivo l'OPC Server;se non specificato è considerato di default il PC locale
IpszServerName As String	Nome del Server OPC
bWait As Boolean	True o False per indicare se l'abilitazione è eseguita in modo sincrono o asincrono

Result Boolean

Example:

```
Option Explicit
Const OPC_SERVER As String = "Softing.OPCToolboxDemo_ServerDA.1"
Const OPC_GROUP As String = "Group1"
Const OPC_PC_NODE As String = ""
Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget
    Dim bRet As Boolean
    Set objOPC = GetOPCClient
    If Not objOPC Is Nothing Then
        Set objOPCDA = objOPC.GetOPCDAClientObject(OPC_SERVER)
        If Not objOPCDA Is Nothing Then
            bRet =
            objOPC.DisableDynOPCGroup(OPC_PC_NODE,OPC_SERVER,OPC_GROUP,True)
            Debug.Print "Dynamic - OPC Group " & OPC_GROUP & " Disable:" & CStr(bRet)
        End If
    End If
    Set objOPCDA = Nothing
    Set objOPC = Nothing
```

End Sub

EnableDynOPCGroup, OPCClientCmdTarget Function

Syntax EnableDynOPCGroup (IpszNodeName, IpszServerName, IpszGroupName, bWait)

Description Abilita il Gruppo OPC associato ad una Variabile collegata attraverso la proprietà "Indirizzo Fisico I/O" (Link Dinamico) ad un Item OPC e sulla quale è impostata la proprietà "Nome Gruppo OPC" col nome del Gruppo OPC presente nella risorsa "OPC Client DA"



se la variabile è associata ad un Item OPC il metodo "EnableDynOPCGroup" non ha significato, deve essere utilizzato invece il metodo EnableGroup

Parameter	Description
IpszNodeName As String	Nome del PC sul quale è attivo l'OPC Server; se non specificato è considerato di default il PC locale
IpszServerName As String	
bWait As Boolean	
	Nome del Server OPC
	True o False per indicare se l'abilitazione è eseguita in modo sincrono o asincrono

Result Boolean

Example:

```
Option Explicit
Const OPC_SERVER As String = "Softing.OPCToolboxDemo_ServerDA.1"
Const OPC_GROUP As String = "Group1"
Const OPC_PC_NODE As String = ""
Public Sub Click()
Dim objOPC As OPCClientCmdTarget
Dim objOPCDA As OPCClientObjCmdTarget
Dim bRet As Boolean
Set objOPC = GetOPCClient
If Not objOPC Is Nothing Then
Set objOPCDA = objOPC.GetOPCDAClientObject(OPC_SERVER)
If Not objOPCDA Is Nothing Then
bRet =
objOPC.EnableDynOPCGroup(OPC_PC_NODE, OPC_SERVER, OPC_GROUP, True)
Debug.Print "Dynamic - OPC Group " & OPC_GROUP & " Enable:" & CStr(bRet)
End If
End If
Set objOPCDA = Nothing
Set objOPC = Nothing
End Sub
```

GetOPCAEClientObject, OPCClientCmdTarget Function

Syntax GetOPCAEClientObject(_IpszServerName)

Description La funzione restituisce un oggetto di tipo OPAECmdTarget attraverso il quale è possibile gestire le proprietà e i metodi relativi alla comunicazione dello standard OPC Alarm Event.



Questa funzione non è supportata in Windows CE.(se utilizzata restituisce sempre 'null')

Parameter	Description
IpszServerName As String	Nome di registrazione del server OPC

Result Object
Viene restituito un oggetto di tipo OPAECmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCAE As OPCAECmdTarget  
    Dim vResult As Long  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCAE = objOPC.GetOPCAEClientObject("Softing OPC Toolbox Demo  
    OPC AE Server")  
    If objOPCAE Is Nothing Then Exit Sub  
    MsgBox "OPCAE_XMLSettings are: " &  
    objOPCAE.GetXMLSettings,vbInformation,GetProjectTitle  
    Set objOPC = Nothing  
End Sub
```

GetOPCDAClientObject, OPCClientCmdTarget Function

Syntax GetOPCDAClientObject(_IpszServerName)

Description La funzione restituisce un oggetto di tipo OPCClientObjCmdTarget attraverso il quale è possibile gestire le proprietà e i metodi relativi alla comunicazione dello standard OPC Data Access.

Parameter	Description
IpszServerName As String	Nome di registrazione del server OPC

Result Object
Viene restituito un oggetto di tipo OPCClientObjCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim objOPCAE As OPCAECmdTarget  
    Dim vResult As Long  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
  
    Set objOPCAE = objOPC.GetOPCAEClientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If Not objOPCAE Is Nothing Then  
        MsgBox "objOPCAE_XMLSettings are: " &  
        objOPCAE.GetXMLSettings,vbInformation,GetProjectTitle  
        Set objOPCAE = Nothing  
    End If
```

```

Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo
OPC DA Server")
If objOPCDA Is Nothing Then Exit Sub
MsgBox "OPCDA_XMLSettings are: " &
objOPCDA.GetXMLSettings,vbInformation,GetProjectTitle

Set objOPCDA = Nothing
Set objOPC = Nothing
End Sub

```

GetXMLSettings, OPCClientCmdTarget Function

Syntax GetXMLSettings

Description La funzione restituisce la stringa di definizione dell'oggetto OPCClient in formato XML.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim vResult As String
    Set objOPC = GetOPCClient
    vResult = objOPC.GetXMLSettings
    MsgBox "XMLSettings are: " & vResult,vbInformation,GetProjectTitle
    Set objOPC = Nothing
End Sub

```

ReconnectAll, OPCClientCmdTarget Function

Syntax ReconnectAll(__bWait)

Description La funzione abilita o disabilita il tempo di attesa necessario per il Client prima della riconnessione al Server nel caso di perdita di comunicazione.

Parameter	Description
bWait As Boolean	abilitazione per l'attesa prima della riconnessione al server

Result Boolean

Example:

```

Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim bResult As Boolean
    Set objOPC = GetOPCClient
    vResult = objOPC.ReconnectAll(True)
    MsgBox "ReconnectAll = " & bResult,vbInformation,GetProjectTitle
    Set objOPC = Nothing
End Sub

```

Properties

PoolOPCClientTest, OPCClientCmdTarget Property

Syntax PoolOPCClientTest = _Long

Description Questa proprietà imposta o restituisce il numero di Threads da usare per il processo di test degli Item OPC al momento della connessione al Server.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    MsgBox "PoolOPCClientTest = " &  
        objOPC.PoolOPCClientTest,vbInformation,GetProjectTitle  
    Set objOPC = Nothing  
End Sub
```

RefreshRateDynamicOPC, OPCClientCmdTarget Property

Syntax RefreshRateDynamicOPC = _Long

Description Questa proprietà imposta o restituisce il tempo di rinfresco per gli Item OPC dinamici del progetto.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    MsgBox "RefreshRateDynamicOPC = " &  
        objOPC.RefreshRateDynamicOPC,vbInformation,GetProjectTitle  
    Set objOPC = Nothing  
End Sub
```

StartupTimeout, OPCClientCmdTarget Property

Syntax StartupTimeout = _Long

Description Questa proprietà imposta o restituisce il tempo di timeout, espresso in msec, per il processo di inizializzazione degli Item OPC al momento della connessione al Server.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    MsgBox " StartupTimeout = " & objOPC.  
StartupTimeout,vbInformation,GetProjectTitle  
    Set objOPC = Nothing  
End Sub
```

TimeoutDynamicOperation, OPCClientCmdTarget Property

Syntax TimeoutDynamicOperation = _Long

Description Questa proprietà imposta o restituisce il tempo di Timeout, espresso in millisecondi, per l'operazione di inserimento di un nuovo Item OPC di tipo dinamico.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    MsgBox "TimeoutDynamicOperation = " &  
objOPC.TimeoutDynamicOperation,vbInformation,GetProjectTitle  
    Set objOPC = Nothing  
End Sub
```

TimeoutOPCClientTest, OPCClientCmdTarget Property

Syntax TimeoutOPCClientTest = _Long

Description Questa proprietà imposta o restituisce il tempo di Timeout, espresso in millisecondi, per l'operazione di test di un Item OPC al momento della connessione al Server.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    MsgBox "TimeoutOPCClientTest= " &  
        objOPC.TimeoutOPCClientTest,vbInformation,GetProjectTitle  
    Set objOPC = Nothing  
End Sub
```

1.16.6. OPCClientGroupObjCmdTarget

Functions

ConnectGroup, OPCClientGroupObjCmdTarget Function

Syntax ConnectGroup(_bWait)

Description Questa funzione consente di forzare la connessione relativa agli Item del gruppo referenziato. Il parametro booleano specifica se Movicon deve attendere il tempo di riconnessione prima di effettuare il collegamento.

Parameter	Description
bWait As Boolean	Abilitazione all'attesa del tempo di riconnessione.

Result Boolean

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set GroupOne = objOPCDA.GetGroupObject("Group one")  
    GroupOne.ConnectGroup(False)  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
    Set GroupOne = Nothing  
End Sub
```


GetItemObject, OPCClientGroupObjCmdTarget Function

Syntax GetItemObject(_lpszItemName)

Description Questa funzione permette di reperire l'oggetto di tipo OPCClientItemObjCmdTarget relativo all'item referenziato.

Parameter	Description
lpszItemName As String	Nome dell'item di cui si intende reperire l'oggetto.

Result Object
Viene restituito un oggetto di tipo OPCClientItemObjCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget  
    Dim ItemOne As OPCClientItemObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAlientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set GroupOne = objOPCDA.GetGroupObject("Group one")  
    If GroupOne Is Nothing Then Exit Sub  
    Set ItemOne = GroupOne.GetItemObject("Var00001")  
    If ItemOne Is Nothing Then Exit Sub  
    MsgBox "ItemOne ItemID is" & ItemOne.ItemID,vbInformation,GetProjectTitle  
    Set ItemOne = Nothing  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
    Set GroupOne = Nothing  
End Sub
```

GetName, OPCClientGroupObjCmdTarget Function

Syntax GetName()

Description Questa funzione restituisce il nome del Gruppo referenziato.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget
```

```

Dim ItemOne As OPCClientItemObjCmdTarget

Set objOPC = GetOPCClient
If objOPC Is Nothing Then Exit Sub
Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
DA Server")
If objOPCDA Is Nothing Then Exit Sub
Set GroupOne = objOPCDA.GetGroupObject("Group one")
If GroupOne Is Nothing Then Exit Sub
    MsgBox "GroupOne Name is" &
        GroupOne.GetName,vbInformation,GetProjectTitle
Set ItemOne = Nothing
Set objOPCDA = Nothing
Set objOPC = Nothing
Set GroupOne = Nothing
End Sub

```

GetNumObjectsInHeap, OPCClientGroupObjCmdTarget Function

Syntax GetNumObjectsInHeap

Description Questa funzione restituisce il numero di oggetti caricati in memoria.

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget
    Dim GroupOne As OPCClientGroupObjCmdTarget

    Set objOPC = GetOPCClient
    If objOPC Is Nothing Then Exit Sub
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
    DA Server")
    If objOPCDA Is Nothing Then Exit Sub
    Set GroupOne = objOPCDA.GetGroupObject("Group one")
    If GroupOne Is Nothing Then Exit Sub
    MsgBox "NumObjectsInHeap are " &
        GroupOne.GetNumObjectsInHeap,vbInformation,GetProjectTitle
    Set objOPCDA = Nothing
    Set objOPC = Nothing
    Set GroupOne = Nothing
End Sub

```

GetServerObject, OPCClientGroupObjCmdTarget Function

Syntax GetServerObject()

Description Questa funzione restituisce l'oggetto di tipo OPCClientObjCmdTarget relativo al server di appartenenza del Gruppo referenziato.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo OPCClientObjCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget
    Dim GroupOne As OPCClientGroupObjCmdTarget

    Set objOPC = GetOPCClient
    If objOPC Is Nothing Then Exit Sub
    Set objOPCDA = objOPC.GetOPCDAlientObject("Softing OPC Toolbox Demo OPC
    DA Server")
    If objOPCDA Is Nothing Then Exit Sub
    Set GroupOne = objOPCDA.GetGroupObject("Group one")
    If GroupOne Is Nothing Then Exit Sub
    GroupOne.GetServerObject.ConnectServer(True)
    Set objOPCDA = Nothing
    Set objOPC = Nothing
    Set GroupOne = Nothing
End Sub
```

GetXMLSettings, OPCClientGroupObjCmdTarget Function

Syntax GetXMLSettings()

Description Questa funzione restituisce il testo XML di impostazione e configurazione del gruppo OPC referenziato ed inserito nel progetto. Di fatto è il contenuto del file di risorse "nomeprogetto.movopcclient" relativo al solo gruppo referenziato.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget
    Dim GroupOne As OPCClientGroupObjCmdTarget

    Set objOPC = GetOPCClient
    If objOPC Is Nothing Then Exit Sub
    Set objOPCDA = objOPC.GetOPCDAlientObject("Softing OPC Toolbox Demo OPC
    DA Server")
    If objOPCDA Is Nothing Then Exit Sub
    Set GroupOne = objOPCDA.GetGroupObject("Group one")
    If GroupOne Is Nothing Then Exit Sub
    MsgBox "XMLSettings are " &
    GroupOne.GetXMLSettings,vbInformation,GetProjectTitle

    Set objOPCDA = Nothing
    Set objOPC = Nothing
    Set GroupOne = Nothing
```

End Sub

IsGroupConnected, OPCClientGroupObjCmdTarget Function

Syntax IsGroupConnected()

Description Questa funzione restituisce il valore booleano True se il Gruppo di Item è connesso.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set GroupOne = objOPCDA.GetGroupObject("Group one")  
    If GroupOne Is Nothing Then Exit Sub  
    If GroupOne.IsGroupConnected Then  
        MsgBox "GroupOne is connected",vbInformation,GetProjectTitle  
    Else  
        MsgBox "GroupOne is NOT connected",vbInformation,GetProjectTitle  
    End If  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
    Set GroupOne = Nothing  
End Sub
```

UpdateGroupProperties, OPCClientGroupObjCmdTarget Function

Syntax UpdateGroupProperties()

Description Questa funzione forza l'aggiornamento delle proprietà di un gruppo di item OPC.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
```

```

Dim objOPC As OPCClientCmdTarget
Dim objOPCDA As OPCClientObjCmdTarget
Dim GroupOne As OPCClientGroupObjCmdTarget

Set objOPC = GetOPCClient
If objOPC Is Nothing Then Exit Sub
Set objOPCDA = objOPC.GetOPCDAlientObject("Softing OPC Toolbox Demo OPC DA Server")
If objOPCDA Is Nothing Then Exit Sub
Set GroupOne = objOPCDA.GetGroupObject("Group one")
If GroupOne Is Nothing Then Exit Sub
GroupOne.DeadBand = 0.50
GroupOne.UpdateGroupProperties

Set objOPCDA = Nothing
Set objOPC = Nothing
Set GroupOne = Nothing
End Sub

```

Properties

Active, OPCClientGroupObjCmdTarget Property

Syntax Active = _Boolean

Description Questa proprietà è di sola lettura e restituisce lo stato dell'opzione "Active" impostata in sviluppo. Tramite la proprietà "EnableGroup" è poi possibile abilitare/disabilitare un gruppo in runtime. Attenzione però che se la gestione delle variabili in uso è abilitata, ogni volta che una variabile del gruppo torna in uso il gruppo viene automaticamente riattivato.

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget
    Dim GroupOne As OPCClientGroupObjCmdTarget

    Set objOPC = GetOPCClient
    If objOPC Is Nothing Then Exit Sub
    Set objOPCDA = objOPC.GetOPCDAlientObject("Softing OPC Toolbox Demo OPC DA Server")
    If objOPCDA Is Nothing Then Exit Sub
    Set GroupOne = objOPCDA.GetGroupObject("Group one")
    If GroupOne.Active then
        MsgBox "GroupOne is active",vbInformation,GetProjectTitle
    Else
        MsgBox "GroupOne is NOT active",vbInformation,GetProjectTitle
    End If
    Set objOPCDA = Nothing
    Set objOPC = Nothing
    Set GroupOne = Nothing
End Sub

```

DeadBand, OPCClientGroupObjCmdTarget Property

Syntax DeadBand = _Single

Description Questa proprietà imposta o restituisce un valore percentuale da 0 a 100 di banda morta, per gli Item contenuti all'interno del gruppo referenziato. La banda morta è applicata agli Item del Gruppo che hanno il parametro EU Type impostato su Analog, in questo caso i parametri EU Low e EU High vengono usati per calcolare l'intervallo dell'Item. L'intervallo viene moltiplicato per la banda morta per generare un'eccezione di limite nel caso in cui la differenza fra il valore precedentemente letto e il nuovo superi tale calcolo.
La banda morta è usata per eliminare problemi dovuti a disturbi nella lettura di valori analogici: infatti in caso di eccezione, l'Item rimane al valore precedentemente letto.

Parameter	Description
None	None

Result Single

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set GroupOne = objOPCDA.GetGroupObject("Group one")  
    MsgBox "DeadBand is " & GroupOne.DeadBand ,vbInformation,GetProjectTitle  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
    Set GroupOne = Nothing  
End Sub
```

EnableGroup, OPCClientGroupObjCmdTarget Property

Syntax EnableGroup = _Boolean

Description Questa proprietà abilita o disabilita il Gruppo referenziato e di conseguenza gli Item contenuti al suo interno indipendentemente dal fatto che la Gestione delle Variabili In Uso sia abilitata o meno.
Pertanto indipendentemente dalla gestione delle variabili in uso se la proprietà EnableGroup è impostata a false, ogni volta che una variabile del gruppo torna 'In Uso' il Gruppo OPC non viene automaticamente riattivato.
Di conseguenza quando un gruppo viene disabilitato, non sarà mai gestito fino a quando non viene riabilitato, a meno che in quella condizione non venga disattivata la Gestione delle variabili 'In Uso' tramite la proprietà 'EnableInUseVarMng' dell'interfaccia 'DBVariableCmdTarget'.

In lettura 'EnableGroup' potrebbe ritornare il valore 'true' anche se in realtà il gruppo OPC è stato disabilitato dalla gestione delle variabili in uso; mentre se la gestione delle variabili in uso è disabilitata la proprietà 'EnableGroup' rispecchia sempre lo stato attuale del gruppo.

Inoltre se 'EnableGroup' è impostata su 'false' potrebbe ritornare sempre il valore 'false' nel momento in cui gestione delle variabili in uso da abilitata viene impostata disabilitata.

La funzione "EnableGroup" interroga il server opc e gli chiede se il gruppo OPC è attivo nel suo workspace. Quando non è marcata l'opzione "Disattiva se non in uso" nel gruppo in Movicon, la funzione 'EnableGroup' può ritornare sempre 'true' anche dopo avere eseguito "EnableGroup=false", perchè nel server opc il gruppo è ancora attivo (semplicemente è cambiato il refresh time).



la proprietà 'EnableGroup' non ha effetto per le variabili collegate ad un OPC Item tramite la loro proprietà Indirizzo Fisico I/O se il Gruppo OPC è presente nella risorsa 'OPC Client DA' ed impostato sulla proprietà 'Nome Gruppo OPC' della variabile. E' necessario utilizzare il metodo 'EnableDynOPCGroup' dell'interfaccia 'OPCClientCmdTarget'.



Attenzione! La sintassi "EnableGroup = Not EnableGroup" (ev. inserita in un pulsante) restituisce sempre un 'EnableGroup=False'; per ottenere la negazione della proprietà sarà quindi necessario inserire ad esempio 2 pulsanti nei quali inserire: nel primo 'EnableGroup=true' e nel secondo 'EnableGroup=false'.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAlientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set GroupOne = objOPCDA.GetGroupObject("Group one")  
    If GroupOne.EnableGroupthen  
        MsgBox "GroupOne is enabled",vbInformation,GetProjectTitle  
    Else  
        MsgBox "GroupOne is NOT enabled",vbInformation,GetProjectTitle  
    End If  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
    Set GroupOne = Nothing  
End Sub
```

LocalID, OPCClientGroupObjCmdTarget Property

Syntax LocalID = _Long

Description Questa proprietà restituisce il LocalID relativo al gruppo referenziato.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget
    Dim GroupOne As OPCClientGroupObjCmdTarget

    Set objOPC = GetOPCClient
    If objOPC Is Nothing Then Exit Sub
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
    DA Server")
    If objOPCDA Is Nothing Then Exit Sub
    Set GroupOne = objOPCDA.GetGroupObject("Group one")
    MsgBox "LocalID is " & GroupOne.LocalID,vbInformation,GetProjectTitle
    Set objOPCDA = Nothing
    Set objOPC = Nothing
    Set GroupOne = Nothing
End Sub
```

NotInUseDisableGroup, OPCClientGroupObjCmdTarget Property

Syntax NotInUseDisableGroup = _Boolean

Description Questa proprietà permette di disattivare il Gruppo quando le variabili associate agli Item non sono in uso.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget
    Dim GroupOne As OPCClientGroupObjCmdTarget

    Set objOPC = GetOPCClient
    If objOPC Is Nothing Then Exit Sub
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
    DA Server")
    If objOPCDA Is Nothing Then Exit Sub
    Set GroupOne = objOPCDA.GetGroupObject("Group one")
    MsgBox "NotInUseDisableGroup is " & GroupOne.NotInUseDisableGroup,
    vbInformation,GetProjectTitle
    Set objOPCDA = Nothing
    Set objOPC = Nothing
    Set GroupOne = Nothing
End Sub
```


NotInUseRefreshRate, OPCClientGroupObjCmdTarget Property

Syntax NotInUseRefreshRate = _Long

Description Questa proprietà permette di impostare il tempo in millisecondi di aggiornamento degli Item all'interno del Gruppo quando le variabili associate agli Item non sono in uso nel progetto.



Questo tempo di rinfresco dovrà ovviamente essere impostato superiore al tempo "Rinfresca Ogni".

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAlientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set GroupOne = objOPCDA.GetGroupObject("Group one")  
    MsgBox "NotInUseRefreshRate is " & GroupOne.NotInUseRefreshRate  
    ,vbInformation,GetProjectTitle  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
    Set GroupOne = Nothing  
End Sub
```

TimeBias, OPCClientGroupObjCmdTarget Property

Syntax TimeBias = _Long

Description Questa proprietà permette di impostare un valore in minuti che servirà per convertire la proprietà Time Stamp degli Item contenuti all'interno del Gruppo al tempo locale del dispositivo. Normalmente non è necessario modificare tale impostazione dal valore zero di default.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget
```

```

Dim objOPCDA As OPCClientObjCmdTarget
Dim GroupOne As OPCClientGroupObjCmdTarget

Set objOPC = GetOPCClient
If objOPC Is Nothing Then Exit Sub
Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
DA Server")
If objOPCDA Is Nothing Then Exit Sub
Set GroupOne = objOPCDA.GetGroupObject("Group one")
MsgBox "TimeBias is " & GroupOne.TimeBias,vbInformation,GetProjectTitle
Set objOPCDA = Nothing
Set objOPC = Nothing
Set GroupOne = Nothing
End Sub

```

UpdateRate, OPCClientGroupObjCmdTarget Property

Syntax UpdateRate = _Long

Description Questa proprietà permette di impostare il tempo in millisecondi di aggiornamento degli Item all'interno del Gruppo.



Questo tempo di rinfresco è un parametro che viene passato al Server. Nel caso in cui il tempo di refresh risulti essere troppo basso il Server utilizza un tempo di refresh consono alle proprie possibilità.

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
Dim objOPC As OPCClientCmdTarget
Dim objOPCDA As OPCClientObjCmdTarget
Dim GroupOne As OPCClientGroupObjCmdTarget

Set objOPC = GetOPCClient
If objOPC Is Nothing Then Exit Sub
Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
DA Server")
If objOPCDA Is Nothing Then Exit Sub
Set GroupOne = objOPCDA.GetGroupObject("Group one")
MsgBox "UpdateRate is " &
GroupOne.UpdateRate,vbInformation,GetProjectTitle
Set objOPCDA = Nothing
Set objOPC = Nothing
Set GroupOne = Nothing
End Sub

```

1.16.7. OPCClientItemObjCmdTarget

Functions

ForceReadData, OPCClientItemObjCmdTarget Function

Syntax ForceReadData(_bFromDevice)

Description Questa funzione forza una rilettura dell'item referenziato.

Parameter	Description
bFromDevice	Se impostato al valore booleano true specifica a Movicon una richiesta di rilettura dei dati dal campo.

Result Boolean

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget  
    Dim ItemOne As OPCClientItemObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set GroupOne = objOPCDA.GetGroupObject("Group one")  
    If GroupOne Is Nothing Then Exit Sub  
    Set ItemOne = GroupOne.GetItemObject("Var00001")  
    If ItemOne Is Nothing Then Exit Sub  
    ItemOne.ForceReadData(True)  
    Set ItemOne = Nothing  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
    Set GroupOne = Nothing  
End Sub
```

ForceWriteData, OPCClientItemObjCmdTarget Function

Syntax ForceWriteData = Boolean

Description Questo metodo consente di forzare la scrittura dell'item, anche se il valore della variabile collegata non è cambiato. Questo metodo può risultare utile quando si utilizzato OPC Server che si interfacciano a reti LON o EIB, in cui a volte è necessario rispedire lo stesso valore anche se questo non è cambiato.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget  
    Dim ItemOne As OPCClientItemObjCmdTarget  
    Dim i As Integer  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set GroupOne = objOPCDA.GetGroupObject("Group one")  
    If GroupOne Is Nothing Then Exit Sub  
    Set ItemOne = GroupOne.GetItemObject("Var00001")  
    If ItemOne Is Nothing Then Exit Sub  
    Dim i As Integer  
        MsgBox "SyncDataAtStartup is " &  
        CBool(ItemOne.ForceWriteData), vbInformation, GetProjectTitle  
    Next i  
    Set ItemOne = Nothing  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
    Set GroupOne = Nothing  
End Sub
```

GetGroupObject, OPCClientItemObjCmdTarget Function

Syntax GetGroupObject()

Description Questa funzione restituisce un oggetto di tipo OPCClientGroupObjCmdTarget per la gestione dei metodi e delle proprietà del gruppo di appartenenza dell'item referenziato.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo OPCClientGroupObjCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget  
    Dim ItemOne As OPCClientItemObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set GroupOne = objOPCDA.GetGroupObject("Group one")  
    If GroupOne Is Nothing Then Exit Sub  
    Set ItemOne = GroupOne.GetItemObject("Var00001")  
    If ItemOne Is Nothing Then Exit Sub  
    ItemOne.GetGroupObject.DeadBand = 0.25
```

```

ItemOne.GetGroupObject.UpdateGroupProperties
Set ItemOne = Nothing
Set objOPCDA = Nothing
Set objOPC = Nothing
Set GroupOne = Nothing
End Sub

```

GetItemQuality, OPCClientItemObjCmdTarget Function

Syntax GetItemQuality()

Description Questa funzione restituisce la qualità dell'item referenziato.

I valori restituiti sono i valori di 'qualità' come da specifiche OPC:

```

252 = OPC_STATUS_MASK
3 = OPC_LIMIT_MASK
0 = OPC_QUALITY_BAD
64 = OPC_QUALITY_UNCERTAIN
192 = OPC_QUALITY_GOOD
4 = OPC_QUALITY_CONFIG_ERROR
8 = OPC_QUALITY_NOT_CONNECTED
12 = OPC_QUALITY_DEVICE_FAILURE
16 = OPC_QUALITY_SENSOR_FAILURE
20 = OPC_QUALITY_LAST_KNOWN
24 = OPC_QUALITY_COMM_FAILURE
28 = OPC_QUALITY_OUT_OF_SERVICE
68 = OPC_QUALITY_LAST_USABLE
80 = OPC_QUALITY_SENSOR_CAL
84 = OPC_QUALITY_EGU_EXCEEDED
88 = OPC_QUALITY_SUB_NORMAL
216 = OPC_QUALITY_LOCAL_OVERRIDE

```

Parameter	Description
None	None

Result Integer

Example:

```

Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget
    Dim GroupOne As OPCClientGroupObjCmdTarget
    Dim ItemOne As OPCClientItemObjCmdTarget

    Set objOPC = GetOPCClient
    If objOPC Is Nothing Then Exit Sub
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
    DA Server")
    If objOPCDA Is Nothing Then Exit Sub
    Set GroupOne = objOPCDA.GetGroupObject("Group one")
    If GroupOne Is Nothing Then Exit Sub
    Set ItemOne = GroupOne.GetItemObject("Var00001")
    If ItemOne Is Nothing Then Exit Sub
    MsgBox "ItemOne Quality is " &
    ItemOne.GetItemQuality, vbInformation, GetProjectTitle
    Set ItemOne = Nothing
    Set objOPCDA = Nothing
    Set objOPC = Nothing
    Set GroupOne = Nothing
End Sub

```

GetItemTimeStamp, OPCClientItemObjCmdTarget Function

Syntax GetItemTimeStamp()

Description Questa funzione restituisce l'ultimo valore di Time Stamp relativo all'item referenziato.

Parameter	Description
None	None

Result Date

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget  
    Dim ItemOne As OPCClientItemObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set GroupOne = objOPCDA.GetGroupObject("Group one")  
    If GroupOne Is Nothing Then Exit Sub  
    Set ItemOne = GroupOne.GetItemObject("Var00001")  
    If ItemOne Is Nothing Then Exit Sub  
    MsgBox "Item TimeStamp is " &  
    ItemOne.GetItemTimeStamp, vbInformation, GetProjectTitle  
    Set ItemOne = Nothing  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
    Set GroupOne = Nothing  
End Sub
```

GetLinkedVariableObject, OPCClientItemObjCmdTarget Function

Syntax GetLinkedVariableObject()

Description Questa funzione restituisce l'oggetto di tipo DBVarObjCmdTarget relativo alla variabile del Real Time DB del progetto associata all'Item OPC referenziato.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo DBVarObjCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget
```

```

Dim GroupOne As OPCClientGroupObjCmdTarget
Dim ItemOne As OPCClientItemObjCmdTarget

Set objOPC = GetOPCClient
If objOPC Is Nothing Then Exit Sub
Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
DA Server")
If objOPCDA Is Nothing Then Exit Sub
Set GroupOne = objOPCDA.GetGroupObject("Group one")
If GroupOne Is Nothing Then Exit Sub
Set ItemOne = GroupOne.GetItemObject("Var00001")
If ItemOne Is Nothing Then Exit Sub
MsgBox "Linked variable BGColorProp is " &
ItemOne.GetLinkedVariableObject.BGColorProp ,vbInformation,GetProjectTitle
Set ItemOne = Nothing
Set objOPCDA = Nothing
Set objOPC = Nothing
Set GroupOne = Nothing
End Sub

```

GetNumObjectsInHeap, OPCClientItemObjCmdTarget Function

Syntax GetNumObjectsInHeap()

Description Questa funzione restituisce il numero di oggetti caricati in memoria.

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
Dim objOPC As OPCClientCmdTarget
Dim objOPCDA As OPCClientObjCmdTarget
Dim GroupOne As OPCClientGroupObjCmdTarget
Dim ItemOne As OPCClientItemObjCmdTarget

Set objOPC = GetOPCClient
If objOPC Is Nothing Then Exit Sub
Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
DA Server")
If objOPCDA Is Nothing Then Exit Sub
Set GroupOne = objOPCDA.GetGroupObject("Group one")
If GroupOne Is Nothing Then Exit Sub
Set ItemOne = GroupOne.GetItemObject("Var00001")
If ItemOne Is Nothing Then Exit Sub
MsgBox "NumObjectsInHeap are " &
ItemOne.GetNumObjectsInHeap,vbInformation,GetProjectTitle
Set ItemOne = Nothing
Set objOPCDA = Nothing
Set objOPC = Nothing
Set GroupOne = Nothing
End Sub

```

GetXMLSettings, OPCClientItemObjCmdTarget Function

Syntax GetXMLSettings()

Description Questa funzione restituisce il testo XML di impostazione e configurazione dell'item OPC referenziato ed inserito nel progetto. Di fatto è il contenuto del file di risorse "nomeprogetto.movopcclient" relativo al solo item referenziato.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget  
    Dim ItemOne As OPCClientItemObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set GroupOne = objOPCDA.GetGroupObject("Group one")  
    If GroupOne Is Nothing Then Exit Sub  
    Set ItemOne = GroupOne.GetItemObject("Var00001")  
    If ItemOne Is Nothing Then Exit Sub  
    MsgBox "XML Settings are " &  
    ItemOne.GetXMLSettings,vbInformation,GetProjectTitle  
    Set ItemOne = Nothing  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
    Set GroupOne = Nothing  
End Sub
```

IsItemConnected, OPCClientItemObjCmdTarget Function

Syntax IsItemConnected()

Description Questa funzione restituisce il valore booleano true se l'item referenziato risulta connesso.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget
```



```

Dim GroupOne As OPCClientGroupObjCmdTarget
Dim ItemOne As OPCClientItemObjCmdTarget

Set objOPC = GetOPCClient
If objOPC Is Nothing Then Exit Sub
Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
DA Server")
If objOPCDA Is Nothing Then Exit Sub
Set GroupOne = objOPCDA.GetGroupObject("Group one")
If GroupOne Is Nothing Then Exit Sub
Set ItemOne = GroupOne.GetItemObject("Var00001")
If ItemOne Is Nothing Then Exit Sub
    MsgBox "Is " & CBool(ItemOne.IsItemConnected) & " that the item is
connected",vbInformation,GetProjectTitle
Set ItemOne = Nothing
Set objOPCDA = Nothing
Set objOPC = Nothing
Set GroupOne = Nothing
End Sub

```

ReconnectItem, OPCClientItemObjCmdTarget Function

Syntax ReconnectItem(_bWait)

Description Questa funzione consente di forzare la riconnessione dell'item specificando se è necessario o meno attendere il tempo definito nella proprietà 'Tempo di Riconnessione'.

Parameter	Description
bWait As Boolean	abilitazione all'attesa del tempo di riconnessione

Result Boolean

Example:

```

Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget
    Dim GroupOne As OPCClientGroupObjCmdTarget
    Dim ItemOne As OPCClientItemObjCmdTarget

    Set objOPC = GetOPCClient
    If objOPC Is Nothing Then Exit Sub
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
DA Server")
    If objOPCDA Is Nothing Then Exit Sub
    Set GroupOne = objOPCDA.GetGroupObject("Group one")
    If GroupOne Is Nothing Then Exit Sub
    Set ItemOne = GroupOne.GetItemObject("Var00001")
    If ItemOne Is Nothing Then Exit Sub
    ItemOne.ReconnectItem(True)
    Set ItemOne = Nothing
    Set objOPCDA = Nothing
    Set objOPC = Nothing
    Set GroupOne = Nothing
End Sub

```

Properties

EnableRead, OPCClientItemObjCmdTarget Property

Syntax EnableRead = _Boolean

Description Questa proprietà, se impostata al valore booleano true, abilita in lettura l'item referenziato. In questo caso una modifica dell'item all'interno del server comporta un aggiornamento della variabile del progetto di Movicon.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
Dim objOPC As OPCClientCmdTarget  
Dim objOPCDA As OPCClientObjCmdTarget  
Dim GroupOne As OPCClientGroupObjCmdTarget  
Dim ItemOne As OPCClientItemObjCmdTarget  
  
Set objOPC = GetOPCClient  
If objOPC Is Nothing Then Exit Sub  
Set objOPCDA = objOPC.GetOPCDAlientObject("Softing OPC Toolbox Demo OPC  
DA Server")  
If objOPCDA Is Nothing Then Exit Sub  
Set GroupOne = objOPCDA.GetGroupObject("Group one")  
If GroupOne Is Nothing Then Exit Sub  
Set ItemOne = GroupOne.GetItemObject("Var00001")  
If ItemOne Is Nothing Then Exit Sub  
MsgBox "ItemOne.EnableRead is " & CBool(ItemOne.EnableRead)  
,vbInformation,GetProjectTitle  
Set ItemOne = Nothing  
Set objOPCDA = Nothing  
Set objOPC = Nothing  
Set GroupOne = Nothing  
End Sub
```

EnableWrite, OPCClientItemObjCmdTarget Property

Syntax EnableWrite = _Boolean

Description Questa proprietà, se impostata al valore booleano true, abilita in scrittura l'item referenziato. In questo caso la modifica della variabile all'interno del progetto di Movicon comporta la scrittura dell'item sul server.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget  
    Dim ItemOne As OPCClientItemObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set GroupOne = objOPCDA.GetGroupObject("Group one")  
    If GroupOne Is Nothing Then Exit Sub  
    Set ItemOne = GroupOne.GetItemObject("Var00001")  
    If ItemOne Is Nothing Then Exit Sub  
    MsgBox "ItemOne.EnableWrite is " & CBool(ItemOne.EnableWrite)  
    ,vbInformation,GetProjectTitle  
    Set ItemOne = Nothing  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
    Set GroupOne = Nothing  
End Sub
```

ItemID, OPCClientItemObjCmdTarget Property

Syntax ItemID = _String

Description Questa proprietà restituisce l'ID relativo all'item referenziato.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget  
    Dim ItemOne As OPCClientItemObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set GroupOne = objOPCDA.GetGroupObject("Group one")  
    If GroupOne Is Nothing Then Exit Sub  
    Set ItemOne = GroupOne.GetItemObject("Var00001")  
    If ItemOne Is Nothing Then Exit Sub  
    MsgBox "ItemID is " & ItemOne.ItemID ,vbInformation,GetProjectTitle  
    Set ItemOne = Nothing  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
    Set GroupOne = Nothing  
End Sub
```

ItemPath, OPCClientItemObjCmdTarget Property

Syntax ItemPath = _String

Description Questa proprietà restituisce il percorso OPC relativo all'item referenziato.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget  
    Dim ItemOne As OPCClientItemObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set GroupOne = objOPCDA.GetGroupObject("Group one")  
    If GroupOne Is Nothing Then Exit Sub  
    Set ItemOne = GroupOne.GetItemObject("Var00001")  
    If ItemOne Is Nothing Then Exit Sub  
    MsgBox "ItemPath is " & ItemOne.ItemPath  
    ,vbInformation,GetProjectTitle  
    Set ItemOne = Nothing  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
    Set GroupOne = Nothing  
End Sub
```

LinkedVariable, OPCClientItemObjCmdTarget Property

Syntax LinkedVariable = _String

Description Questa proprietà restituisce il nome della variabile del Real Time DB del progetto da associare all'Item OPC del Server.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
```

```

Dim objOPC As OPCClientCmdTarget
Dim objOPCDA As OPCClientObjCmdTarget
Dim GroupOne As OPCClientGroupObjCmdTarget
Dim ItemOne As OPCClientItemObjCmdTarget

Set objOPC = GetOPCClient
If objOPC Is Nothing Then Exit Sub
Set objOPCDA = objOPC.GetOPCDAlientObject("Softing OPC Toolbox Demo OPC
DA Server")
If objOPCDA Is Nothing Then Exit Sub
Set GroupOne = objOPCDA.GetGroupObject("Group one")
If GroupOne Is Nothing Then Exit Sub
Set ItemOne = GroupOne.GetItemObject("Var00001")
If ItemOne Is Nothing Then Exit Sub
    MsgBox "LinkedVariable is " & ItemOne.LinkedVariable
    ,vbInformation,GetProjectTitle
Set ItemOne = Nothing
Set objOPCDA = Nothing
Set objOPC = Nothing
Set GroupOne = Nothing
End Sub

```

ReRead, OPCClientItemObjCmdTarget Property

Syntax ReRead = _Boolean

Description Questa proprietà consente di decidere se eseguire una lettura sincrona del valore ogni qualvolta viene eseguita una scrittura (sincrona o asincrona). Potrebbe essere necessario utilizzando alcuni OPC Server (vedi RsLinx della Rockwell) che non gestiscono in modo corretto le notifiche asincrone.

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget
    Dim GroupOne As OPCClientGroupObjCmdTarget
    Dim ItemOne As OPCClientItemObjCmdTarget

    Set objOPC = GetOPCClient
    If objOPC Is Nothing Then Exit Sub
    Set objOPCDA = objOPC.GetOPCDAlientObject("Softing OPC Toolbox Demo OPC
    DA Server")
    If objOPCDA Is Nothing Then Exit Sub
    Set GroupOne = objOPCDA.GetGroupObject("Group one")
    If GroupOne Is Nothing Then Exit Sub
    Set ItemOne = GroupOne.GetItemObject("Var00001")
    If ItemOne Is Nothing Then Exit Sub
        MsgBox "SyncDataAtStartup is " & CBoole(ItemOne.ReRead)
        ,vbInformation,GetProjectTitle
    Set ItemOne = Nothing
    Set objOPCDA = Nothing
    Set objOPC = Nothing
    Set GroupOne = Nothing
End Sub

```

SyncDataAtStartup, OPCClientItemObjCmdTarget Property

Syntax SyncDataAtStartup = _Boolean

Description Questa proprietà consente di decidere se eseguire una lettura dell'item allo startup del progetto, dopo che è stato creato, per sincronizzare il valore della variabile con quello del campo. Di default questa funzionalità è disabilitata per evitare che l'inizializzazione della comunicazione OPC richieda troppo tempo a causa della sincronizzazione impostata su tutti gli item. Si può abilitare la proprietà negli item su cui è veramente necessario sincronizzare i valori.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget  
    Dim ItemOne As OPCClientItemObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set GroupOne = objOPCDA.GetGroupObject("Group one")  
    If GroupOne Is Nothing Then Exit Sub  
    Set ItemOne = GroupOne.GetItemObject("Var00001")  
    If ItemOne Is Nothing Then Exit Sub  
    MsgBox "SyncDataAtStartup is " &  
        CBoole(ItemOne.SyncDataAtStartup), vbInformation, GetProjectTitle  
    Set ItemOne = Nothing  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
    Set GroupOne = Nothing  
End Sub
```

vtType, OPCClientItemObjCmdTarget Property

Syntax vtType = _Integer

Description Questa proprietà permette di impostare il tipo di variabile in lettura dal Server anche se è comunque consigliato lasciare il formato nativo.

Le selezioni possibili sono:

- Server di Default: Abilitando questa casella di opzione, in seguito alla lettura o scrittura di un item, non viene fatta alcun tipo di conversione. In questo caso si consiglia di utilizzare una variabile Movicon dello stesso tipo di quella definita nelle proprietà dell'item.
 - il valore associato a questa opzione è 0
- Short, Long, Float, Double, ecc.: A seconda del formato impostato, viene eseguita la corrispondente conversione sull'item, quindi la variabile di progetto, assegnata, conterrà il dato nella forma indicata qui.

- short
- long
- float
- double
- string
- boolean
- char
- byte
- word
- dword

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget
    Dim GroupOne As OPCClientGroupObjCmdTarget
    Dim ItemOne As OPCClientItemObjCmdTarget

    Set objOPC = GetOPCClient
    If objOPC Is Nothing Then Exit Sub
    Set objOPCDA = objOPC.GetOPCDAlientObject("Softing OPC Toolbox Demo OPC
    DA Server")
    If objOPCDA Is Nothing Then Exit Sub
    Set GroupOne = objOPCDA.GetGroupObject("Group one")
    If GroupOne Is Nothing Then Exit Sub
    Set ItemOne = GroupOne.GetItemObject("Var00001")
    If ItemOne Is Nothing Then Exit Sub
    MsgBox "vtType is " & ItemOne.vtType ,vbInformation,GetProjectTitle
    Set ItemOne = Nothing
    Set objOPCDA = Nothing
    Set objOPC = Nothing
    Set GroupOne = Nothing
End Sub
```

WriteSync, OPCClientItemObjCmdTarget Property

Syntax WriteSync = _Boolean

Description Questa proprietà consente di decidere se eseguire le scrittura sincrone (default) o asincrone. Le scritture asincrone possono essere eseguite molto velocemente, rispetto a quelle sincrone, ma il Server OPC deve essere sviluppato in modo da considerare l'eventualità di ricevere un numero molto elevato di richieste, ad esempio se le variabili nel progetto Movicon cambiano continuamente di valore. Per questo motivo il valore di default è impostato su Sincrona che è una gestione più lenta ma più sicura (va bene con tutti gli OPC Server).

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget
    Dim GroupOne As OPCClientGroupObjCmdTarget
    Dim ItemOne As OPCClientItemObjCmdTarget

    Set objOPC = GetOPCClient
    If objOPC Is Nothing Then Exit Sub
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
    DA Server")
    If objOPCDA Is Nothing Then Exit Sub
    Set GroupOne = objOPCDA.GetGroupObject("Group one")
    If GroupOne Is Nothing Then Exit Sub
    Set ItemOne = GroupOne.GetItemObject("Var00001")
    If ItemOne Is Nothing Then Exit Sub
        MsgBox "SyncDataAtStartup is " & CBoole(ItemOne.WriteSync)
        ,vbInformation,GetProjectTitle
    Set ItemOne = Nothing
    Set objOPCDA = Nothing
    Set objOPC = Nothing
    Set GroupOne = Nothing
End Sub

```

1.16.8. OPCClientObjCmdTarget

Functions

ConnectServer, OPCClientObjCmdTarget Function

Syntax ConnectServer(_bWait)

Description Questa funzione consente di forzare la connessione al server specificando se è necessario o meno attendere il tempo definito nella proprietà 'Tempo di Riconnessione'.

Parameter	Description
bWait As Boolean	abilitazione per l'attesa prima della connessione al server

Result Boolean

Example:

```

Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget
    Set objOPC = GetOPCClient

    If objOPC Is Nothing Then Exit Sub
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
    DA Server")
    If objOPCDA Is Nothing Then Exit Sub
    objOPCDA.ConnectServer(True)
    Set objOPCDA = Nothing
    Set objOPC = Nothing
End Sub

```


GetGroupObject, OPCClientObjCmdTarget Function

Syntax GetGroupObject(_IpszGroupName)

Description Questa funzione restituisce un oggetto di tipo OPCClientGroupObjCmdTarget per la gestione delle proprietà e dei metodi relativi al gruppo di Item referenziato all'interno dell'OPC client.

Parameter	Description
IpszGroupName As String	Nome del gruppo da reperire

Result Object
Viene restituito un oggetto di tipo OPCClientGroupObjCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim GroupOne As OPCClientGroupObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set GroupOne = objOPCDA.GetGroupObject("Group one")  
    MsgBox "GroupOne.Active is " & GroupOne.Active,vbInformation,GetProjectTitle  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
    Set GroupOne = Nothing  
End Sub
```

GetNodeName, OPCClientObjCmdTarget Function

Syntax GetNodeName()

Description Questa funzione restituisce il nome del nodo OPC relativo al server OPC DA referenziato.

Parameter	Description
None	None

Result String

GetNumObjectsInHeap, OPCClientObjCmdTarget Function

Syntax GetNumObjectsInHeap()

Description Questa funzione restituisce il numero di oggetti caricati in memoria.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    MsgBox "NumObjectsInHeap are " &  
    objOPCDA.GetNumObjectsInHeap,vbInformation,GetProjectTitle  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
End Sub
```

GetOPCClientDocObj, OPCClientObjCmdTarget Function

Syntax GetOPCClientDocObj()

Description Questa funzione restituisce il Doc Object relativo al server OPC referenziato.

Parameter	Description
None	None

Result Object

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
    Dim vResult As Long  
    Dim DOCobj As Object  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA =  
    objOPC.GetOPCDAClientObject("Softing.OPCToolboxDemo_ServerDA.1")  
    If objOPCDA Is Nothing Then Exit Sub  
    Set DOCobj = objOPCDA.GetOPCClientDocObj
```

```

        If Not DOCobj Is Nothing Then
            ...
        End If
        Set DOCobj = Nothing
        Set objOPCDA = Nothing
        Set objOPC = Nothing
    End Sub

```

GetServerCLSID, OPCClientObjCmdTarget Function

Syntax GetServerCLSID()

Description Questa funzione restituisce il CLS ID relativo al server OPC DA referenziato.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget

    Set objOPC = GetOPCClient
    If objOPC Is Nothing Then Exit Sub
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
    DA Server")
    If objOPCDA Is Nothing Then Exit Sub
    MsgBox "Server CLS ID is " &
    objOPCDA.GetServerCLSID,vbInformation,GetProjectTitle
    Set objOPCDA = Nothing
    Set objOPC = Nothing
End Sub

```

GetServerName, OPCClientObjCmdTarget Function

Syntax GetServerName()

Description Questa funzione restituisce il nome del server OPC DA referenziato.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim objOPC As OPCClientCmdTarget

```

```

Dim objOPCDA As OPCClientObjCmdTarget

Set objOPC = GetOPCClient
If objOPC Is Nothing Then Exit Sub
Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
DA Server")
If objOPCDA Is Nothing Then Exit Sub
MsgBox "Server Name is " &
objOPCDA.GetServerName,vbInformation,GetProjectTitle
Set objOPCDA = Nothing
Set objOPC = Nothing
End Sub

```

GetServerStatus, OPCClientObjCmdTarget Function

Syntax GetServerStatus()

Description Questa funzione restituisce lo stato del server OPC DA referenziato.

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget

    Set objOPC = GetOPCClient
    If objOPC Is Nothing Then Exit Sub
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
    DA Server")
    If objOPCDA Is Nothing Then Exit Sub
    MsgBox "Server Status is " &
    objOPCDA.GetServerStatus,vbInformation,GetProjectTitle
    Set objOPCDA = Nothing
    Set objOPC = Nothing
End Sub

```

GetServerVendorInfo, OPCClientObjCmdTarget Function

Syntax GetServerVendorInfo()

Description Questa funzione restituisce informazioni relative al server OPC DA referenziato.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAlientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    MsgBox "Server VendorInfo are " &  
    objOPCDA.GetServerVendorInfo,vbInformation,GetProjectTitle  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
End Sub
```

GetXMLSettings, OPCClientObjCmdTarget Function

Syntax GetXMLSettings()

Description Questa funzione restituisce il testo XML di impostazione e configurazione dell'oggetto OPC referenziato ed inserito nel progetto. Di fatto è il contenuto del file di risorse "nomeprogetto.movopcclient" relativo al solo server referenziato.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim objOPC As OPCClientCmdTarget  
    Dim objOPCDA As OPCClientObjCmdTarget  
  
    Set objOPC = GetOPCClient  
    If objOPC Is Nothing Then Exit Sub  
    Set objOPCDA = objOPC.GetOPCDAlientObject("Softing OPC Toolbox Demo OPC  
    DA Server")  
    If objOPCDA Is Nothing Then Exit Sub  
    MsgBox "XMLSettings are " &  
    objOPCDA.GetXMLSettings,vbInformation,GetProjectTitle  
    Set objOPCDA = Nothing  
    Set objOPC = Nothing  
End Sub
```

IsConnected, OPCClientObjCmdTarget Function

Syntax IsConnected()

Description Questa funzione restituisce il valore booleano true se il server è connesso.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget

    Set objOPC = GetOPCClient
    If objOPC Is Nothing Then Exit Sub
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
    DA Server")
    If objOPCDA Is Nothing Then Exit Sub
    MsgBox "Server 'IsConnected' is " &
    objOPCDA.IsConnected,vbInformation,GetProjectTitle
    Set objOPCDA = Nothing
    Set objOPC = Nothing
End Sub
```

Properties

ReconnectTime, OPCClientObjCmdTarget Property

Syntax ReconnectTime = _Long

Description Questaproprietà imposta o restituisce il tempo, espresso in millisecondi, dopo il quale il Server verrà riconnesso.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
    Dim objOPC As OPCClientCmdTarget
    Dim objOPCDA As OPCClientObjCmdTarget

    Set objOPC = GetOPCClient
    If objOPC Is Nothing Then Exit Sub
    Set objOPCDA = objOPC.GetOPCDAClientObject("Softing OPC Toolbox Demo OPC
    DA Server")
    If objOPCDA Is Nothing Then Exit Sub
    MsgBox "Server ReconnectTime is " &
    objOPCDA.ReconnectTime,vbInformation,GetProjectTitle
    Set objOPCDA = Nothing
    Set objOPC = Nothing
End Sub
```

ReReadDynamicItems, OPCClientObjCmdTarget Property

Syntax ReReadDynamicItems = _Bool

Description Questa proprietà, che viene applicata soltanto agli item creati in modo dinamico, ovvero inserendo direttamente il link OPC nella proprietà "Indirizzo Fisico I/O" di una variabile, consente di decidere se eseguire una lettura sincrona del valore ogni qualvolta viene eseguita una scrittura (sincrona o asincrona). Potrebbe essere necessario utilizzando alcuni OPC Server (vedi RsLinx della Rockwell) che non gestiscono in modo corretto le notifiche asincrone. La proprietà può essere sia letta che scritta, ma la sua modifica a runtime non verrà applicata agli item dinamici già creati.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim objOPCClient As OPCClientObjCmdTarget  
  
    Set objOPCClient = GetOPCClient().GetOPCDAClientObject("Softing OPC Toolbox  
    Demo OPC DA Server")  
    If objOPCClient Is Nothing Then Exit Sub  
    MsgBox "ReReadDynamicItems is " &  
    objOPCClient.ReReadDynamicItems, vbInformation, GetProjectTitle  
    Set objOPCClient = Nothing  
End Sub
```

SyncDynamicItemsAtStartup, OPCClientObjCmdTarget Property

Syntax SyncDynamicItemsAtStartup = _Bool

Description Questa proprietà, che viene applicata soltanto agli item creati in modo dinamico, ovvero inserendo direttamente il link OPC nella proprietà "Indirizzo Fisico I/O" di una variabile, consente di decidere se eseguire una lettura iniziale all'avvio del progetto del valore. La proprietà può essere sia letta che scritta, ma la sua modifica a runtime non ha alcun effetto: è possibile eseguire uno script in sviluppo e rendere persistente tale proprietà.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit  
Dim objOPCClient As OPCClientObjCmdTarget  
Public Sub Click()  
    Set objOPCClient =  
    GetOPCClient().GetOPCDAClientObject("Softing.OPCToolboxDemo_ServerDA.1")  
    'Softing OPC Toolbox Demo OPC DA Server
```

```

    If objOPCClient Is Nothing Then Exit Sub
    MsgBox "SyncDynamicItemsAtStartup is " &
objOPCClient.SyncDynamicItemsAtStartup,vbInformation,GetProjectTitle
    Set objOPCClient = Nothing
End Sub

```

WatchdogTime, OPCClientObjCmdTarget Property

Syntax WatchdogTime = _Long

Description Consente di impostare un tempo di attesa (ms) prima di controllare lo stato di connessione del Server OPC DA.
Il Watchdog viene monitorato solo per gli OPC Server che presentano dei gruppi attivi.

Parameter	Description
None	None

Result Long

Example:

```

Option Explicit
Public Sub Click()
Dim objOPC As OPCClientCmdTarget
Dim objOPCDA As OPCClientObjCmdTarget
    Set objOPC = GetOPCClient
    If Not objOPC Is Nothing Then
        Set objOPCDA =
objOPC.GetOPCDAClientObject("Softing.OPCToolboxDemo_ServerDA.1")
        If Not objOPCDA Is Nothing Then
            objOPCDA.WatchdogTime = 5000
            Debug.Print "WatchdogTime: " & CStr(objOPCDA.WatchdogTime)
        End If
        Set objOPCDA = Nothing
        Set objOPC = Nothing
    End If
End Sub

```

1.16.9. OPCServerCmdTarget

Properties

ServerStatus, OPCServerCmdTarget Property

Syntax ServerStatus()

Description Questa proprietà non è gestita. Il valore restituito è sempre "0".

Parameter	Description
None	None

Result Integer

Example:

Functions

FireAEEvent, OPCServerCmdTarget Function

Syntax FireAEEvent(_lpszSource, _lpszMessage, _dwPriority, _dwSeverity)

Description Questa funzione non è gestita. Il valore restituito è sempre "True".

Parameter	Description
None	None

Result Boolean

Example:

GetNumServingTags, OPCServerCmdTarget Function

Syntax GetNumServingTags()

Description Questa funzione restituisce il numero di tag che il Server OPC Movicon del progetto pubblica verso i Client OPC.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim objOPCServe As OPCServerCmdTarget  
    Dim sOPCNumServingTags As Long  
  
    Set objOPCServe = GetOPCServer()  
    If Not objOPCServe Is Nothing Then  
        sOPCNumServingTags = objOPCServe.GetNumServingTags  
        MsgBox "OPC Num Serving Tags = " & sOPCNumServingTags,  
            vbInformation,  
            GetProjectTitle  
        Set objOPCServe = Nothing  
    End If
```

End Sub

GetServerName,OPCServerCmdTarget Function

Syntax GetServerName()

Description Questa funzione restituisce il nome del Server OPC Movicon del progetto.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim objOPCServe As OPCServerCmdTarget  
    Dim sOPCServerName As String  
  
    Set objOPCServe = GetOPCServer()  
    If Not objOPCServe Is Nothing Then  
        sOPCServerName = objOPCServe.GetServerName  
        MsgBox "OPC Server Name = " & sOPCServerName, vbInformation,  
        GetProjectTitle  
        Set objOPCServe = Nothing  
    End If  
End Sub
```

GetNumConnectedClients, OPCServerCmdTarget Function

Syntax GetNumConnectedClients()

Description Questa funzione restituisce il numero di OPC Client connessi al Server OPC Movicon del progetto.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim objOPCServe As OPCServerCmdTarget  
    Dim sOPCNumConnectedClients As Long  
  
    Set objOPCServe = GetOPCServer()  
    If Not objOPCServe Is Nothing Then  
        sOPCNumConnectedClients = objOPCServe.GetNumConnectedClients  
        MsgBox "OPC Num Connected Clients = " & sOPCNumConnectedClients,  
        vbInformation,  
        GetProjectTitle  
    End If  
End Sub
```

```

        Set objOPCServe = Nothing
    End If
End Sub

```

1.16.10. OPCUAClientCmdTarget

Functions

GetOPCUAClientObject, OPCUAClientCmdTarget Function

Syntax GetOPCUAClientObject()

Description Ritorna un oggetto OPCUAClientObjCmdTarget attraverso il quale è possibile gestire le proprietà e i metodi relativi alla comunicazione dello standard OPC UA Data Access.

Parameter	Description
IpszServerName As String	Nome del Server OPC UA

Result Object
Viene restituito un oggetto di tipo OPCUAClientObjCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto è Nothing.

Example:

```

Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
    Set objOPCUAClientCmdTarget = GetOPCUAClient()
    Set objOPCUAClient
=objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")

    Debug.Print CStr(objOPCUAClient.GetEndpoint)

    Set objOPCUAClient = Nothing
End Sub

```

GetXMLSettings, OPCUAClientCmdTarget Function

Syntax GetXMLSettings

Description Ritorna il contenuto XML della risorsa OPC Client UA

Parameter	Description
None	None

Result String

Example:
Option Explicit
Sub Click()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient

MsgBox objOPCUAClientCmdTarget.GetXMLSettings

Set objOPCUAClientCmdTarget= Nothing
End Sub

Property

StartupTimeout, OPCUAClientCmdTarget Property

Syntax StartupTimeout = _Long

Description Questa proprietà imposta o restituisce il tempo di timeout, espresso in msec, per il processo di inizializzazione dei Tag OPC UA al momento della connessione al Server OPC UA.
Nota: la scrittura del valore non è permanente ma vale solo per la sessione in Runtime

Parameter	Description
None	None

Result Long

Example:
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient()

objOPCUAClientCmdTarget.StartupTimeout = 3000

Debug.Print CStr(objOPCUAClientCmdTarget.StartupTimeout)

Set objOPCUAClientCmdTarget = Nothing
End Sub

1.16.11. OPCUAClientObjCmdTarget

Functions

GetBackupEndpoint, OPCUAClientObjCmdTarget Function

Syntax GetBackupEndpoint()

Description La funzione restituisce l'indirizzo Endpoint del Server OPC UA di backup a cui è collegata la risorsa OPC UA Client

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient()
Set objOPCUAClient
=objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
If Not objOPCUAClient Is Nothing Then
Debug.Print CStr(objOPCUAClient.GetBackupEndpoint)
Set objOPCUAClient = Nothing
End If

End Sub
```

GetEndpoint, OPCUAClientObjCmdTarget Function

Syntax GetEndpoint()

Description La funzione restituisce l'indirizzo Endpoint del Server OPC UA a cui è collegata la risorsa OPC UA Client

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient()
Set objOPCUAClient
=objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
If Not objOPCUAClient Is Nothing Then
Debug.Print CStr(objOPCUAClient.GetBackupEndpoint)
Set objOPCUAClient = Nothing
End If

End Sub
```

GetNodeName, OPCUAClientObjCmdTarget Function

Syntax GetNodeName()

Description La funzione è obsoleta e restituisce una stringa vuota

Parameter	Description
None	None

Result String

GetOPCUAClientDocObj, OPCUAClientObjCmdTarget Function

Syntax GetOPCUAClientDocObj()

Description La funzione restituisce il Doc Object OPCUAClientCmdTarget relativo al Server OPC UA referenziato

Parameter	Description
None	None

Result Object

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objDoc As OPCUAClientCmdTarget

Set objOPCUAClientCmdTarget = GetOPCUAClient()
Set objOPCUAClient
=objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")

Set objDoc = objOPCUAClient.GetOPCUAClientDocObj()

If Not objDoc Is Nothing Then
    Debug.Print CStr(objDoc.StartupTimeout)
    Set objDoc = Nothing
End If

End Sub
```

GetSecurityMode, OPCUAClientObjCmdTarget Function

Syntax GetSecurityMode

Description La funzione restituisce l'indice di selezione della Modalità di Sicurezza selezionata per l'accesso al Server OPC UA

Parameter	Description
None	None

Result 0 - Nessuno
1 - Sign
2 - SignAndEncrypt

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient = objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
If Not objOPCUAClient Is Nothing Then
Debug.Print CStr(objOPCUAClient .GetSecurityMode)
Set objOPCUAClient = Nothing
End If
Set objOPCUAClientCmdTarget = Nothing
End Sub
```

GetServerName, OPCUAClientObjCmdTarget Function

Syntax GetOPCUAClientDocObj()

Description La funzione restituisce il Nome della Risorsa OPC UA Client per il Server OPC UA referenziato

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient = objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
If Not objOPCUAClient Is Nothing Then
Debug.Print CStr(objOPCUAClient .GetServerName)
Set objOPCUAClient = Nothing
End If
```

```
Set objOPCUAClientCmdTarget = Nothing
End Sub
```

GetSecurityPolicy, OPCUAClientObjCmdTarget Function

Syntax GetSecurityPolicy()

Description	La funzione è obsoleta e restituisce una stringa vuota
--------------------	--

Parameter	Description
None	None

Result	String
--------	--------

GetSessionObject, OPCUAClientObjCmdTarget Function

Syntax `GetSessionObject()`

Description	La funzione restituisce un oggetto di tipo OPCUAClientSessionObjCmdTarget per la gestione delle proprietà e dei metodi relativi alla Sessione dei Tag OPC UA referenziata all'interno dell'OPC UA Client.
--------------------	---

Parameter	Description
IpszSessionName As String	Nome della Sessione OPC UA Client da referenziare

Result	Object Viene restituito un oggetto di tipo OPCUAClientSessionObjCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto risulta a valore Nothing.
---------------	---

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAclientCmdTarget As OPCUAclientCmdTarget
Dim objOPCUAclient As OPCUAclientObjCmdTarget
Dim objSessionObj As OPCUAclientSessionObjCmdTarget
    Set objOPCUAclientCmdTarget = GetOPCUAclient
    Set                                objOPCUAclient                                =
objOPCUAclientCmdTarget.GetOPCUAclientObject("opc.tcp://localhost:62841-None")
If Not objOPCUAclient Is Nothing Then
    Set objSessionObj = objOPCUAclient.GetSessionObject("Session one")
    If not objSessionObj Is Nothing Then
        Debug.Print CStr(objSessionObj.GetName)
    End If
    Set objSessionObj = Nothing
    Set objOPCUAclient = Nothing
End If
Set objOPCUAclientCmdTarget = Nothing
```


GetXMLSettings, OPCUAClientObjCmdTarget Function

Syntax GetXMLSettings()

Description La funzione restituisce il contenuto XML che descrive la configurazione della risorsa OPC UA Client del Server OPC UA referenziato. Di fatto è il contenuto del file di risorse "NomeProgetto.MovOpcUaClient" relativo allo specifico Server OPC UA referenziato.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient = objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None") =
If Not objOPCUAClient Is Nothing Then
Debug.Print objOPCUAClient.GetXMLSettings
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
End Sub
```

IsConnected, OPCUAClientObjCmdTarget Function

Syntax IsConnected()

Description La funzione restituisce il valore booleano True se il Server OPC UA referenziato è connesso alla risorsa Client OPC UA.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
```

```

Set objOPCUAClient = objOPCUAClient
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
If Not objOPCUAClient Is Nothing Then
    Debug.Print CStr(objOPCUAClient.IsConnected)
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
End Sub

```

Property

Endpoint, OPCUAClientObjCmdTarget Property

Syntax Endpoint = _String

Description Proprietà che legge l'Endpoint configurato o lo imposta per modificare l'indirizzo del Server OPC UA per la sessione di Runtime della risorsa Client OPC UA.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient = objOPCUAClient
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
If Not objOPCUAClient Is Nothing Then
    Debug.Print CStr("Old Endpoint: " & objOPCUAClient.Endpoint)
    'Set new Endpoint
    objOPCUAClient.Endpoint = "opc.tcp://mylocalhost:62841"
    Debug.Print CStr("New Endpoint: " & objOPCUAClient.Endpoint)
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
End Sub

```

Retries, OPCUAClientObjCmdTarget Property

Syntax Endpoint = _Long

Description Proprietà che legge o imposta il numero di tentativi di connessione al Server OPC UA principale prima di effettuare lo switching al Server OPC UA di Backup.

Parameter	Description
-----------	-------------

None	None
------	------

Result Long

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient = objOPCUAClient
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None") =
If Not objOPCUAClient Is Nothing Then
Debug.Print CStr("Previous Retries: " & objOPCUAClient.Retries)
'Set new Retries
objOPCUAClient.Retries= 10

Debug.Print CStr("New Rreties: " & objOPCUAClient.Retries)
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
End Sub
```

StatusVariable, OPCUAClientObjCmdTarget Property

Syntax Endpoint = _String

Description Proprietà che legge o imposta la proprietà Variabile di Stato Comando impostata nelle proprietà del Server OPC UA referenziato dalla risorsa Client OPC UA.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient = objOPCUAClient
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None") =
If Not objOPCUAClient Is Nothing Then
Debug.Print objOPCUAClient.StatusVariable
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
End Sub
```

BackupEndpoint, OPCUAClientObjCmdTarget Property

Syntax BackupEndpoint = _String

Description Proprietà che legge il BackupEndpoint configurato o lo imposta per modificare l'indirizzo del Server OPC UA di Backup per la sessione di Runtime della risorsa Client OPC UA.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient = objOPCUAClient =
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
If Not objOPCUAClient Is Nothing Then
Debug.Print CStr("Old Endpoint: " & objOPCUAClient.BackupEndpoint)
'Set new Endpoint
objOPCUAClient.BackupEndpoint= "opc.tcp://mylocalhost:62841"
Debug.Print CStr("New Endpoint: " & objOPCUAClient.BackupEndpoint)
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
End Sub
```

ReconnectTime, OPCUAClientObjCmdTarget Property

Syntax Endpoint = _Long

Description Proprietà che legge o imposta il tempo di attesa prima di effettuare lo switching al Server OPC UA di Backup se il Server OPC UA Principale non è più disponibile, e viceversa una volta che il Server OPC UA Principale ritorna attivo.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
```

```

Set objOPCUAClient =
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
If Not objOPCUAClient Is Nothing Then
    Debug.Print CStr("Previous Retries: " & objOPCUAClient.ReconnectTime)
    'Set new Retries
    objOPCUAClient.ReconnectTime= 5000

    Debug.Print CStr("New Reconnect Time: " & objOPCUAClient.ReconnectTime)
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
End Sub

```

1.16.12. OPCUAClientSessionObjCmdTarget

Property

ConnectTimeout, OPCUAClientSessionObjCmdTarget Property

Syntax ConnectTimeout = _Long

Description Proprietà che permette di leggere o impostare il tempo, espresso in millisecondi, della durata massima della fase di connessione dopo il quale indicare lo stato di errore di Timeout di Connessione.

Parameter	Description
None	None

Result Long

Example:

```

Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objSessionObj As OPCUAClientSessionObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient =
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")
If Not objSessionObj Is Nothing Then
    Debug.Print CStr("Old Connection Timeout: " &
CStr(objSessionObj.ConnectTimeout))
    objSessionObj.ConnectTimeout = 2000
    Debug.Print CStr("New Connection Timeout: " &
CStr(objSessionObj.ConnectTimeout))
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
Set objSessionObj = Nothing
End Sub

```

PublishingInterval, OPCUAClientSessionObjCmdTarget Property

Syntax PublishingInterval = _Long

Description Proprietà che permette di leggere o impostare il tempo, in millisecondi, di notifica delle variazioni dei valori o stato delle Tag per la Sessione corrente dall'OPC Server UA verso il Client OPC UA di Movicon.

Questo tempo di notifica richiesto al Server OPC UA è in realtà a discrezione del Server stesso in base alle proprie possibilità

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objSessionObj As OPCUAClientSessionObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient =
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")
If Not objSessionObj Is Nothing Then
Debug.Print CStr("Publishing Interval: " & CStr(objSessionObj.PublishingInterval))
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
Set objSessionObj = Nothing
End Sub
```

SamplingInterval, OPCUAClientSessionObjCmdTarget Property

Syntax SamplingInterval = _Long

Description Proprietà che permette di leggere o impostare il tempo, in millisecondi, di aggiornamento delle Tag sul Server OPC UA e sottoscritte nella Sessione corrente dal Client OPC UA del progetto Movicon.

Questo tempo di campionamento richiesto al Server OPC UA è in realtà a discrezione del Server stesso in base alle proprie possibilità

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objSessionObj As OPCUAClientSessionObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient =
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")
If Not objSessionObj Is Nothing Then
Debug.Print CStr("Sampling Interval: " & CStr(objSessionObj.SamplingInterval))
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
Set objSessionObj = Nothing
End Sub
```

Functions

GetItemObject, OPCUAClientSessionObjCmdTarget Function

Syntax GetItemObject()

Description La funzione restituisce l'oggetto di tipo OPCUAClientItemObjCmdTarget relativo alla Tag OPC UA Client referenziata.

Parameter	Description
lpszItemName As String	Nome della Tag OPC UA Client di cui reperire l'oggetto.

Result Object
Viene restituito un oggetto di tipo OPCUAClientItemObjCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objSessionObj As OPCUAClientSessionObjCmdTarget
Dim objItem As OPCUAClientItemObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient =
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")

Set objItem = objSessionObj.GetItemObject("Tags_Variable2")
If Not objItem Is Nothing Then
Debug.Print objItem.NodeID
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
Set objSessionObj = Nothing
Set objItem = Nothing
End Sub
```

GetName, OPCUAClientSessionObjCmdTarget Function

Syntax GetName()

Description La funzione restituisce il nome della Sessione referenziato appartenente alla risorsa Client OPC UA del progetto.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objSessionObj As OPCUAClientSessionObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient =
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")

Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")
If Not objSessionObj Is Nothing Then
    Debug.Print objSessionObj.GetName
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
Set objSessionObj = Nothing
End Sub
```

GetServerObject, OPCUAClientSessionObjCmdTarget Function

Syntax GetServerObject()

Description La funzione restituisce l'oggetto di tipo OPCUAClientObjCmdTarget a cui appartiene la risorsa Sessione referenziata e relativa al Server OPC UA a cui è connessa la risorsa OPC UA Client stessa.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo OPCClientObjCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
```



```

Dim objSessionObj As OPCUAClientSessionObjCmdTarget
Dim objOPCUACliObj As OPCUAClientObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient =
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")

Set objOPCUACliObj = objSessionObj.GetServerObject
If Not objOPCUACliObj Is Nothing Then
    Debug.Print objOPCUACliObj.EndPoint
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
Set objSessionObj = Nothing
Set objOPCUACliObj = Nothing
End Sub

```

GetXMLSettings, OPCUAClientSessionObjCmdTarget Function

Syntax GetXMLSettings()

Description La funzione restituisce il testo XML di definizione della risorsa Sessione appartenente all'oggetto di tipo OPCUAClientObjCmdTarget referenziata. Di fatto è il contenuto del file di risorse "NomepProgetto.movOPCUAClient" relativo alla sola Sessione referenziata.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objSession01Obj As OPCUAClientSessionObjCmdTarget
Dim objSession02Obj As OPCUAClientSessionObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient =
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")

Set objSession01Obj = objOPCUAClient.GetSessionObject("Session one")
Set objSession02Obj = objOPCUAClient.GetSessionObject("Session two")

If (Not objSession01Obj Is Nothing) And (Not objSession02Obj Is Nothing) Then
    Debug.Print "Session one: " & objSession01Obj.GetXMLSettings() & " - " & _
        "Session two: " & objSession02Obj.GetXMLSettings()
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
Set objSession01Obj = Nothing
Set objSession02Obj = Nothing
End Sub

```

IsSessionConnected, OPCUAClientSessionObjCmdTarget Function

Syntax IsSessionConnected()

Description La funzione restituisce un valore Booleano True o False dello stato di connessione al Server OPC UA della risorsa Sessione OPC UA configurata nella risorsa OPC UA Client stessa.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objSessionObj As OPCUAClientSessionObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient =
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")

If Not objSessionObj Is Nothing Then
    Debug.Print "Sessione " & CStr(objSessionObj.GetName) & " Connected: " &
CStr(objSessionObj.IsSessionConnected())
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
Set objSessionObj = Nothing
End Sub
```

1.16.13. OPCUAClientItemObjCmdTarget

Functions

GetItemQuality, OPCUAClientItemObjCmdTarget Function

Syntax GetItemQuality()

Description La funzione restituisce l'indice di qualità della connessione della Tag OPC UA referenziata esposta dal Server OPC UA a cui è connessa la risorsa Client OPC UA.

I valori restituiti sono i valori di 'qualità' per le Tag OPC UA come da specifiche:

252 = OPC_STATUS_MASK
0 = OPC_QUALITY_BAD
64 = OPC_QUALITY_UNCERTAIN
192 = OPC_QUALITY_GOOD
4 = OPC_QUALITY_CONFIG_ERROR
8 = OPC_QUALITY_NOT_CONNECTED
12 = OPC_QUALITY_DEVICE_FAILURE

16 = OPC_QUALITY_SENSOR_FAILURE
 20 = OPC_QUALITY_LAST_KNOWN
 24 = OPC_QUALITY_COMM_FAILURE
 28 = OPC_QUALITY_OUT_OF_SERVICE
 68 = OPC_QUALITY_LAST_USABLE
 80 = OPC_QUALITY_SENSOR_CAL
 84 = OPC_QUALITY_EGU_EXCEEDED
 88 = OPC_QUALITY_SUB_NORMAL
 216 = OPC_QUALITY_LOCAL_OVERRIDE

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objSessionObj As OPCUAClientSessionObjCmdTarget
Dim objItem As OPCUAClientItemObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient =
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")
Set objItem = objSessionObj.GetItemObject("Tags_Variable1")
If Not objItem Is Nothing Then
Debug.Print CStr(objItem.GetItemQuality)
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
Set objSessionObj = Nothing
Set objItem = Nothing
End Sub
  
```

GetItemTimeStamp, OPCUAClientItemObjCmdTarget Function

Syntax GetItemTimeStamp()

Description La funzione restituisce l'ultimo valore di Time Stamp relativo alla Tag OPC UA referenziata e connessa la risorsa Client OPC UA.

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
  
```

```

Dim objSessionObj As OPCUAClientSessionObjCmdTarget
Dim objItem As OPCUAClientItemObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClientCmdTarget = objOPCUAClient
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")
Set objItem = objSessionObj.GetItemObject("Tags_Variable1")
If Not objItem Is Nothing Then
    Debug.Print CStr(objItem.GetItemTimeStamp)
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
Set objSessionObj = Nothing
Set objItem = Nothing
End Sub

```

GetLinkedVariableObject, OPCUAClientItemObjCmdTarget Function

Syntax GetItemQuality()

Description La funzione restituisce l'interfaccia DBVarObjCmdTarget della Variabile del RealtimeDB collegata alla Tag OPC UA della risorsa Client OPC UA.

Parameter	Description
None	None

Result Object

Example:

```

Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objSessionObj As OPCUAClientSessionObjCmdTarget
Dim objItem As OPCUAClientItemObjCmdTarget
Dim objVar As DBVarObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClientCmdTarget = objOPCUAClient
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")
Set objItem = objSessionObj.GetItemObject("Tags_Variable1")
Set objVar = objItem.GetLinkedVariableObject
If Not objVar Is Nothing Then
    Debug.Print objVar.GetName
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
Set objSessionObj = Nothing
Set objItem = Nothing
Set objVar = Nothing
End Sub

```

GetName, OPCUAClientItemObjCmdTarget Function

Syntax GetName()

Description La funzione restituisce il nome del Tag OPC UA Client dell'oggetto OPC UA Item stesso presente nella risorsa Client OPC UA.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objSessionObj As OPCUAClientSessionObjCmdTarget
Dim objItem As OPCUAClientItemObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient = objOPCUAClient
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")
Set objItem = objSessionObj.GetItemObject("Tags_Variable1")
If Not objItem Is Nothing Then
    Debug.Print objItem.GetName
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
Set objSessionObj = Nothing
Set objItem = Nothing
End Sub
```

GetSessionObject, OPCUAClientItemObjCmdTarget Function

Syntax GetSessionObject()

Description La funzione restituisce un oggetto del tipo OPCUAClientSessionObjCmdTarget della risorsa Sessione OPC UA referenziata all'interno della risorsa Client OPC UA del progetto, permettendo di reperire metodi e proprietà relativi all'interfaccia dell'oggetto Sessione OPC UA stesso.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
```

```

Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objSessionObj As OPCUAClientSessionObjCmdTarget
    Set objOPCUAClientCmdTarget = GetOPCUAClient
    Set                                     objOPCUAClient                                     =
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")

    Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")
    If Not objSessionObj Is Nothing Then
        Debug.Print objSessionObj.GetXMLSettings
    End If
    Set objOPCUAClient = Nothing
    Set objOPCUAClientCmdTarget = Nothing
    Set objSessionObj = Nothing
End Sub

```

GetXMLSettings, OPCUAClientItemObjCmdTarget Function

Syntax GetXMLSettings()

Description La funzione restituisce il contenuto testuale XML che descrive il Tag OPC UA Client referenziato presente nella risorsa Client OPC UA. Di fatto è il contenuto del file di risorse "NomeProgetto.movOPCUAClient" relativo al solo Tag OCP UA referenziato.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objSessionObj As OPCUAClientSessionObjCmdTarget
Dim objItem As OPCUAClientItemObjCmdTarget
    Set objOPCUAClientCmdTarget = GetOPCUAClient
    Set                                     objOPCUAClient                                     =
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None")
    Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")
    Set objItem = objSessionObj.GetItemObject("Tags_Variable1")
    If Not objItem Is Nothing Then
        Debug.Print objItem.GetXMLSettings
    End If
    Set objOPCUAClient = Nothing
    Set objOPCUAClientCmdTarget = Nothing
    Set objSessionObj = Nothing
    Set objItem = Nothing
End Sub

```

Property

LinkedVariable, OPCUAClientItemObjCmdTarget Property

Syntax LinkedVariable = _String

Description Proprietà che restituisce o imposta il nome della variabile del Real Time DB del progetto associata o da associare al Tag OPC UA della risorsa OPC UA Client connessa al Server OPC UA.

Le modifiche alla proprietà sono volatili e valgono solo per la sessione runtime in corso

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objSessionObj As OPCUAClientSessionObjCmdTarget
Dim objItem As OPCUAClientItemObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient = objOPCUAClient
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None") =
Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")
Set objItem = objSessionObj.GetItemObject("Tags_Variable1")
If Not objItem Is Nothing Then
Debug.Print "Linked Variable to the OPC UA Tag: " & objItem.LinkedVariable)
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
Set objSessionObj = Nothing
Set objItem = Nothing
End Sub
```

NodeID, OPCUAClientItemObjCmdTarget Property

Syntax NodeID = _String

Description Proprietà che restituisce l'ID relativo al Tag OPC UA della risorsa OPC UA Client connessa al Server OPC UA.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objSessionObj As OPCUAClientSessionObjCmdTarget
Dim objItem As OPCUAClientItemObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient = objOPCUAClient
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None") =
Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")
Set objItem = objSessionObj.GetItemObject("Tags_Variable1")
If Not objItem Is Nothing Then
Debug.Print "Tag NodeID: " & objItem.NodeID)
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
Set objSessionObj = Nothing
Set objItem = Nothing
End Sub
```

EnableRead, OPCUAClientItemObjCmdTarget Property

Syntax EnableRead = _String

Description Proprietà Booleana che restituisce o imposta l'abilitazione alla lettura del Tag OPC UA referenziato della risorsa OPC UA Client connessa ad un Server OPC UA. Se abilitata a 'True' consente la lettura delle eventuali modifiche notificate dal Server OPC UA ed il conseguente aggiornamento della Variabile del progetto Movicon.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objSessionObj As OPCUAClientSessionObjCmdTarget
Dim objItem As OPCUAClientItemObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient = objOPCUAClient
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None") =
Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")
Set objItem = objSessionObj.GetItemObject("Tags_Variable1")
If Not objItem Is Nothing Then
Debug.Print "EnableRead : " & objItem.EnableRead)
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
Set objSessionObj = Nothing
Set objItem = Nothing
```


End Sub

EnableWrite, OPCUAClientItemObjCmdTarget Property

Syntax EnableWrite = _String

Description Proprietà Booleana che restituisce o imposta l'abilitazione alla Scrittura del Tag OPC UA referenziato della risorsa OPC UA Client connessa ad un Server OPC UA. Se abilitata a 'True' consente le notifiche al Server OPC UA delle eventuali modifiche della Variabile effettuate nel progetto Movicon.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Sub Main()
Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget
Dim objOPCUAClient As OPCUAClientObjCmdTarget
Dim objSessionObj As OPCUAClientSessionObjCmdTarget
Dim objItem As OPCUAClientItemObjCmdTarget
Set objOPCUAClientCmdTarget = GetOPCUAClient
Set objOPCUAClient = objOPCUAClient
objOPCUAClientCmdTarget.GetOPCUAClientObject("opc.tcp://localhost:62841-None") =
Set objSessionObj = objOPCUAClient.GetSessionObject("Session one")
Set objItem = objSessionObj.GetItemObject("Tags_Variable1")
If Not objItem Is Nothing Then
Debug.Print "EnableWrite : " & objItem.EnableWrite
End If
Set objOPCUAClient = Nothing
Set objOPCUAClientCmdTarget = Nothing
Set objSessionObj = Nothing
Set objItem = Nothing
End Sub
```

1.16.14. PmeDocCmdTarget

1.17. Utilizzo della PmeDocCmdTarget

L'utilizzo dell'interfaccia di programmazione "PmeDocCmdTarget" è diretto, cioè non è necessario istanziare un oggetto di tipo "PmeDocCmdTarget" ma è possibile utilizzare le sue proprietà e metodi direttamente nel codice VB Script. In pratica tutte le proprietà e funzioni della "PmeDocCmdTarget" sono disponibili direttamente dall'Intellisense indipendentemente dal contesto VB Script in cui ci si trova.

Functions

AckAllAlarms, PmeDocCmdTarget Function

Syntax AckAllAlarms

Description Riconoscimento (Ack) di tutti gli allarmi attualmente presenti nel progetto di Movicon.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    AckAllAlarms  
End Sub
```

AddSysLogMessage, PmeDocCmdTarget Function

Syntax AddSysLogMessage(_IpszMessage)

Description Con questo metodo è possibile inviare dei messaggi di sistema al log di Movicon. I messaggi verranno quindi tracciati nella finestra di output "Sistema" e nel file del Log Storico.

Parameter	Description
IpszMessage As String	Testo del messaggio da stampare sul log storico.

Result Boolean

Example:

```
Option Explicit  
Sub Main  
    AddSysLogMessage("Test Message")  
End Sub
```

CreateObjectLic, PmeDocCmdTarget Function

Syntax CreateObjectLic(_IpszServerName, _IpszLicense)

Description Permette di creare oggetti da codice basic (ad esempio ActiveX) che necessitano di un numero di licenza. Se non si conoscono già i parametri da passare alla funzione vedere la sezione "Come utilizzare ActiveX con Licenza".

Parameter	Description
ByVal IpszServerName as String	Nome del server che crea il riferimento all'ActiveX.
ByVal IpszLicense as String	Numero di Licenza di Runtime o Sviluppo dell'ActiveX.

Result Object

Example:

```
Sub Main
    Dim Socket As MSWinsockLib.Winsock
    Set Socket = CreateObjectLic ("MSWinsock.Winsock.1", "2c49f800-xxx-xxx-xxx-0080c7e7b78d")
    If Socket Is Nothing Then Exit Sub
    Socket.Protocol = sckTCPProtocol
    Socket.LocalPort = 1000
    Socket.Listen
End Sub
```

CreateRemoteObject, PmeDocCmdTarget Function

Syntax CreateRemoteObject(_IpszServerName, _IpszLocation)

Description Definisce un oggetto collegato ad un componente server remoto installato su un altro computer. Un componente server remoto è progettato in modo compatibile allo standard DCOM ed è quindi visibile attraverso la rete.

Parameter	Description
IpszServerName As String	Nome del componente server DCOM.
IpszLocation As String	Nome del computer remoto o indirizzo IP.

Result Object

Example:

```
Sub Main
    Dim objRemote As Object

    Set objRemote =
CreateRemoteObject("RemoteServerName","RemotePCName")
    ...
    Set objRemote = Nothing
End Sub
```

GetAlarm, PmeDocCmdTarget Function

Syntax GetAlarm(_IpszAlarmName)

Description La funzione restituisce l'oggetto allarme identificato dal nome in formato stringa (parametro IpszAlarmName). Nel caso in cui l'allarme sia stato associato alla variabile (allarme come

template) non è più sufficiente specificare solo il nome dell'allarme, ma si dovrà specificare anche il nome della variabile:

GetAlarmObject(<Nome Allarme> <Nome Variabile>)

Parameter	Description
IpszAlarmName As String	Nome dell'allarme da reperire.

Result Object
Viene restituito un oggetto di tipo AlarmCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

Example1:

```
Public Sub Click()  
    Dim objAlarm As AlarmCmdTarget  
    Dim bResult As Boolean  
    Set objAlarm = GetAlarm("Alarm1")  
    bResult = objAlarm.Enabled  
    Debug.Print bResult  
    Set objAlarm = Nothing  
End Sub
```

Example2:

```
Public Sub Click()  
    'Alarm as template  
    Dim objAlarm As AlarmCmdTarget  
    Dim bResult As Boolean  
    Set objAlarm = GetAlarm("Alarm2 Var0001")  
    bResult = objAlarm.Enabled  
    Debug.Print bResult  
    Set objAlarm = Nothing  
End Sub
```

GetAlarmsPath, PmeDocCmdTarget Function

Syntax GetAlarmsPath()

Description Restituisce una stringa contenente la cartella di lavoro nella quale Movicon provvederà ad archiviare gli eventuali commenti associati dall'operatore a ciascun allarme.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sResult As String  
    sResult = GetAlarmsPath  
    Debug.Print sResult  
End Sub
```

GetAppTimeZone, PmeDocCmdTarget Function

Syntax GetAppTimeZone

Description Questa funzione consente di conoscere il timezone in minuti impostato nel computer che sta eseguendo il progetto.
Il timezone restituito tiene sempre conto dell'eventuale ora legale.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim sMsg As String  
    sMsg = CStr(GetAppTimeZone) & " min."  
    MsgBox "Time Zone : " & sMsg, vbOkOnly + vbInformation, GetProjectTitle  
End Sub
```

GetChildProject, PmeDocCmdTarget Function

Syntax GetChildProject(_IpszName)

Description Restituisce l'oggetto progetto figlio relativo al nome specificato. In questo modo sarà possibile accedere ai metodi e alle proprietà specifiche per il progetto figlio. L'oggetto restituito sarà di tipo PmeDocCmdTarget.

Parameter	Description
IpszName As String	Nome del progetto da reperire.

Result Object
Viene restituito un oggetto di tipo PmeDocCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()  
    Dim myObject As PmeDocCmdTarget  
    Set myObject = GetChildProject("ChildProject1")  
    'If String Table has already been initialised (Se inizializzata la Tabella delle Stringhe)  
    Debug.Print "Child.ActiveLanguage: " & myObject.ActiveLanguage  
    Set mObject = Nothing  
End Sub
```

GetCurrentListAlarms, PmeDocCmdTarget Function

Syntax GetCurrentListAlarms()

Description Restituisce una stringa contenente gli allarmi attualmente attivi. La stringa sarà composta in successione per ogni allarme dalla descrizione, dal nome della variabile associata e dal nome della soglia. La funzione ritorna una stringa vuota solo quando tutti gli allarmi sono stati resettati.

Il carattere che nella stringa separa gli allarmi è il "vbLf". Inoltre per ogni allarme la stringa è composta nel seguente modo:

NomeAllarme|Nomesoglia

se l'allarme è stato gestito come template allora la stringa risulterà essere la sequenet:

NomeAllarme NomeVariabile|Nomesoglia

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim aListAlarms() As String  
    Dim i As Integer  
  
    aListAlarms() = Split(CStr(GetCurrentListAlarms), vbLf)  
    On Error Resume Next  
    For i = 0 To UBound(aListAlarms)  
        Debug.Print "Alarm" & i & " -> " & aListAlarms(i)  
    Next  
End Sub
```

GetDataLoggerRecipe, PmeDocCmdTarget Function

Syntax GetDataLoggerRecipe(_IpszName)

Description Restituisce l'oggetto Data Logger identificato dal parametro IpszName.

Parameter	Description
IpszName As String	Nome del Data Logger da reperire.

Result Object
Viene restituito un oggetto di tipo DLRCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()  
    Dim mObject As DLRCmdTarget  
    Dim bResult As Boolean  
    Set mObject = GetDataLoggerRecipe("Log 5 sec")  
    bResult = mObject.Enabled  
    Debug.Print bResult  
    Set mObject = Nothing  
End Sub
```

GetDataLoggerRecipePath, PmeDocCmdTarget Function

Syntax GetDataLoggerRecipePath()

Description Restituisce una stringa contenente la cartella di lavoro nella quale Movicon provvederà ad archiviare i dati registrati dai Data Logger del progetto. Ulteriori informazioni sono riportate nella sezione Data Logger.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sResult As String  
    sResult = GetDataLoggerRecipePath  
    Debug.Print sResult  
End Sub
```

GetDataPath, PmeDocCmdTarget Function

Syntax GetDataPath()

Description Restituisce una stringa contenente la cartella di lavoro nella quale Movicon provvederà ad archiviare i dati relativi alle variabili retentive definite nel Real Time DB.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sResult As String  
    sResult = GetDataPath  
    Debug.Print sResult  
End Sub
```

GetDrawingPath, PmeDocCmdTarget Function

Syntax GetDrawingPath()

Description Restituisce una stringa contenente la cartella di lavoro nella quale Movicon ricercherà le immagini di tipo bitmap o jpg utilizzate all'interno del progetto.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sResult As String
    sResult = GetDrawingPath
    Debug.Print sResult
End Sub
```

GetDriverInterface, PmeDocCmdTarget Function

Syntax GetDriverInterface()

Description Permette di accedere ai metodi e alle proprietà della DriverInterface.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo DriverInterface se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()
    Dim myObject As DriverInterface
    Set myObject = GetDriverInterface("PC Adapter")
    Debug.Print myObject.DelayEvents
    Set myObject = Nothing
End Sub
```

GetEvent, PmeDocCmdTarget Function

Syntax GetEvent(_IpszEventName)

Description Restituisce l'oggetto Evento identificato dal parametro IpszEventName.

Parameter	Description
IpszEventName As String	Nome dell'evento da reperire.

Result Object
Viene restituito un oggetto di tipo EventCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:


```

Public Sub Click()
    Dim obj As EventCmdTarget
    Dim bResult As Boolean
    Set obj = GetEvent("Event")
    bResult = obj.Enabled
    Debug.Print bResult
    Set obj = Nothing
End Sub

```

GetFatherProject, PmeDocCmdTarget Function

Syntax GetFatherProject(_IpszName)

Description Restituisce l'oggetto progetto padre relativo al nome specificato. In questo modo sarà possibile accedere ai metodi e alle proprietà specifiche per il progetto padre. L'oggetto restituito sarà di tipo PmeDocCmdTarget.

Parameter	Description
IpszName As String	Nome del progetto da reperire.

Result Object
Viene restituito un oggetto di tipo PmeDocCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```

Public Sub Click()
    Dim mObject As PmeDocCmdTarget
    Set mObject = GetFatherProject("ProgettoFiglio1")
    'If String Table has already been initialised (Se inizializzata la Tabella delle Stringhe)
    Debug.Print "Father.ActiveLanguage: " & mObject.ActiveLanguage
    Set mObject = Nothing
End Sub

```

GetHisLogADOConn, PmeDocCmdTarget Function

Syntax GetHisLogADOConn()

Description Questa funzione consente di recuperare la connessione ADODB relativa alle Impostazioni Log Storico del Progetto.



Questa funzione se utilizzata su Windows CE restituisce sempre un'oggetto di tipo ADOCE.connection.3.1. Evitare inoltre la chiusura della connessione ADO usando il metodo "close", altrimenti anche Movicon non sarà più in grado di accedere a quel database.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo ADODB.Connection se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Sub Main
    Dim Conn1 As New ADODB.Connection
    Dim Rs1 As New ADODB.Recordset
    Dim contFields As Integer
    Dim sQuery As String
    Set Conn1 = GetHisLogADOConn
    sQuery = "SELECT * FROM SysMsgs"
    Set Rs1 = CreateObject("ADODB.Recordset")
    Rs1.Open sQuery, Conn1, adOpenForwardOnly, adLockReadOnly,
    ADODB.adCmdText
    ' Loop per stampare tutti i campi del recorset
    While Not Rs1.EOF
        For contFields = 0 To (Rs1.Fields.Count-1)
            Debug.Print Rs1.Fields(contFields).Name & " = " &
            Rs1.Fields(contFields).Value
        Next
        Rs1.MoveNext
    Wend
    Rs1.Close
    Conn1.Close
End Sub
```

GetHisLogDNSConnectionString, PmeDocCmdTarget Function

Syntax GetHisLogDNSConnectionString()

Description Restituisce il nome del collegamento ODBC che Movicon utilizza per la registrazione di tutti gli eventi di progetto o di sistema avvenuti durante l'elaborazione runtime del progetto applicativo. La registrazione degli eventi di Log viene eseguita su due file database differenti, a seconda del tipo di evento. In particolare tutti gli eventi che riguardano il "Tracing delle Variabili" verranno salvati nel file "NomeProgetto_TraceDB.mdb" all'interno della cartella di progetto "DATA". Tutte le altre informazioni invece verranno salvate nel File Database di Log "NomeProgetto_HisLog.mdb" all'interno della cartella di progetto "LOGS".

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sResult As String
    sResult = GetHisLogDNSConnectionString
    Debug.Print sResult
End Sub
```

GetIOPortInterface, PmeDocCmdTarget Function

Syntax GetIOPortInterface

Description Permette di accedere ai metodi e alle proprietà della IOPortInterface per la gestione delle porte di comunicazione.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo IOPortInterface se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()  
    Dim objIOPort As IOPortInterface  
    Dim ID As Long  
    Set objIOPort = GetIOPortInterface  
    ID = objIOPort.IOPortOpen("COM1:9600,n,8,1")  
    Set objIOPort = Nothing  
    MsgBox "PortOpen = " & ID, vbInformation, GetProjectTitle  
End Sub
```

GetOPCUAClient, PmeDocCmdTarget Function

Syntax GetOPCUAClient()

Description Permette di accedere ai metodi e alle proprietà della OPCUAClientCmdTarget.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo OPCUAClientCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto è Nothing.

Example:

```
Option Explicit  
Sub Main()  
    Dim objOPCUAClientCmdTarget As OPCUAClientCmdTarget  
    Set objOPCUAClientCmdTarget = GetOPCUAClient()  
    Debug.Print CStr(objOPCUAClientCmdTarget.StartupTimeout)  
End Sub
```

GetLastAlarmOn, PmeDocCmdTarget Function

Syntax GetLastAlarmOn

Description Permette di accedere ai metodi e alle proprietà della AlarmThresholdCmdTarget relativi all'ultimo allarme intervenuto.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo AlarmThresholdCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Option Explicit
Public Sub Click()
    Dim objAlarmThreshold As AlarmThresholdCmdTarget
    Set objAlarmThreshold = GetLastAlarmOn
    If objAlarmThreshold Is Nothing Then Exit Sub
    MsgBox("LastAlarmThreshold Name is " & objAlarmThreshold.Name,vbOkOnly, "")
End Sub
```

GetLogPath, PmeDocCmdTarget Function

Syntax GetLogPath()

Description Restituisce una stringa contenente la cartella di lavoro nella quale Movicon provvederà ad archiviare i dati registrati dal Log Storico Eventi e dal Trace delle Variabili del progetto. Ulteriori informazioni sono riportate nella sezione Log Storico.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sResult As String
    sResult = GetLogPath
    Debug.Print sResult
End Sub
```

GetNetworkClient, PmeDocCmdTarget Function

Syntax GetNetworkClient()

Description Permette di accedere ai metodi e alle proprietà descritti nell'interfaccia NetworkClientCmd.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo NetworkClientCmd se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()  
    Dim myObject As NetworkClientCmd  
    Set myObject = GetNetworkClient()  
    myObject.CloseAllConnections  
    Set myObject = Nothing  
End Sub
```

GetNetworkRedundancy, PmeDocCmdTarget Function

Syntax GetNetworkRedundancy()

Description Permette di accedere ai metodi e alle proprietà descritti nell'interfaccia NetworkRedudancyCmd.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo NetworkRedudancyCmd se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()  
    Dim objNR As NetworkRedudancyCmd  
    Set objNR = GetNetworkRedudancy  
    If Not objNR Is Nothing Then  
        MsgBox "ActNumRetries = " &  
            objNR.ActNumRetries, vbInformation, GetProjectTitle  
    End If  
End Sub
```

GetNetworkServer, PmeDocCmdTarget Function

Syntax GetNetworkServer()

Description Permette di accedere ai metodi e alle proprietà descritti nell'interfaccia NetworkServerCmd.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo NetworkServerCmd se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()  
    Dim myObject As NetworkServerCmd  
    Set myObject = GetNetworkServer()  
    myObject.CloseAllConnections  
    Set myObject = Nothing  
End Sub
```

GetNetworkUserLogPath, PmeDocCmdTarget Function

Syntax GetNetworkUserLogPath()

Description Restituisce una stringa contenente la cartella di lavoro nella quale Movicon provvederà ad inserire gli utenti di rete del progetto.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sResult As String  
    sResult = GetNetworkUserLogPath  
    Debug.Print sResult  
End Sub
```

GetOPCClient, PmeDocCmdTarget Function

Syntax GetOPCClient()

Description Permette di accedere ai metodi e alle proprietà della OPCClientCmdTarget.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo OPCClientCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()  
    Dim myObject As OPCClientCmdTarget  
    Dim bResult As Long  
    Set myObject = GetOPCClient()  
    bResult = myObject.TimeoutDynamicOperation
```

```

        Debug.Print bResult
        Set myObject = Nothing
    End Sub

```

GetOPCServer, PmeDocCmdTarget Function

Syntax GetOPCServer()

Description Permette di accedere ai metodi e alle proprietà della OPCServerCmdTarget.



Questa funzione non è supportata in Windows CE. (se utilizzata restituisce sempre 'null')

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo OPCServerCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```

Public Sub Click()
    Dim myObject As OPCServerCmdTarget
    Dim bResult As String
    Set myObject = GetOPCServer()
    bResult = myObject.ServerStatus
    Debug.Print bResult
    Set myObject = Nothing
End Sub

```

GetProjectFileName, PmeDocCmdTarget Function

Syntax GetProjectFileName()

Description Restituisce una stringa con il percorso e il nome del progetto in esecuzione.

Parameter	Description
None	None

Result String

Example:

```

Sub Main
    Dim sResult As String
    sResult = GetProjectFileName
    MsgBox("GetProjectFileName=" & sResult)
End Sub

```

GetProjectTitle, PmeDocCmdTarget Function

Syntax GetProjectTitle()

Description Restituisce una stringa con il titolo assegnato alla cartella principale nella finestra di progetto.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    Dim sResult As String
    sResult = GetProjectTitle
    MsgBox("GetProjectTitle=" & sResult)
End Sub
```

GetRealTimeDB, PmeDocCmdTarget Function

Syntax GetRealTimeDB()

Description Permette di accedere ai metodi e alle proprietà descritte nella DBVariableCmdTarget, per la gestione del RealTimeDB di Movicon da codice basic.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo DBVariableCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Sub Main
    Dim RealTimeDB As DBVariableCmdTarget
    Set RealTimeDB = GetRealTimeDB
    If RealTimeDB Is Nothing Then Exit Sub
    MsgBox "EnableInUseVarMng is " & RealTimeDB.EnableInUseVarMng
End Sub
```

GetResourcePath, PmeDocCmdTarget Function

Syntax GetResourcePath()

Description Restituisce una stringa contenente la cartella di lavoro nella quale Movicon provvederà a salvare le risorse multiple del progetto (Sinottici, Menù, Acceleratori, ecc.).

Parameter	Description
-----------	-------------

None	None
------	------

Result String

Example:

```
Public Sub Click()
    Dim sResult As String
    sResult = GetResourcePath
    Debug.Print sResult
End Sub
```

GetSiteCode, PmeDocCmdTarget Function

Syntax GetSiteCode(ByRef pSiteCode as Variant)

Description consente di acquisire il codice Site Code nel parametro stringa e prevede un valore di ritorno boolean per indicare l'esito dell'operazione. In caso di fallimento la descrizione dell'errore viene riportato nel file CrypKey.



Questa funzione non è supportata in Windows CE.

Parameter	Description
pSiteCode As Variant	Codice site code

Result boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim vSiteCode As Variant
    If Not GetSiteCode(vSiteCode) Then MsgBox "GetSiteCode() problems! Read the CrypKey log file please!"
    If Not IsEmpty(vSiteCode) Then MsgBox "Soft Key: " & CStr(vSiteCode)
End Sub
```

GetScaling, PmeDocCmdTarget Function

Syntax GetScaling(_lpszScalingName)

Description Restituisce l'oggetto Normalizzazione Variabili identificato dal parametro lpszScalingName.

Parameter	Description
lpszScalingName As String	Nome dell'oggetto normalizzazione variabili da reperire.

Result Object
Viene restituito un oggetto di tipo ScalingCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()  
    Dim obj As ScalingCmdTarget  
    Dim bResult As Boolean  
    Set obj = GetScaling("Scaling")  
    bResult = obj.Enabled  
    Debug.Print bResult  
    Set obj = Nothing  
End Sub
```

GetScheduler, PmeDocCmdTarget Function

Syntax GetScheduler(_IpszSchedulerName)

Description Restituisce l'oggetto Scheduler Comandi identificato dal parametro IpszScalingName.

Parameter		Description
IpszSchedulerName	As String	Nome dell'oggetto scheduler comandi da reperire.

Result Object
Viene restituito un oggetto di tipo SchedulerCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()  
    Dim obj As SchedulerCmdTarget  
    Dim bResult As Boolean  
    Set obj = GetScheduler("Scaling")  
    bResult = obj.Enabled  
    Debug.Print bResult  
    Set obj = Nothing  
End Sub
```

GetSynopticInterface, PmeDocCmdTarget Function

Syntax GetSynopticInterface()

Description Permette di accedere ai metodi e alle proprietà descritte nella SynopticCmdTarget da risorsa script.

Tramite l'uso delle funzioni 'GetSynopticInterface' e 'GetSubObject' è possibile creare dei riferimenti ad un sinottico, e ad un oggetto all'interno del sinottico. Occorre prestare molta attenzione ad usare queste funzioni all'interno di una risorsa basic script, perchè essendo un contesto esterno al sinottico, potrebbe accadere che degli oggetti di tipo 'SynopticCmdTarget' o 'DrawCmdTarget' siano stati creati precedentemente in un determinato sinottico caricato in memoria, e vengano poi utilizzati quando il sinottico non è più in memoria. Quando questo accade, il basic script va in errore restituendo il messaggio "ActiveX Automation Error" e la sua esecuzione si interrompe.

Al fine di evitare il presentarsi di questa situazione, si consiglia di evitare l'uso di queste funzioni negli script o quanto meno di gestire l'errore con le funzioni "OnError" del basic script.



Attenzione! Questo metodo non è supportato su piattaforma WinCE. Questo perchè usa delle API DCOM che potrebbero non essere supportate dalla piattaforma WinCE.



L'uso della funzione "GetSynopticInterface" all'interno del codice script dei disegni di un sinottico è fortemente sconsigliato. Utilizzare invece la funzione "GetSynopticObject" dell'interfaccia "DrawCmdTarget".

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo SynopticCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Sub Main
    Dim Screen As SynopticCmdTarget
    Set Screen = GetSynopticInterface("Screen1")
    If Screen Is Nothing Then Exit Sub
    Screen.BackColor = RGB(34,67,234)
End Sub
```

GetUserAndGroup, PmeDocCmdTarget Function

Syntax GetUserAndGroup()

Description Restituisce l'oggetto relativo alla gestione di Utenti e Gruppi Utenti. I metodi e le proprietà dell'oggetto restituito, sono quelli descritti nell'interfaccia UserAndGroupCmdTarget.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo UserAndGroupCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()
    Dim myObject As UserGroupCmdTarget
    Dim bResult As Boolean
    Set myObject = GetUserAndGroup()
    bResult = myObject.EnableAutoLogoff
    Debug.Print bResult
    Set myObject = Nothing
End Sub
```

IsAlarmAreaActive, PmeDocCmdTarget Function

Syntax IsAlarmAreaActive(_IpszServer, _IpszArea)

Description Questo metodo consente di verificare se un'area allarmi contiene degli allarmi attivi. Se il metodo restituisce il valore "false" allora nessun allarme è attivo.

Parameter	Description
IpszServer As String	Nome o indirizzo IP del Server da interrogare. Se la stringa è vuota verrà considerato il Server locale
IpszArea As String	Nome dell'area da controllare.

Result Boolean

Example:

```
Sub Main
    IsAlarmAreaActive("", "Area1")
End Sub
```

IsAlarmAreaON, PmeDocCmdTarget Function

Syntax IsAlarmAreaON(_IpszServer, _IpszArea)

Description Questo metodo consente di verificare se un'area allarmi contiene degli allarmi nello stato ON. Se il metodo restituisce il valore "false" allora nessun allarme è nello stato ON.

Parameter	Description
IpszServer As String	Nome o indirizzo IP del Server da interrogare. Se la stringa è vuota verrà considerato il Server locale
IpszArea As String	Nome dell'area da controllare.

Result Boolean

Example:

```
Sub Main
    IsAlarmAreaON("", "Area1")
End Sub
```

IsChildProject, PmeDocCmdTarget Function

Syntax IsChildProject()

Description Restituisce il valore booleano True se richiamato all'interno di codice di un progetto figlio. Questa funzione può essere utilizzata solo all'interno degli script di tipo risorsa, non può quindi essere utilizzata all'interno del codice script associato ad oggetti sul sinottico.

Parameter	Description
-----------	-------------

None	None
------	------

Result Boolean

Example:

```
Sub Main()
    Dim bResult As Boolean
    bResult = IsChildProject
    Debug.Print "IsChildProject: " & bResult
End Sub
```

IsInStoppingMode, PmeDocCmdTarget Function

Syntax IsInStoppingMode()

Description Questa funzione verifica lo stato di uscita o chiusura applicazione. Solitamente viene usata come test di un loop.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    ...
    While IsInStoppingMode = 0
    ...
    Wend
    ...
End Sub
```

IsRunning, PmeDocCmdTarget Function

Syntax IsRunning()

Description Verifica lo stato di esecuzione del progetto.
Questa funzione può essere invocata da programmi esterni, compreso un altro progetto Movicon, e informa dello stato del progetto: in esecuzione o non in esecuzione.
Il valore di ritorno True indica che il progetto è in esecuzione, mentre False indica che nessun progetto è in esecuzione.

Parameter	Description
None	None

Result Boolean

Example:

```

Sub Main
    Dim mObject As PmeDocCmdTarget
    Set myObject = GetFatherProject("TestPrj")
    'If String Table has already been initialised (Se inizializzata la Tabella delle
    Stringhe)
    Debug.Print "Father.Is Running: " & myObject.IsRunning
    Set mObject = Nothing
End Sub

```

PlaySoundFile, PmeDocCmdTarget Function

Syntax PlaySoundFile(_IpszSoundFile, _bAsync, _bLoop)

Description Tramite questa funzione è possibile eseguire un file audio. Una volta messo in esecuzione si può comunque interrompere la riproduzione del suono tramite la funzione "StopPlaySoundFile".

Parameter	Description
IpszSoundFile As String	Percorso e Nome del file audio da eseguire.
bAsync As Boolean	Definisce se la funzione deve essere eseguita in modo sincrono o asincrono. Impostando il valore a False l'esecuzione della routine basic proseguirà solo dopo che il file audio è stato eseguito completamente. Impostando il valore True invece il file audio sarà messo in esecuzione e la routine basic proseguirà immediatamente ad eseguire le istruzioni successive.
bLoop As Boolean	Definisce se il file audio deve essere eseguito solo una volta o se deve essere eseguito in loop continuo. Impostando il valore a True verrà eseguito in loop continuo. Ad ogni modo il loop verrà eseguito solo se il parametro "bAsync" è stato impostato a True, ovvero l'esecuzione della funzione è asincrona.

Result Boolean

Example:

```

Option Explicit
Sub Main
    PlaySoundFile("C:\FileName.wav", True, False)
End Sub

```

ResetAllAlarms, PmeDocCmdTarget Function

Syntax ResetAllAlarms

Description Esegue il reset di tutti gli allarmi (se precedentemente tacitati con Ack) presenti nel progetto di Movicon.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    ResetAllAlarms
End Sub
```

RunningOnCE, PmeDocCmdTarget Function

Syntax RunningOnCE()

Description Restituisce True se il codice basic è eseguito su una piattaforma Windows Ce.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    If RunningOnCE Then
        MsgBox("This is a WinCe operating system !")
    End If
End Sub
```

RunScript, PmeDocCmdTarget Function

Syntax RunScript(_lpszScriptName, _dwTimeout, _bSpawnNewInstanceAllowed)

Description Esecuzione della funzione Basic Script specificata come parametro e precedentemente inserita nella risorse Basic Script di Movicon.
Il nome del basic script identificato dal parametro lpszScriptName può essere seguito da uno o più parametri suddivisi ciascuno da una virgola. Quindi la funzione GetParameter, descritta nella ScriptMEInterface, servirà per reperire il valore di ciascun parametro passato.

Parameter	Description
lpszScriptName As String	Nome dello script.
dwTimeout As Long	Valore di timeout per l'esecuzione dello script. Questo valore ha senso soltanto se lo script non è impostato come "Thread Separato".
bSpawnNewInstanceAllowed As Boolean	Abilitazione all'esecuzione di più istanze contemporanee dello script. E' comunque necessario che la proprietà "Numero massimo istanze" dello script sia maggiore di uno.

Result Boolean

Example:

```
Public Sub Click()
    ...
    RunScript("BasicScript1",100,False)
    ...
End Sub
```

Example2:

```
Public Sub Click()
    ...
```

```
RunScript("BasicScript1,Param1,Param2",100,False)
...
End Sub
```

SendDispatcherMessage, PmeDocCmdTarget Function

Syntax SendDispatcherMessage(_IpszMedia, _IpszMessage, _IpszUser, _nSeverity)

Description Con questo metodo è possibile inviare al dispatcher una notifica fra quelle supportate dall'AlarmDispatcher. Questo metodo avvia il dispatcher in modo licenziato nel caso in cui non fosse stato già avviato in precedenza.

Parameter	Description
IpszMedia as String	Tipo di notifica da inviare al dispatcher. Può assumere i seguenti valori: "sms", "email", "voice", "smpp", "gsm", "smtp", "mapi" e "fax".
IpszMessage as String	Messaggio da inviare con il dispatcher. Nel caso di invio e-mail il messaggio si arricchisce di una sintassi particolare, che include dati specifici che compaiono nei messaggi di posta elettronica. In caso di invio e-mail è possibile anche allegare uno o più file in attachment. La sintassi e-mail ad esempio è: from~subject~mail message~FilePath1~FilePath2~...~FilePath(n) dove si usa "~" come carattere separatore (es. geronimo@sioux.augh~Subject di prova~Messaggio di prova~C:\MyAttachment.txt).
IpszUser as String	Nome dell'utente del progetto Movicon a cui inviare il messaggio. Movicon recupera da questo utente i dati necessari per notificare al dispatcher a chi inviare il messaggio.
nSeverity as Long	Indice di gravità dell'allarme. Questo numero viene usato dal dispatcher per definire i tempi di invio del messaggio in base alle configurazioni del progetto dell'AlarmDispatcher.

Result Boolean

Example:

```
Option Explicit
Sub Main
```

```
SendDispatcherMessage("sms", "Test message", "User", 1)
```

```
End Sub
```

SetSiteCode, PmeDocCmdTarget Function

Syntax SetSiteCode(ByRef pSiteCode as string)

Description permette di impostare il codice di attivazione Soft Key, che deve essere passato nel parametro stringa, e prevede un valore di ritorno boolean per indicare l'esito dell'operazione. In caso di fallimento la descrizione dell'errore viene riportato nel file CrypKey.



Questa funzione non è supportata in Windows CE.

Parameter	Description
-----------	-------------

pSiteKey As String	Codice Soft Key
--------------------	-----------------

Result boolean

Example:

```
Option Explicit
Public Sub Click()
Dim SoftKey As string
SoftKey = "XXXX XXXX XXXX XXXX XXXX XXXX XX"
If SoftKey<>"" Then
If Not SetSiteKey(SoftKey) Then
MsgBox "It was not possible to set the SoftKey!"
Else
MsgBox "SoftKey set with successfull!"
End If
End If
End Sub
```

StartAlarmDispatcher, PmeDocCmdTarget Function

Syntax StartAlarmDispatcher

Description Con questo metodo è possibile eseguire l'AlarmDispatcher in modo licenziato. Ovviamente l'AlarmDispatcher viene eseguito soltanto se la corrispondente opzione nella licenza è presente, altrimenti viene emesso il messaggio:

"Your license restriction doesn't allow to run the Alarm Dispatcher"

Il metodo è utile per chi vuole usare l'AlarmDispatcher in modo personalizzato con il codice VBA senza usare le funzioni degli allarmi di Movicon.

NOTA: Movicon mantiene una referenza all'oggetto che l'AlarmDispatcher crea nella ROT. Quindi il metodo può essere usato un'unica volta, dopo l'esecuzione del progetto, per avviare il dispatcher.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Sub Main
StartAlarmDispatcher
End Sub
```

StopPlaySoundFile, PmeDocCmdTarget Function

Syntax StopPlaySoundFile()

Description Tramite questa funzione è possibile interrompere la riproduzione di un file audio eseguito tramite la funzione "PlaySoundFile".

Parameter	Description
None	None

Result Boolean

Example:

Option Explicit

Sub Main

 PlaySoundFile("C:\FileName.wav", True, True)

 ...

StopPlaySoundFile

End Sub

UnloadScript, PmeDocCmdTarget Function

Syntax UnloadScript(_IpszScriptName)

Description Termina lo script identificato dal parametro IpszScriptName.

Parameter	Description
IpszScriptName As String	Nome dello script.

Result Boolean

Example:

Public Sub Click()

 ...

UnloadScript("Basic Script1")

 ...

End Sub

Properties

ActiveLanguage, PmeDocCmdTarget Property

Syntax ActiveLanguage = _String

Description Permette di impostare o leggere la lingua attiva.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sLanguage As String
    sLanguage = ActiveLanguage
    Debug.Print sLanguage
End Sub
```

ChildProjectActiveNetworkServer, PmeDocCmdTarget Property

Syntax ChildProjectActiveNetworkServer = _String

Description Questa proprietà restituisce il nome o l'indirizzo IP del Server di Networking attivo dal quale il progetto figlio andrà a recuperare i dati. La proprietà è di sola lettura.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    MsgBox "Active Child Network Server = " &
        GetChildProject("ChildProject1").ChildProjectActiveNetworkServer,
        vbInformation, GetProjectTitle
End Sub
```

ChildProjectBackupNetworkServer, PmeDocCmdTarget Property

Syntax ChildProjectBackupNetworkServer = _String

Description Questa proprietà restituisce il nome o l'indirizzo IP del Server di Networking di BackUp dal quale il progetto figlio andrà a recuperare i dati. La proprietà è di sola lettura.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    MsgBox "Child Network Backup Server = " &
        GetChildProject("ChildProject1").ChildProjectBackupNetworkServer,
        vbInformation, GetProjectTitle
End Sub

```

ChildProjectName, PmeDocCmdTarget Property

Syntax ChildProjectName = _String

Description Questa proprietà restituisce il nome del progetto figlio. Nel caso in cui nella proprietà "Nome Progetto Figlio" del gruppo "Opzione Progetto Figlio" non sia stato inserito nessun valore, allora la proprietà restituirà una stringa vuota. La proprietà è di sola lettura.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    MsgBox "Child Project Name = " &
        GetChildProject("ChildProject1").ChildProjectName, vbInformation,
        GetProjectTitle
End Sub

```

ChildProjectNetworkServer, PmeDocCmdTarget Property

Syntax ChildProjectNetworkServer = _String

Description Questa proprietà restituisce il nome o l'indirizzo IP del Server di Networkin dal quale il progetto figlio andrà a recuperare i dati. La proprietà è di sola lettura.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    MsgBox "Child Network Server = " &
        GetChildProject("ChildProject1").ChildProjectNetworkServer, vbInformation,
        GetProjectTitle
End Sub

```

ChildProjectStartable, PmeDocCmdTarget Property

Syntax ChildProjectStartable = _Boolean

Description Questa proprietà consente di verificare se il progetto figlio è "Eseguibile" oppure no. Se un progetto figlio è eseguibile significa che verrà avviata tutta la parte Server del progetto figlio, ovvero Driver, OPC, Networking, ecc.. Se un progetto non è eseguibile, dal progetto padre sarà comunque possibile accedere ad esempio ai sinottici del figlio. La proprietà è di sola lettura.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    MsgBox "Child Project Startable = " &  
        GetChildProject("ChildProject1").ChildProjectStartable, vbInformation,  
        GetProjectTitle  
End Sub
```

HisLogAlarmDurationDays, PmeDocCmdTarget Property

Syntax HisLogAlarmDurationDays = _Long

Description Permette di impostare o leggere per quanto tempo i messaggi di Allarme dovranno essere mantenuti all'interno del database. L'impostazione di Default è di 180 giorni, ma può essere modificata in base alle proprie esigenze, tenendo anche conto di quante registrazioni verranno fatte nell'intervallo di tempo specificato.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim sResult As Long  
    sResult = HisLogAlarmDurationDays  
    Debug.Print sResult  
End Sub
```

HisLogAlarmDurationHours, PmeDocCmdTarget Property

Syntax HisLogAlarmDurationHours = _Long

Description Permette di impostare o leggere per quanto tempo aggiuntivo in ore i messaggi di Allarme dovranno essere mantenuti all'interno del database.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim sResult As Long  
    sResult = HisLogAlarmDurationHours  
    Debug.Print sResult  
End Sub
```

HisLogAlarmDurationMinutes, PmeDocCmdTarget Property

Syntax HisLogAlarmDurationMinutes = _Long

Description Permette di impostare o leggere per quanto tempo aggiuntivo in minuti i messaggi di Allarme dovranno essere mantenuti all'interno del database.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim sResult As Long  
    sResult = HisLogAlarmDurationMinutes  
    Debug.Print sResult  
End Sub
```

HisLogAlarmTable, PmeDocCmdTarget Property

Syntax HisLogAlarmTable = _String

Description Imposta o restituisce il nome della tabella del Log Storico che conterrà i messaggi inerenti agli "Allarmi" del progetto.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sResult As String
    sResult = HisLogAlarmTable
    Debug.Print sResult
End Sub
```

HisLogCommentColName, PmeDocCmdTarget Property

Syntax HisLogCommentColName

Description Questa proprietà imposta o restituisce il nome della Colonna Commento delle tabelle del Log Storico.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sResult As String
    sResult = HisLogCommentColName
    Debug.Print sResult
End Sub
```

HisLogDefVarCharPrecision, PmeDocCmdTarget Property

Syntax HisLogDefVarCharPrecision = _Long

Description Imposta o restituisce la massima precisione per le colonne di tipo stringa. Il numero impostato rappresenta il numero di caratteri della stringa.

Parameter	Description
-----------	-------------

None	None
------	------

Result Long

Example:

```
Public Sub Click()
    Dim sResult As Long
    sResult = HisLogDefVarCharPrecision
    Debug.Print sResult
End Sub
```

HisLogDescriptionColName, PmeDocCmdTarget Property

Syntax HisLogDescriptionColName

Description Questa proprietà imposta o restituisce il nome della Colonna Descrizione delle tabelle del Log Storico. Se non viene specificato nulla verrà utilizzato il nome di default. La Colonna Descrizione riporta la descrizione dell'evento di quella registrazione.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sResult As String
    sResult = HisLogDescriptionColName
    Debug.Print sResult
End Sub
```

HisLogDriverDurationDays, PmeDocCmdTarget Property

Syntax HisLogDriverDurationDays = _Long

Description Permette di impostare o leggere per quanto tempo i messaggi dei Driver di Comunicazione dovranno essere mantenuti all'interno del database. L'impostazione di Default è di 180 giorni, ma può essere modificata in base alle proprie esigenze, tenendo anche conto di quante registrazioni verranno fatte nell'intervallo di tempo specificato.

Parameter	Description
-----------	-------------

None	None
------	------

Result Long

Example:

```
Public Sub Click()
    Dim sResult As Long
    sResult = HisLogDriverDurationDays
    Debug.Print sResult
End Sub
```

HisLogDriverDurationHours, PmeDocCmdTarget Property

Syntax HisLogDriverDurationHours = _Long

Description Permette di impostare o leggere per quanto tempo aggiuntivo in ore i messaggi dei Driver di Comunicazione dovranno essere mantenuti all'interno del database.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
    Dim sResult As Long
    sResult = HisLogDriverDurationHours
    Debug.Print sResult
End Sub
```

HisLogDriverDurationMinutes, PmeDocCmdTarget Property

Syntax HisLogDriverDurationMinutes = _Long

Description Permette di impostare o leggere per quanto tempo aggiuntivo in minuti i messaggi di Allarme dovranno essere mantenuti all'interno del database.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
    Dim sResult As Long
    sResult = HisLogDriverDurationMinutes
```

```
        Debug.Print sResult
End Sub
```

HisLogDriverTable, PmeDocCmdTarget Property

Syntax HisLogDriverTable = _String

Description Imposta o restituisce il nome della tabella del Log Storico che conterrà i messaggi inerenti ai "Driver" del progetto.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sResult As String
    sResult = HisLogDriverTable
    Debug.Print sResult
End Sub
```

HisLogDsn, PmeDocCmdTarget Property

Syntax HisLogDsn = _String

Description Questa impostazione permette di definire o reperire un eventuale collegamento **ODBC** personalizzato per il Log Storico. Di default Movicon creerà un DSN con lo stesso nome del progetto e suffisso "_HisLog", configurato per accedere al database specificato tramite il "PlugIn ODBC di Default" del progetto. Il nome del DSN sarà del tipo:

NomeProgetto_HisLog

Tramite questa proprietà è però possibile personalizzare il collegamento **ODBC**, creando una base dati personalizzata anche differente da quella definita nel "PlugIn ODBC di Default" del progetto.



Questa proprietà può essere utilizzata in scrittura solo in fase di design del progetto, ad esempio nel dropping code di un simbolo Template. In fase di Runtime invece, anche se può essere modificata, non verrà applicata al Log Storico, che continuerà a registrare utilizzando il DSN iniziale.

Parameter	Description
-----------	-------------

None	None
------	------

Result String

Example:

```
Public Sub Click()
    Dim sResult As String
    sResult = HisLogDsn
    Debug.Print sResult
End Sub
```

HisLogDurationColName, PmeDocCmdTarget Property

Syntax HisLogDurationColName

Description Questa proprietà imposta o restituisce il nome della Colonna Durata delle tabelle del Log Storico. Se non viene specificato nulla verrà utilizzato il nome di default. La Colonna Durata riporta il tempo di durata dell'evento in questione.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sResult As String
    sResult = HisLogDurationColName
    Debug.Print sResult
End Sub
```

HisLogEventTypeColName, PmeDocCmdTarget Property

Syntax HisLogEventTypeColName

Description Questa proprietà imposta o restituisce il nome della Colonna Tipo Evento delle tabelle del Log Storico. Se non viene specificato nulla verrà utilizzato il nome di default. La Colonna Tipo Evento indica il tipo di evento di quella registrazione (Es. Allarme ON, Allarme OFF, Sistema, ecc



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la

proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sResult As String  
    sResult = HisLogEventTypeColName  
    Debug.Print sResult  
End Sub
```

HisLogEventNumColName, PmeDocCmdTarget Property

Syntax HisLogEventNumColName

Description Questa proprietà imposta o restituisce il nome della Colonna Numero Evento delle tabelle del Log Storico. Se non viene specificato nulla verrà utilizzato il nome di default. La Colonna Numero Evento riporta un numero identificativo del tipo di evento di quella registrazione.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sResult As String  
    sResult = HisLogEventNumColName  
    Debug.Print sResult  
End Sub
```

HisLogLocalTimeColName, PmeDocCmdTarget Property

Syntax HisLogLocalTimeColName

Description Questa proprietà imposta o restituisce il nome della Colonna Tempo Locale delle tabelle del Log Storico. Se non viene specificato nulla verrà utilizzato il nome di default. La Colonna Tempo Locale indica la data e l'ora della registrazione tenendo come riferimento l'orario locale.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sResult As String  
    sResult = HisLogLocalTimeColName  
    Debug.Print sResult  
End Sub
```

HisLogMaxCacheBeforeFlush, PmeDocCmdTarget Property

Syntax HisLogMaxCacheBeforeFlush = _Long

Description Imposta o restituisce la massima dimensione della Cache prima che il sistema scarichi i dati su file. Il numero impostato è espresso in Byte.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim sResult As Long  
    sResult = HisLogMaxCacheBeforeFlush  
    Debug.Print sResult  
End Sub
```

HisLogMaxError, PmeDocCmdTarget Property

Syntax HisLogMaxError = _Long

Description Imposta o restituisce il numero massimo di errori DBMS dopo i quali la connessione viene considerata non valida e i dati vengono salvati su file in formato ASCII nelle cartelle relative del progetto ("DLOGGERS", "LOGS", "DATA").

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim sResult As Long  
    sResult = HisLogMaxError  
    Debug.Print sResult  
End Sub
```

HisLogMaxNumberTrans, PmeDocCmdTarget Property

Syntax HisLogMaxNumberTrans = _Long

Description Imposta o restituisce il numero massimo di transizioni per ciclo da aggiornare prima della loro chiusura.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim sResult As Long  
    sResult = HisLogMaxNumberTrans  
    Debug.Print sResult  
End Sub
```

HisLogMSecColName, PmeDocCmdTarget Property

Syntax HisLogMSecColName

Description Questa proprietà imposta o restituisce il nome della Colonna MSec delle tabelle del Log Storico. Se non viene specificato nulla verrà utilizzato il nome di default. La Colonna MSec indica i millisecondi relativi all'orario di registrazione.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sResult As String  
    sResult = HisLogMSecColName  
    Debug.Print sResult  
End Sub
```

HisLogRecycleDBConnection, PmeDocCmdTarget Property

Syntax HisLogRecycleDBConnection = _Boolean

Description Imposta o restituisce il valore della proprietà "Mantieni aperta la Connessione DB". Se impostata a true, permette di mantenere aperta la connessione DBMS, e quindi di utilizzare la stessa connessione per tutte le transizioni che verranno eseguite. Se invece la proprietà viene disabilitata, la connessione DBMS verrà aperta quando viene richiesta una transizione e poi verrà chiusa.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim sResult As Boolean  
    sResult = HisLogRecycleDBConnection  
    Debug.Print sResult  
End Sub
```

HisLogSubEventTypeColName, PmeDocCmdTarget Property

Syntax HisLogSubEventTypeColName

Description Questa proprietà imposta o restituisce il nome della Colonna Sub Evento delle tabelle del Log Storico. Se non viene specificato nulla verrà utilizzato il nome di default. La

Colonna Sub Evento riporta il valore associato alla variabile di sistema "_Sys_HisLogSubEvent_" se questa è presente nel Real Time DB.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sResult As String  
    sResult = HisLogSubEventTypeColName  
    Debug.Print sResult  
End Sub
```

HisLogSysTable, PmeDocCmdTarget Property

Syntax HisLogSysTable = _String

Description Iimposta o restituisce il nome della tabella del Log Storico che conterrà i messaggi inerenti ai "Messaggi di Sistema" del progetto.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sResult As String  
    sResult = HisLogSysTable  
    Debug.Print sResult  
End Sub
```


HisLogSystemsDurationDays, PmeDocCmdTarget Property

Syntax HisLogSystemDurationDays = _Long

Description Permette di impostare o leggere per quanto tempo i messaggi di Sistema dovranno essere mantenuti all'interno del database. L'impostazione di Default è di 180 giorni, ma può essere modificata in base alle proprie esigenze, tenendo anche conto di quante registrazioni verranno fatte nell'intervallo di tempo specificato.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim sResult As Long  
    sResult = HisLogSystemDurationDays  
    Debug.Print sResult  
End Sub
```

HisLogSystemsDurationHours, PmeDocCmdTarget Property

Syntax HisLogSystemDurationHours = _Long

Description Permette di impostare o leggere per quanto tempo aggiuntivo in ore i messaggi di Sistema dovranno essere mantenuti all'interno del database.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim sResult As Long  
    sResult = HisLogSystemDurationHours  
    Debug.Print sResult  
End Sub
```

HisLogSystemsDurationMinutes, PmeDocCmdTarget Property

Syntax HisLogSystemDurationMinutes = _Long

Description Permette di impostare o leggere per quanto tempo aggiuntivo in minuti i messaggi di Sistema dovranno essere mantenuti all'interno del database.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
    Dim sResult As Long
    sResult = HisLogSystemDurationMinutes
    Debug.Print sResult
End Sub
```

HisLogTimeColName, PmeDocCmdTarget Property

Syntax HisLogTimeColName

Description Questa proprietà imposta o restituisce il nome della Colonna Tempo Locale delle tabelle del Log Storico. Se non viene specificato nulla verrà utilizzato il nome di default. La Colonna Tempo Locale indica la data e l'ora della registrazione tenendo come riferimento l'orario locale.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sResult As String
    sResult = HisLogTimeColName
    Debug.Print sResult
End Sub
```

HisLogTransactionIDColName, PmeDocCmdTarget Property

Syntax HisLogTransactionIDColName

Description Questa proprietà imposta o restituisce il nome della Colonna Transaction ID delle tabelle del Log Storico.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sResult As String  
    sResult = HisLogTransactionIDColName  
    MsgBox "HisLogTransactionIDColName = " & sResult, vbInformation,  
    GetProjectTitle  
End Sub
```

HisLogUseIMDB, PmeDocCmdTarget Property

Syntax HisLogUseIMDB = _Boolean

Description Questa proprietà di sola lettura consente di sapere se il log storico del progetto è impostato per registrare i valori attraverso il motore InMemoryDataBase.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main  
    MsgBox "HisLogUseIMDB->" & HisLogUseIMDB  
End Sub
```

HisLogUser, PmeDocCmdTarget Property

Syntax HisLogUser = _String

Description Imposta o restituisce il nome dell'utente utilizzato per la connessione ODBC.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sResult As String  
    sResult = HisLogUser  
    Debug.Print sResult  
End Sub
```

HisLogUserColName, PmeDocCmdTarget Property

Syntax HisLogUserColName

Description Questa proprietà imposta o restituisce il nome della Colonna Utente delle tabelle del Log Storico. Se non viene specificato nulla verrà utilizzato il nome di default. La Colonna Utente indica il nome dell'utente attivo al momento della registrazione.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sResult As String  
    sResult = HisLogUserColName  
    Debug.Print sResult  
End Sub
```

HisLogUniqueIDColName, PmeDocCmdTarget Property

Syntax HisLogUniqueIDColName

Description Questa proprietà imposta o restituisce il nome della Colonna Unique ID delle tabelle del Log Storico.



La scrittura di questa proprietà verrà acquisita soltanto se viene fatta durante la modalità di sviluppo del progetto, ad esempio tramite il Dropping Code di un Simbolo. In questo caso infatti la proprietà verrà modificata staticamente nel progetto. Una volta invece che il progetto è stato avviato in Runtime, la modifica di tale proprietà non verrà considerata.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sResult As String
    sResult = HisLogUniqueIDColName
    MsgBox "HisLogTransactionIDColName = " & sResult, vbInformation,
    GetProjectTitle
End Sub
```

ShutdownScript, PmeDocCmdTarget Property

Syntax ShutdownScript = _String

Description Imposta o restituisce il nome dello script da eseguire alla chiusura del progetto.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sResult As String
    sResult = ShutdownScript
    Debug.Print sResult
End Sub
```

StartChildProjectWithFather, PmeDocCmdTarget Property

Syntax StartChildProjectWithFather = _Boolean

Description Questa proprietà consente di verificare se il progetto figlio è verrà avviato insieme al padre oppure no. Questa proprietà, se è abilitata, permette infatti di avviare automaticamente il progetto figlio quando viene messo in esecuzione il progetto padre. Perchè questa impostazione abbia effetto deve essere stata selezionata anche la proprietà "Eseguibile". La proprietà è di sola lettura.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    MsgBox "Child Project Start with Father = " &  
        GetChildProject("ChildProject1").StartChildProjectWithFather, vbInformation,  
        GetProjectTitle  
End Sub
```

StartupScreen, PmeDocCmdTarget Property

Syntax StartupScreen = _String

Description Imposta o restituisce il nome del sinottico di avvio.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sResult As String  
    sResult = StartupScreen  
    Debug.Print sResult  
End Sub
```

StartupScript, PmeDocCmdTarget Property

Syntax StartupScript = _String

Description Imposta o restituisce il nome dello script di avvio.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sResult As String  
    sResult = StartupScript  
    Debug.Print sResult  
End Sub
```

StoreCryptProject, PmeDocCmdTarget Property

Syntax StoreCryptProject = _Boolean

Description Imposta o restituisce il valore della proprietà 'Progetto Criptato'. Se impostata a true il progetto verrà salvato in un formato criptato, e quindi non sarà possibile aprirlo con un editor differente da Movicon.



L'impostazione di questa proprietà in fase di runtime non ha nessun effetto. Se però la proprietà viene utilizzata in un basic script lanciato dal design la proprietà nel progetto assumerà il nuovo valore impostato.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim sResult As Boolean  
    sResult = StoreCryptProject  
    Debug.Print sResult  
End Sub
```

StoreCryptProjectResources, PmeDocCmdTarget Property

Syntax StoreCryptProjectResources = _Boolean

Description Imposta o restituisce il valore della proprietà 'Risorse del Progetto Criptate'. Se impostata a true, i file delle risorse del progetto verranno salvati in un formato criptato, e quindi non sarà possibile aprirli con un editor differente da Movicon.



L'impostazione di questa proprietà in fase di runtime non ha nessun effetto. Se però la proprietà viene utilizzata in un basic script lanciato dal design la proprietà nel progetto assumerà il nuovo valore impostato.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim sResult As Boolean  
    sResult = StoreCryptProjectResources  
    Debug.Print sResult  
End Sub
```

StoreCryptProjectStrings, PmeDocCmdTarget Property

Syntax StoreCryptProjectStrings = _Boolean

Description Imposta o restituisce il valore della proprietà "Tabella Stringhe Criptata". Se impostata a true, i file della Tabella Stringhe del progetto verranno salvati in un formato criptato, e quindi non sarà possibile aprirli con un editor differente da Movicon.



L'impostazione di questa proprietà in fase di runtime non ha nessun effetto. Se però la proprietà viene utilizzata in un basic script lanciato dal design la proprietà nel progetto assumerà il nuovo valore impostato.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim sResult As Boolean
    sResult = StoreCryptProjectStrings
    Debug.Print sResult
End Sub
```

StoreUnicodeProject, PmeDocCmdTarget Property

Syntax StoreUnicodeProject = _Boolean

Description Imposta o restituisce il valore della proprietà 'Progetto Unicode'. Se impostata a true, tutto il progetto sarà salvato in formato Unicode UTF-16.



L'impostazione di questa proprietà in fase di runtime non ha nessun effetto. Se però la proprietà viene utilizzata in un basic script lanciato dal design la proprietà nel progetto assumerà il nuovo valore impostato.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim sResult As Boolean
    sResult = StoreUnicodeProject
    Debug.Print sResult
End Sub
```


StoreZippedProject, PmeDocCmdTarget Property

Syntax StoreZippedProject = _Boolean

Description Imposta o restituisce il valore della proprietà 'Progetto Compresso'. Se impostata a true, tutto il progetto sarà salvato in formato compresso. La compressione e decompressione dei file verrà gestita automaticamente da Movicon in real-time, e quindi sarà trasparente per il programmatore.



L'impostazione di questa proprietà in fase di runtime non ha nessun effetto. Se però la proprietà viene utilizzata in un basic script lanciato dal design la proprietà nel progetto assumerà il nuovo valore impostato.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim sResult As Boolean  
    sResult = StoreZippedProject  
    Debug.Print sResult  
End Sub
```

StringFromID, PmeDocCmdTarget Property

Syntax StringFromID(_IpszID) = _String

Description Restituisce la stringa in funzione della colonna attiva (Lingua) nella risorsa stringa, specificando l'identificatore ID della medesima.

Parameter	Description
IpszID As String	Identificatore della stringa richiesta.

Result String

Example:

```
Sub Main  
    Dim test As string  
    test = StringFromID ("STR0001")  
    MsgBox test, 64,"String"  
    StringFromID ("STR0001") = "Modificata"  
    test = StringFromID ("STR0001")  
    MsgBox test, 64,"String"  
End Sub
```

TargetClientJ2ME, PmeDocCmdTarget Property

Syntax TargetClientJ2ME = _Boolean

Description Imposta o restituisce il valore della proprietà 'Client J2ME'. Se impostata a true, abilita lo sviluppo del progetto per piattaforme J2ME Client (JavaPhones).
Nota: La selezione multipla delle piattaforme comporta sempre le funzionalità della piattaforma maggiore.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim sResult As Boolean  
    sResult = TargetClientJ2ME  
    Debug.Print sResult  
End Sub
```

TargetClientJ2SE, PmeDocCmdTarget Property

Syntax TargetClientJ2SE = _Boolean

Description Imposta o restituisce il valore della proprietà 'Client J2SE'. Se impostata a true, abilita lo sviluppo del progetto per piattaforme J2SE Client (Linux).
Nota: La selezione multipla delle piattaforme comporta sempre le funzionalità della piattaforma maggiore.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim sResult As Boolean  
    sResult = TargetClientJ2SE  
    Debug.Print sResult  
End Sub
```

TargetClientWin32, PmeDocCmdTarget Property

Syntax TargetClientWin32 = _Boolean

Description Imposta o restituisce il valore della proprietà "Client Win32/64 bit". Se impostata a true, abilita lo sviluppo del progetto per piattaforme Windows 32/64 bit Client.
Nota: La selezione multipla delle piattaforme comporta sempre le funzionalità della piattaforma maggiore.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim sResult As Boolean
    sResult = TargetClientWin32
    Debug.Print sResult
End Sub
```

TargetClientWinCE, PmeDocCmdTarget Property

Syntax TargetClientWinCE = _Boolean

Description Imposta o restituisce il valore della proprietà 'Client WinCE'. Se impostata a true, abilita lo sviluppo del progetto per piattaforme Windows CE Client.
Nota: La selezione multipla delle piattaforme comporta sempre le funzionalità della piattaforma maggiore.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim sResult As Boolean
    sResult = TargetClientWinCE
    Debug.Print sResult
End Sub
```

TargetPlatformWin32, PmeDocCmdTarget Property

Syntax TargetPlatformWin32 = _Boolean

Description Imposta o restituisce il valore della proprietà "Windows 32/64 bit". Se impostata a true, abilita lo sviluppo del progetto per piattaforme Windows 32/64 bit.
Nota: La selezione multipla delle piattaforme comporta sempre le funzionalità della piattaforma maggiore.

Parameter	Description
-----------	-------------

None	None
------	------

Result Boolean

Example:

```
Public Sub Click()
    Dim sResult As Boolean
    sResult = TargetPlatformWin32
    Debug.Print sResult
End Sub
```

TargetPlatformWinCE, PmeDocCmdTarget Property

Syntax TargetPlatformWinCE = _Boolean

Description Imposta o restituisce il valore della proprietà 'WinCE'. Se impostata a true, abilita lo sviluppo del progetto per piattaforme Windows CE.
Nota: La selezione multipla delle piattaforme comporta sempre le funzionalità della piattaforma maggiore.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim sResult As Boolean
    sResult = TargetPlatformWinCE
    Debug.Print sResult
End Sub
```

1.17.1. RASStationInterface

Functions

GetXMLSettings, RASStationInterface Function

Syntax GetXMLSettings()

Description Questa funzione restituisce la stringa di definizione, in formato XML, della Stazione RAS inserita nel progetto.



Attenzione! Questa funzione è disponibile anche nell'interfaccia Basic Script dei Driver di Comunicazione. I due metodi non vanno confusi perchè anche se svolgono la stessa funzione vengono utilizzati in due contesti differenti.

Parameter	Description
-----------	-------------

None	None
------	------

Result String

Example:

```
Dim RasObj As RASStationInterface
Dim NetwObj As NetworkClientCmd
Public Sub Click()
    If NetwObj Is Nothing Then Set NetwObj = GetNetworkClient
    If RasObj Is Nothing And Not NetwObj Is Nothing Then Set RasObj =
        NetwObj.GetRASStation("FirstRAS")
    If Not RasObj Is Nothing Then
        MsgBox RasObj.GetXMLSettings,vbOkOnly,GetProjectTitle
    End If
End Sub
```

Properties

ConnectionVariable, RASStationInterface Property

Syntax ConnectionVariable = _String

Description Questa proprietà consente di leggere o impostare la variabile di connessione utilizzata da una stazione RAS per gestire la chiamata su comando.

Parameter	Description
None	None

Result String

Example:

```
Sub Main
    Dim objRAS As RASStationInterface
    Dim sNewVariable As String

    Set objRAS = GetNetworkClient.GetRASStation("RAS Station00001")
    GetVariableNameFromList(sNewVariable)
    Debug.Print "Before->" & objRAS.ConnectionVariable
    objRAS.ConnectionVariable = sNewVariable
    Debug.Print "Next->" & objRAS.ConnectionVariable
End Sub
```

DisconnectAfterSecs, RASStationInterface Property

Syntax DisconnectAfterSecs = _Long

Description Questa proprietà consente di definire dopo quanto tempo di inattività la connessione deve essere automaticamente chiusa. Il conteggio del tempo inizia dal momento in cui tutte le variabili connesse al Server non sono più in uso.



Attenzione! Questa proprietà è disponibile anche nell'interfaccia Basic Script dei Driver di Comunicazione. Le due proprietà non vanno confuse perchè anche se svolgono la stessa funzione vengono utilizzati in due contesti differenti.

Parameter	Description
None	None

Result Long

Example:

```
Dim RasObj As RASStationInterface
Dim NetwObj As NetworkClientCmd
Public Sub Click()
    If NetwObj Is Nothing Then Set NetwObj = GetNetworkClient
    If RasObj Is Nothing And Not NetwObj Is Nothing Then Set RasObj =
        NetwObj.GetRASStation("FirstRAS")
    If Not RasObj Is Nothing Then
        RasObj.DisconnectAfterSecs = 10
    End If
End Sub
```

EndConnectionTime, RASStationInterface Property

Syntax EndConnectionTime = _Date

Description Questa proprietà restituisce l'ora e il giorno in cui è terminata connessione.



Attenzione! Questa proprietà è disponibile anche nell'interfaccia Basic Script dei Driver di Comunicazione. Le due proprietà non vanno confuse perchè anche se svolgono la stessa funzione vengono utilizzati in due contesti differenti.

Parameter	Description
None	None

Result Date

Example:

```
Dim RasObj As RASStationInterface
Dim NetwObj As NetworkClientCmd
Public Sub Click()
    If NetwObj Is Nothing Then Set NetwObj = GetNetworkClient
    If RasObj Is Nothing And Not NetwObj Is Nothing Then Set RasObj =
        NetwObj.GetRASStation("FirstRAS")
    If Not RasObj Is Nothing Then
        Debug.Print RasObj.EndConnectionTime
    End If
End Sub
```

IsConnected, RASStationInterface Property

Syntax IsConnected = _Boolean

Description Questa proprietà restituisce lo stato della connessione RAS referenziata. Restituisce il valore booleano True se la connessione è stabilita.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim RasObj As RASStationInterface
Dim NetwObj As NetworkClientCmd
Public Sub Click()
    If NetwObj Is Nothing Then Set NetwObj = GetNetworkClient
    If RasObj Is Nothing And Not NetwObj Is Nothing Then Set RasObj =
        NetwObj.GetRASStation("FirstRAS")
    If Not RasObj Is Nothing Then
        RasObj.DisconnectAfterSecs = 10
    End If
End Sub
```

LastConnectionTime, RASStationInterface Property

Syntax LastConnectionTime = _Date

Description Questa proprietà restituisce l'ora e il giorno dell'ultima connessione stabilita.



Attenzione! Questa proprietà è disponibile anche nell'interfaccia Basic Script dei Driver di Comunicazione. Le due proprietà non vanno confuse perchè anche se svolgono la stessa funzione vengono utilizzati in due contesti differenti.

Parameter	Description
None	None

Result Date

Example:

```
Dim RasObj As RASStationInterface
Dim NetwObj As NetworkClientCmd
Public Sub Click()
    If NetwObj Is Nothing Then Set NetwObj = GetNetworkClient
    If RasObj Is Nothing And Not NetwObj Is Nothing Then Set RasObj =
        NetwObj.GetRASStation("FirstRAS")
    If Not RasObj Is Nothing Then
        Debug.Print RasObj.LastConnectionTime
    End If
End Sub
```

LastRASErrorNumber, RASStationInterface Property

Syntax LastRASErrorNumber = _Long

Description Questa proprietà restituisce il numero dell'ultimo errore verificatosi per la connessione RAS referenziata.



Attenzione! Questa proprietà è disponibile anche nell'interfaccia Basic Script dei Driver di Comunicazione. Le due proprietà non vanno confuse perchè anche se svolgono la stessa funzione vengono utilizzati in due contesti differenti.

Parameter	Description
None	None

Result Long

Example:

```
Dim RasObj As RASStationInterface
Dim NetwObj As NetworkClientCmd
Public Sub Click()
    If NetwObj Is Nothing Then Set NetwObj = GetNetworkClient
    If RasObj Is Nothing And Not NetwObj Is Nothing Then Set RasObj =
        NetwObj.GetRASStation("FirstRAS")
    If Not RasObj Is Nothing Then
        Debug.Print RasObj.LastRASErrorNumber
    End If
End Sub
```

LastRASErrorString, RASStationInterface Property

Syntax LastRASErrorString = _String

Description Questa proprietà restituisce la stringa di definizione dell'ultimo errore verificatosi per la connessione RAS referenziata.



Attenzione! Questa proprietà è disponibile anche nell'interfaccia Basic Script dei Driver di Comunicazione. Le due proprietà non vanno confuse perchè anche se svolgono la stessa funzione vengono utilizzati in due contesti differenti.

Parameter	Description
None	None

Result String

Example:

```
Dim RasObj As RASStationInterface
Dim NetwObj As NetworkClientCmd
Public Sub Click()
    If NetwObj Is Nothing Then Set NetwObj = GetNetworkClient
    If RasObj Is Nothing And Not NetwObj Is Nothing Then Set RasObj =
        NetwObj.GetRASStation("FirstRAS")
```



```

        If Not RasObj Is Nothing Then
            Debug.Print RasObj.LastRASErrorString
        End If
    End Sub

```

NumRetries, RASStationInterface Property

Syntax NumRetries = _Byte

Description Questa proprietà consente di definire il numero massimo di tentativi di connessione che vengono eseguiti in caso di fallimento della chiamata.



Attenzione! Questa proprietà è disponibile anche nell'interfaccia Basic Script dei Driver di Comunicazione. Le due proprietà non vanno confuse perchè anche se svolgono la stessa funzione vengono utilizzati in due contesti differenti.

Parameter	Description
None	None

Result Byte

Example:

```

Dim RasObj As RASStationInterface
Dim NetwObj As NetworkClientCmd
Public Sub Click()
    If NetwObj Is Nothing Then Set NetwObj = GetNetworkClient
    If RasObj Is Nothing And Not NetwObj Is Nothing Then Set RasObj =
        NetwObj.GetRASStation("FirstRAS")
    If Not RasObj Is Nothing Then
        Debug.Print RasObj.NumRetries
    End If
End Sub

```

Password, RASStationInterface Property

Syntax Password = _String

Description Questa proprietà consente di definire la password utente con la quale la stazione Client deve essere autenticata dalla stazione Server. Se la proprietà "Connessione" è stata compilata questo campo può essere lasciato vuoto.



Attenzione! Questa proprietà è disponibile anche nell'interfaccia Basic Script dei Driver di Comunicazione. Le due proprietà non vanno confuse perchè anche se svolgono la stessa funzione vengono utilizzati in due contesti differenti.

Parameter	Description
None	None

Result String

Example:

```

Dim RasObj As RASStationInterface

```

```

Dim NetwObj As NetworkClientCmd
Public Sub Click()
    If NetwObj Is Nothing Then Set NetwObj = GetNetworkClient
    If RasObj Is Nothing And Not NetwObj Is Nothing Then Set RasObj =
        NetwObj.GetRASStation("FirstRAS")
    If Not RasObj Is Nothing Then
        Debug.Print RasObj.Password
    End If
End Sub

```

PhoneBookEntry, RASStationInterface Property

Syntax PhoneBookEntry = _String

Description Questa proprietà consente di inserire il nome dell'eventuale Connessione RAS da utilizzare per connettere il Client al Server. In questo caso la Connessione RAS deve comparire tra le connessioni di rete configurate nel Sistema Operativo. Se questo campo viene lasciato vuoto è necessario compilare i campi "Numero Telefonico", "Nome Utente" e "Password".



Attenzione! Questa proprietà è disponibile anche nell'interfaccia Basic Script dei Driver di Comunicazione. Le due proprietà non vanno confuse perchè anche se svolgono la stessa funzione vengono utilizzati in due contesti differenti.

Parameter	Description
None	None

Result String

Example:

```

Dim RasObj As RASStationInterface
Dim NetwObj As NetworkClientCmd
Public Sub Click()
    If NetwObj Is Nothing Then Set NetwObj = GetNetworkClient
    If RasObj Is Nothing And Not NetwObj Is Nothing Then Set RasObj =
        NetwObj.GetRASStation("FirstRAS")
    If Not RasObj Is Nothing Then
        Debug.Print RasObj.PhoneBookEntry
    End If
End Sub

```

PhoneNumber, RASStationInterface Property

Syntax PhoneNumber= _String

Description Questa proprietà consente di definire il numero di telefono che la stazione Client deve comporre. Il numero deve corrispondere ovviamente alla linea connessa alla stazione Server. Se la proprietà "Connessione" è stata compilata questo campo può essere lasciato vuoto.



Attenzione! Questa proprietà è disponibile anche nell'interfaccia Basic Script dei Driver di Comunicazione. Le due proprietà non vanno confuse perchè anche se svolgono la stessa funzione vengono utilizzati in due contesti differenti.

Parameter	Description
None	None

Result String

Example:

```
Dim RasObj As RASStationInterface
Dim NetwObj As NetworkClientCmd
Public Sub Click()
    If NetwObj Is Nothing Then Set NetwObj = GetNetworkClient
    If RasObj Is Nothing And Not NetwObj Is Nothing Then Set RasObj =
        NetwObj.GetRASStation("FirstRAS")
    If Not RasObj Is Nothing Then
        Debug.Print RasObj.PhoneNumber
    End If
End Sub
```

PromptForConnection, RASStationInterface Property

Syntax PromptForConnection = _Boolean

Description Se questa proprietà è abilitata, all'inizio di ogni connessione viene presentata una finestra di conferma. In questo caso l'operatore deve confermare l'esecuzione della chiamata ed eventualmente annullarla.



Attenzione! Questa proprietà è disponibile anche nell'interfaccia Basic Script dei Driver di Comunicazione. Le due proprietà non vanno confuse perché anche se svolgono la stessa funzione vengono utilizzati in due contesti differenti.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim RasObj As RASStationInterface
Dim NetwObj As NetworkClientCmd
Public Sub Click()
    If NetwObj Is Nothing Then Set NetwObj = GetNetworkClient
    If RasObj Is Nothing And Not NetwObj Is Nothing Then Set RasObj =
        NetwObj.GetRASStation("FirstRAS")
    If Not RasObj Is Nothing Then
        RasObj.PromptForConnection = True
    End If
End Sub
```

RetryAfterSecs, RASStationInterface Property

Syntax DisconnectAfterSecs = _Long

Description Questa proprietà consente di definire il tempo di attesa prima di ritentare la connessione in caso di fallimento.



Attenzione! Questa proprietà è disponibile anche nell'interfaccia Basic Script dei Driver di Comunicazione. Le due proprietà non vanno confuse perchè anche se svolgono la stessa funzione vengono utilizzati in due contesti differenti.

Parameter	Description
None	None

Result Long

Example:

```
Dim RasObj As RASStationInterface
Dim NetObj As NetworkClientCmd
Public Sub Click()
    If NetObj Is Nothing Then Set NetObj = GetNetworkClient
    If RasObj Is Nothing And Not NetObj Is Nothing Then Set RasObj =
        NetObj.GetRASStation("FirstRAS")
    If Not RasObj Is Nothing Then
        RasObj.DisconnectAfterSecs = 30
    End If
End Sub
```

ShowConnectionDlg, RASStationInterface Property

Syntax ShowConnectionDlg = _Boolean

Description Se questa proprietà è abilitata, durante la connessione viene presentata una finestra per la visualizzazione dello stato della stessa.



Attenzione! Questa proprietà è disponibile anche nell'interfaccia Basic Script dei Driver di Comunicazione. Le due proprietà non vanno confuse perchè anche se svolgono la stessa funzione vengono utilizzati in due contesti differenti.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim RasObj As RASStationInterface
Dim NetObj As NetworkClientCmd
Public Sub Click()
    If NetObj Is Nothing Then Set NetObj = GetNetworkClient
    If RasObj Is Nothing And Not NetObj Is Nothing Then Set RasObj =
        NetObj.GetRASStation("FirstRAS")
    If Not RasObj Is Nothing Then
        RasObj.ShowConnectionDlg = True
    End If
End Sub
```

StartConnectionTime, RASStationInterface Property

Syntax StartConnectionTime = _Date

Description Questa proprietà restituisce l'ora e il giorno in cui la connessione si è stabilita.



Attenzione! Questa proprietà è disponibile anche nell'interfaccia Basic Script dei Driver di Comunicazione. Le due proprietà non vanno confuse perché anche se svolgono la stessa funzione vengono utilizzati in due contesti differenti.

Parameter	Description
None	None

Result Date

Example:

```
Dim RasObj As RASStationInterface
Dim NetwObj As NetworkClientCmd
Public Sub Click()
    If NetwObj Is Nothing Then Set NetwObj = GetNetworkClient
    If RasObj Is Nothing And Not NetwObj Is Nothing Then Set RasObj =
        NetwObj.GetRASStation("FirstRAS")
    If Not RasObj Is Nothing Then
        Debug.Print RasObj.StartConnectionTime
    End If
End Sub
```

TotalConnectionTime, RASStationInterface Property

Syntax TotalConnectionTime = _Date

Description Questa proprietà restituisce il numero di ore totali di connessione.



Attenzione! Questa proprietà è disponibile anche nell'interfaccia Basic Script dei Driver di Comunicazione. Le due proprietà non vanno confuse perché anche se svolgono la stessa funzione vengono utilizzati in due contesti differenti.

Parameter	Description
None	None

Result Date

Example:

```
Dim RasObj As RASStationInterface
Dim NetwObj As NetworkClientCmd
Public Sub Click()
    If NetwObj Is Nothing Then Set NetwObj = GetNetworkClient
    If RasObj Is Nothing And Not NetwObj Is Nothing Then Set RasObj =
        NetwObj.GetRASStation("FirstRAS")
    If Not RasObj Is Nothing Then
        Debug.Print RasObj.TotalConnectionTime
    End If
End Sub
```

End If
End Sub

UserName, RASStationInterface Property

Syntax UserName = _String

Description Questa proprietà consente di inserire il nome utente con il quale la stazione Client deve essere autenticata dalla stazione Server. In questo caso l'utente deve essere un utente riconosciuto dal Sistema Operativo e quindi non necessariamente un utente del progetto Server, ma un utente dichiarato nel Sistema Operativo. Se la proprietà "Connessione" è stata compilata questo campo può essere lasciato vuoto.



Attenzione! Questa proprietà è disponibile anche nell'interfaccia Basic Script dei Driver di Comunicazione. Le due proprietà non vanno confuse perché anche se svolgono la stessa funzione vengono utilizzati in due contesti differenti.

Parameter	Description
None	None

Result String

Example:

```
Dim RasObj As RASStationInterface
Dim NetwObj As NetworkClientCmd
Public Sub Click()
    If NetwObj Is Nothing Then Set NetwObj = GetNetworkClient
    If RasObj Is Nothing And Not NetwObj Is Nothing Then Set RasObj =
        NetwObj.GetRASStation("FirstRAS")
    If Not RasObj Is Nothing Then
        Debug.Print RasObj.PhoneBookEntry
    End If
End Sub
```

1.17.2. RecipeWndCmdTarget

Events

OnActivateRecipe, RecipeWndCmdTarget Event

Description Evento notificato ogni volta che viene premuto il tasto "Attiva" o il tasto di scelta rapida (acceleratore) corrispondente.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

OnCopyRecipe, RecipeWndCmdTarget Event

Description Evento notificato ogni volta che viene premuto il tasto "Copia" o il tasto di scelta rapida (acceleratore) corrispondente.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

OnDeleteRecipe, RecipeWndCmdTarget Event

Description Evento notificato ogni volta che viene premuto il tasto "Cancella" o il tasto di scelta rapida (acceleratore) corrispondente.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

OnExportRecipe, RecipeWndCmdTarget Event

Description Evento notificato ogni volta che viene premuto il tasto "Esporta" o il tasto di scelta rapida (acceleratore) corrispondente.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

OnImportRecipe, RecipeWndCmdTarget Event

Description Evento notificato ogni volta che viene premuto il tasto "Importa" o il tasto di scelta rapida (acceleratore) corrispondente.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

OnPasteRecipe, RecipeWndCmdTarget Event

Description Evento notificato ogni volta che viene premuto il tasto "Paste" o il tasto di scelta rapida (acceleratore) corrispondente.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

OnPrintRecipe, RecipeWndCmdTarget Event

Description Evento notificato ogni volta che viene premuto il tasto "Print" o il tasto di scelta rapida (acceleratore) corrispondente.

Parameter	Description
bRet As Boolean	Abilitazione alla stampa del report.

OnReadRecipe, RecipeWndCmdTarget Event

Description Evento notificato ogni volta che viene premuto il tasto "Leggi" o il tasto di scelta rapida (acceleratore) corrispondente.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

OnRecipeIndexChanged, RecipeWndCmdTarget Event

Description Evento notificato ogni volta che viene modificata la ricetta selezionata nella lista a discesa

Parameter	Description
ChangedRecipeIndex As String	Restituisce il nome della ricetta selezionata.

OnRefreshRecipe, RecipeWndCmdTarget Event

Description Evento notificato ogni volta che viene premuto il tasto "Aggiorna" o il tasto di scelta rapida (acceleratore) corrispondente.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

OnSaveRecipe, RecipeWndCmdTarget Event

Description Evento notificato ogni volta che viene premuto il tasto "Salva" o il tasto di scelta rapida (acceleratore) corrispondente.

Parameter	Description
-----------	-------------

bRet As Boolean

Abilitazione al cambiamento di stato.

Functions

EditLayout, RecipeWndCmdTarget Function

Syntax EditLayout()

Description Questa funzione apre la finestra di configurazione dei campi da visualizzare nella finestra di visualizzazione della ricetta.



La funzione verrà eseguita soltanto se nell'oggetto Finestra è stata abilitata la proprietà "Mostra Finestra di Controllo". In caso contrario la "Finestra Scelta Campo" non verrà aperta e la funzione ritornerà il valore "False"

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objRecipe As RecipeWndCmdTarget
Public Sub Click()
    Debug.Print objRecipe.EditLayout
End Sub
Public Sub SymbolLoading()
    Set objRecipe=
    GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface
End Sub
```

ExportRecipeToCSV, RecipeWndCmdTarget Function

Syntax ExportRecipeToCSV(_IpszFileName)

Description Questa funzione esporta i valori della griglia nel file specificato in formato csv. Il carattere di separazione usato per i vari elementi della griglia è quello impostato nelle proprietà dell'oggetto stesso.
Utilizzando la funzione come nell'esempio sotto riportato si può personalizzare il comportamento predefinito del pulsante "Esporta" dell'oggetto.
Il valore di ritorno consente di sapere se l'operazione è avvenuta con successo o meno.

Parameter	Description
IpszFileName As String	Nome del file in cui devono essere esportati i dati. Il nome deve comprendere il percorso e l'estensione del file. Nel caso in cui venga specificata una stringa vuota apparirà la finestra di esplorazione delle risorse di Windows con la richiesta di indicare il file da utilizzare.

Result Boolean

Example:

```
Public Sub OnExportRecipe(ByRef bRet As Boolean)
    bRet = False

    Dim bResult As Boolean
    Dim sFileName As String

    sFileName = GetDataLoggerRecipePath & Recipe & ".csv"
    bResult = ExportRecipeToCSV(sFileName)

    If bResult Then
        MsgBox "Recipe exported successfully!"
    Else
        MsgBox "Error on exporting the Recipe!"
    End If
End Sub
```

ImportRecipeFromCSV, RecipeWndCmdTarget Function

Syntax ImportRecipeFromCSV (_IpszFileName)

Description Questa funzione importa i valori nella griglia usando il file in formato csv specificato. Il carattere di separazione usato per i vari elementi della griglia è quello impostato nelle proprietà dell'oggetto stesso.
Utilizzando la funzione come nell'esempio sotto riportato si può personalizzare il comportamento predefinito del pulsante "Importa" dell'oggetto.
Il valore di ritorno consente di sapere se l'operazione è avvenuta con successo o meno.

Parameter	Description
IpszFileName As String	Nome del file in cui devono essere importati i dati. Il nome deve comprendere il percorso e l'estensione del file. Nel caso in cui venga specificata una stringa vuota apparirà la finestra di esplorazione delle risorse di Windows con la richiesta di indicare il file da utilizzare.

Result Boolean

Example:

```
Public Sub OnImportRecipe(ByRef bRet As Boolean)
    bRet = False

    Dim bResult As Boolean
    Dim sFileName As String

    sFileName = GetDataLoggerRecipePath & Recipe & ".csv"
    bResult = ImportRecipeFromCSV (sFileName)

    If bResult Then
        MsgBox "Recipe imported successfully!"
    Else
        MsgBox "Error on importing the Recipe!"
    End If
End Sub
```

LoadExtSettings, RecipeWndCmdTarget Function

Syntax LoadExtSettings

Description Questa funzione permette di caricare la configurazione dell'oggetto dal file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".XML".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
Dim objSymbol As RecipeWndCmdTarget  
Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface  
If objSymbol Is Nothing Then Exit Sub  
objSymbol.ExtSettingsFile = "test.sxml"  
objSymbol.LoadExtSettings  
Set objSymbol = Nothing  
End Sub
```

RecalcLayout, RecipeWndCmdTarget Function

Syntax RecalcLayout()

Description Questa funzione esegue un'aggiornamento grafico dell'oggetto. E' necessario eseguire tale funzione dopo che e' stata modificata una proprietà il cui impatto grafico sia particolarmente consistente.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objRecipe As RecipeWndCmdTarget  
Public Sub Click()  
    objRecipe.RecalcLayout  
End Sub  
Public Sub SymbolLoading()  
    Set objRecipe=  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
End Sub
```

Reconnect, RecipeWndCmdTarget Function

Syntax Reconnect()

Description Questa funzione esegue una riconnessione al database in base alla Risorsa Ricetta associata alla finestra di visualizzazione ed un aggiornamento dei dati rileggendoli dalla corrispondente tabella di database. Questa funzione deve essere chiamata per ricaricare i dati se si è cambiata la Risorsa Ricetta associata alla finestra oppure se ci si è collegati ad un un Server di Rete tramite la proprietà "NetworkServerName".

Parameter	Description
None	None

Result Boolean

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.Recipe = "Recipe1"  
    objRecipe.Reconnect  
    Set objRecipe = Nothing  
End Sub
```

Refresh, RecipeWndCmdTarget Function

Syntax Refresh()

Description Questa funzione esegue un aggiornamento dei dati visualizzati nell'oggetto ricaricando i dati presenti nel database. Se si è cambiata la Risorsa Ricetta associata alla finestra occorre invece eseguire il metodo Reconnect.

Parameter	Description
None	None

Result Boolean

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.Refresh  
    Set objRecipe = Nothing  
End Sub
```

SaveExtSettings, RecipeWndCmdTarget Function

Syntax SaveExtSettings

Description Questa funzione permette di salvare la configurazione dell'oggetto nel file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
Dim objSymbol As RecipeWndCmdTarget
Set objSymbol = GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface
If objSymbol Is Nothing Then Exit Sub
objSymbol.ExtSettingsFile = "test.sxml"
objSymbol.SaveExtSettings
Set objSymbol = Nothing
End Sub
```

Properties

ActivateBtnText, RecipeWndCmdTarget Property

Syntax ActivateBtnText = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato nel pulsante "Attiva" (se presente) della finestra di visualizzazione della ricetta. Se questo testo non è specificato, verrà utilizzato il testo di default. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()
Dim objRecipe As RecipeWndCmdTarget
Set objRecipe =
GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface
objRecipe.ActivateBtnText = "Activate recipe"
objRecipe.RecalcLayout
Set objRecipe = Nothing
End Sub
```

ActivateMessage, RecipeWndCmdTarget Property

Syntax ActivateMessage = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato come stringa per la Message Box di richiesta conferma che verrà aperta quando si esegue un comando di "Attiva" ricetta. Impostando la proprietà con una stringa vuota la Message Box non verrà visualizzata e il comando verrà eseguito senza chiedere alcuna conferma. E' possibile inserire anche un ID di stringa.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()
    Dim objRecipe As RecipeWndCmdTarget
    Set objRecipe =
    GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface
    objRecipe.ActivateMessage = "Do you want to activate the selected Recipe?"
    Set objRecipe = Nothing
End Sub
```

AutoLayout, RecipeWndCmdTarget Property

Syntax AutoLayout = _Boolean

Description Abilitando questa proprietà, il layout della lista verrà impostato in modo automatico. Questo significa che le colonne della tabella verranno automaticamente ridimensionate in modo che risultino visibili tutte quante nell'area della Finestra Data Logger/Ricette. Disabilitando invece questa proprietà, all'apertura della finestra le colonne avranno la dimensione definita in programmazione, con la possibilità che le ultime sulla destra non siano visibili se non attivando la barra di scorrimento orizzontale.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objRecipe As RecipeWndCmdTarget
Public Sub Click()
    Debug.Print objRecipe.AutoLayout
End Sub
Public Sub SymbolLoading()
    Set
    GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface
End Sub
```

ButtonPos, RecipeWndCmdTarget Property

Syntax ButtonPos = _Integer

Description Questa impostazione restituisce la posizione in cui dovranno apparire i pulsanti nella finestra di visualizzazione dei dati.

Le possibilità sono:
0 = sinistra
1 = sopra
2 = destra
3 = sotto

Parameter	Description
None	None

Result Integer

Example:

```
Dim objRecipe As ButtonPos, RecipeWndCmdTarget Property
Public Sub Click()
    If Not objRecipe Is Nothing Then
        MsgBox "objRecipe 's ButtonPos is " & objRecipe
        .ButtonPos,vbInformation,GetProjectTitle
        objRecipe.ButtonPos = 2
        objRecipe.RecalcLayout
    Else
        MsgBox "objRecipe is nothing",vbInformation,GetProjectTitle
    End If
End Sub
Public Sub SymbolLoading()
    Set objRecipe =
    GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface
End Sub
```

ButtonSize, RecipeWndCmdTarget Property

Syntax ButtonSize = _Integer

Description Questa impostazione restituisce la dimensione dei pulsanti che verranno poi visualizzati nella finestra di visualizzazione dei dati.

Le possibilità sono:
0 = piccolo
1 = medio
2 = grande

Parameter	Description
None	None

Result Integer

Example:

```
Dim objRecipe As ButtonSize, RecipeWndCmdTarget Property
Public Sub Click()
    If Not objRecipe Is Nothing Then
        MsgBox "objRecipe 's ButtonSize is " &
        objRecipe.ButtonSize,vbInformation,GetProjectTitle
        objRecipe.ButtonSize = 2
        objRecipe.RecalcLayout
    Else
        MsgBox "objRecipe is nothing",vbInformation,GetProjectTitle
    End If
End Sub
```

```

Public Sub SymbolLoading()
    Set objRecipe = GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface
End Sub

```

Clickable, RecipeWndCmdTarget Property

Syntax Clickable = _Boolean

Description Questa proprietà permette di definire se l'operatore potrà interagire con la finestra di Visualizzazione. Impostando a False questa proprietà il controllo non potrà infatti essere gestito nè con il mouse nè con la tastiera. In questo caso quindi non sarà possibile, ad esempio, ordinare le colonne, vedere l'eventuale help, eseguire eventuali comandi presenti nella finestra.

Parameter	Description
None	None

Result Boolean

Example:

```

Dim objRecipe As Clickable, RecipeWndCmdTarget Property
Public Sub Click()
    Debug.Print objRecipe.Project
End Sub
Public Sub SymbolLoading()
    Set objRecipe = GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface
End Sub

```

CopyBtnText, RecipeWndCmdTarget Property

Syntax CopyBtnText = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato nel pulsante "Copia" (se presente) della finestra di visualizzazione della ricetta. Se questo testo non è specificato, verrà utilizzato il testo di default. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result String

Example:

```

Public sub Click()
    Dim objRecipe As RecipeWndCmdTarget
    Set objRecipe = GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface
    objRecipe.CopyBtnText = "Copy recipe"
    objRecipe.RecalcLayout
    Set objRecipe = Nothing
End Sub

```


CurrentRecipeVariable, RecipeWndCmdTarget Property

Syntax CurrentRecipeVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile del progetto sulla quale verrà inserito il nome della ricetta attivata. Corrisponde alla proprietà "Tag Ricetta Attiva" della finestra Gestione Ricetta.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Dim sText as String  
    GetVariableNameFromList(sText)  
    Set objRecipe =  
    GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.CurrentRecipeVariable = sText  
    Set objRecipe = Nothing  
End Sub
```

DeleteBtnText, RecipeWndCmdTarget Property

Syntax DeleteBtnText = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato nel pulsante "Elimina" (se presente) della finestra di visualizzazione della ricetta. Se questo testo non è specificato, verrà utilizzato il testo di default. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
    GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.DeleteBtnText = "Delete recipe"  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

DeleteMessage, RecipeWndCmdTarget Property

Syntax DeleteMessage = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato come stringa per la Message Box di richiesta conferma che verrà aperta quando si esegue un comando di "Cancella" ricetta. Impostando la proprietà con una stringa vuota la Message Box non verrà visualizzata e il comando verrà eseguito senza chiedere alcuna conferma. E' possibile inserire anche un ID di stringa.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
        objRecipe.DeleteMessage = "Do you want to delete the selected Recipe?"  
    Set objRecipe = Nothing  
End Sub
```

ErrorString, RecipeWndCmdTarget Property

Syntax ErrorString = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato come stringa di errore durante l'impostazione di un valore nella colonna "Valore" della griglia, se editabile. Impostando la proprietà con una stringa vuota verrà generato un semplice "Beep". E' possibile per una variabile numerica visualizzare i limiti Min e Max nella stringa di errore utilizzando la notazione "%d" al posto del valore (vedi esempio).

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
        objRecipe.ErrorString = "Value out of range: min = %d, max = %d"  
    Set objRecipe = Nothing  
End Sub
```

ExportBtnText, RecipeWndCmdTarget Property

Syntax ExportBtnText = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato nel pulsante "Esporta" (se presente) della finestra di visualizzazione della ricetta. Se questo testo non è specificato, verrà utilizzato il testo di default. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.ExportBtnText = "Export recipe"  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

ExtSettingsFile, RecipeWndCmdTarget Property

Syntax ExtSettingsFile = _String

Description Questa proprietà imposta o restituisce il file di configurazione esterna per l'oggetto referenziato. Il file può essere specificato anche in fase di design nella proprietà "File di Configurazione" dell'oggetto. L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    Dim objSymbol As ExtSettingsFile, RecipeWndCmdTarget Property  
    Set objSymbol =  
        GetSynopticObject.GetSubObject("TestObject").GetObjectInterface  
    If objSymbol Is Nothing Then Exit Sub  
    objSymbol.ExtSettingsFile = "test.sxml"  
    objSymbol.SaveExtSettings  
    Set objSymbol = Nothing  
End Sub
```

GraphicButtons, RecipeWndCmdTarget Property

Syntax GraphicButtons = _Boolean

Description Abilitando questa proprietà i pulsanti della Finestra Gestione Ricetta verranno disegnati usando un'icona invece del testo. Il testo opportuno verrà invece visualizzato come tooltip posizionando il mouse sopra ad un pulsante.



Il tooltip non viene gestito nella versione Windows CE.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()  
    GraphicButtons = True  
    RecalcLayout  
End Sub
```

ImpExpSeparator, RecipeWndCmdTarget Property

Syntax ImpExpSeparator = _Integer

Description Questa proprietà imposta o restituisce il codice ASCII del carattere usato come separatore tra le colonne nel file di esportazione/importazione della ricetta.

Parameter	Description
None	None

Result Integer

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.ImpExpSeparator = Asc(",")  
    Set objRecipe = Nothing  
End Sub
```

ImportBtnText, RecipeWndCmdTarget Property

Syntax ImportBtnText = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato nel pulsante "Importa" (se presente) della finestra di visualizzazione della ricetta. Se questo testo non è specificato, verrà utilizzato il testo di default. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.ImportBtnText = "Import recipe"  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

NetworkBackupServerName, RecipeWndCmdTarget Property

Syntax NetworkBackupServerName = _String

Description Questa proprietà imposta o restituisce il nome dell'eventuale Server di Networking di Backup usato per prelevare i dati da visualizzare nella Finestra Gestione Ricette quando il Server Primario, ovvero quello impostato nella proprietà "NetowrkServerName", è in timeout.



Per visualizzare i dati di un Server è necessario che la Ricetta sia presente anche sul progetto Client, al fine di poter recuperare la struttura del Database. Tuttavia sul progetto Client la Ricetta può essere creato soltanto come struttura, senza quindi dover associare alcuna variabile alle colonne.

Parameter	Description
None	None

Result String

Example:

```
Dim objRecipeWnd As RecipeWndCmdTarget  
Public Sub Click()  
    Debug.Print objRecipeWnd.NetworkBackupServerName  
End Sub  
Public Sub SymbolLoading()
```

```

Set objRecipeWnd =
GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface
End Sub

```

NetworkServerName, RecipeWndCmdTarget Property

Syntax NetworkServerName = _String

Description Questa proprietà restituisce il nome dell'eventuale Server di Networking dal quale dovranno essere prelevati i dati da visualizzare nella Finestra Gestione Ricette.



Per visualizzare i dati di un Server è necessario che la Ricetta sia presente anche sul progetto Client, al fine di poter recuperare la struttura del Database. Tuttavia sul progetto Client la Ricetta può essere creato soltanto come struttura, senza quindi dover associare alcuna variabile alle colonne.

Parameter	Description
None	None

Result String

Example:

```

Dim objRecipe As RecipeWndCmdTarget
Public Sub Click()
    Debug.Print objRecipe.NetworkServerName
End Sub
Public Sub SymbolLoading()
    Set                                     objRecipe
    GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface =
End Sub

```

PasteBtnText, RecipeWndCmdTarget Property

Syntax PasteBtnText = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato nel pulsante "Incolla" (se presente) della finestra di visualizzazione della ricetta. Se questo testo non è specificato, verrà utilizzato il testo di default. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result String

Example:

```

Public sub Click()
    Dim objRecipe As RecipeWndCmdTarget
    Set objRecipe =
    GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface
    objRecipe.PasteBtnText = "Paste recipe"

```

```

        objRecipe.RecalcLayout
        Set objRecipe = Nothing
    End Sub

```

PrintBtnText, RecipeWndCmdTarget Property

Syntax PrintBtnText = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato nel pulsante "Stampa" (se presente) della finestra di visualizzazione della ricetta. Se questo testo non è specificato, verrà utilizzato il testo di default. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result String

Example:

```

Public sub Click()
    Dim objRecipe As RecipeWndCmdTarget
    Set objRecipe =
    GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface
    objRecipe.PrintBtnText = "Print recipe"
    objRecipe.RecalcLayout
    Set objRecipe = Nothing
End Sub

```

Project, RecipeWndCmdTarget Property

Syntax Project = _String

Description Questa proprietà imposta o restituisce il nome del progetto figlio dal quale si vogliono recuperare i dati da visualizzare. Se il campo viene lasciato vuoto verrà utilizzato il progetto corrente.



In questa proprietà va inserito esclusivamente il nome di un eventuale progetto figlio del progetto corrente.

Parameter	Description
None	None

Result String

Example:

```

Dim objRecipe As Project, RecipeWndCmdTarget Property
Public Sub Click()
    Debug.Print objRecipe.Project
End Sub
Public Sub SymbolLoading()
    Set objRecipe =
    GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface
End Sub

```

PromptPad, RecipeWndCmdTarget Property

Syntax PromptPad = _Boolean

Description Questa proprietà abilita o disabilita la possibilità di visualizzare il Pad Numerico o Alfanumerico quando l'utente entra in editazione (con un click del mouse o da tastiera) su una cella della colonna "Valore" della griglia, se editabile. Il Pad Numerico compare se la variabile è di tipo numerico, altrimenti comparirà il Pad Alfanumerico.

Parameter	Description
None	None

Result Boolean

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
        objRecipe.PromptPad = Not objRecipe.PromptPad  
    Set objRecipe = Nothing  
End Sub
```

ReadBtnText, RecipeWndCmdTarget Property

Syntax ReadBtnText = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato nel pulsante "Leggi" (se presente) della finestra di visualizzazione della ricetta. Se questo testo non è specificato, verrà utilizzato il testo di default. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.ReadBtnText = "Read recipe"  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```


Recipe, RecipeWndCmdTarget Property

Syntax Recipe = _String

Description Questa proprietà imposta o restituisce il nome della Risorsa Ricetta associata alla finestra di visualizzazione. Se si modifica la Risorsa Ricetta associata alla finestra di visualizzazione e si vogliono visualizzare i dati relativi alla nuova Risorsa Ricetta, occorre eseguire il metodo Reconnect.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    Debug.Print objRecipe.Recipe  
    objRecipe.Recipe = "Recipe1"  
    objRecipe.Reconnect  
    Set objRecipe = Nothing  
End Sub
```

RefreshBtnText, RecipeWndCmdTarget Property

Syntax RefreshBtnText = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato nel pulsante "Aggiorna" (se presente) della finestra di visualizzazione della ricetta. Se questo testo non è specificato, verrà utilizzato il testo di default. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.RefreshBtnText = "Refresh recipe"  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

SaveBtnText, RecipeWndCmdTarget Property

Syntax SaveBtnText = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato nel pulsante "Salva" (se presente) della finestra di visualizzazione della ricetta. Se questo testo non è specificato, verrà utilizzato il testo di default. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.SaveBtnText = "Save recipe"  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

SaveMessage, RecipeWndCmdTarget Property

Syntax SaveMessage = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato come stringa per la Message Box di richiesta conferma che verrà aperta quando si esegue un comando di "Salva" ricetta. Impostando la proprietà con una stringa vuota la Message Box non verrà visualizzata e il comando verrà eseguito senza chiedere alcuna conferma. E' possibile inserire anche un ID di stringa.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.SaveMessage = "Do you want to save the selected Recipe?"  
    Set objRecipe = Nothing  
End Sub
```

ShowActivateBtn, RecipeWndCmdTarget Property

Syntax ShowActivateBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante "Attiva" nella finestra di visualizzazione ricetta. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result Boolean

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.ShowActivateBtn = Not objRecipe.ShowActivateBtn  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

ShowCopyBtn, RecipeWndCmdTarget Property

Syntax ShowCopyBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante "Copia" nella finestra di visualizzazione ricetta. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result Boolean

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.ShowCopyBtn = Not objRecipe.ShowCopyBtn  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

ShowDeleteBtn, RecipeWndCmdTarget Property

Syntax ShowDeleteBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante "Cancella" nella finestra di visualizzazione ricetta. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result Boolean

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
    GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.ShowDeleteBtn = Not objRecipe.ShowActivateBtn  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

ShowExportBtn, RecipeWndCmdTarget Property

Syntax ShowExportBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante "Esporta" nella finestra di visualizzazione ricetta. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result Boolean

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
    GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.ShowExportBtn = Not objRecipe.ShowExportBtn  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

ShowImportBtn, RecipeWndCmdTarget Property

Syntax ShowImportBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante "Importa" nella finestra di visualizzazione ricetta. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result Boolean

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.ShowImportBtn = Not objRecipe.ShowImportBtn  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

ShowPasteBtn, RecipeWndCmdTarget Property

Syntax ShowPasteBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante "Incolla" nella finestra di visualizzazione ricetta. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result Boolean

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.ShowPasteBtn = Not objRecipe.ShowPasteBtn  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

ShowPrintBtn, RecipeWndCmdTarget Property

Syntax ShowPrintBtn= _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante "Stampa" nella finestra di visualizzazione ricetta. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result Boolean

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.ShowPrintBtn = Not objRecipe.ShowPrintBtn  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

ShowReadBtn, RecipeWndCmdTarget Property

Syntax ShowReadBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante "Leggi" nella finestra di visualizzazione ricetta. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result Boolean

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.ShowReadBtn = Not objRecipe.ShowReadBtn  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

ShowRefreshBtn, RecipeWndCmdTarget Property

Syntax ShowRefreshBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante "Aggiorna" nella finestra di visualizzazione ricetta. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result Boolean

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.ShowRefreshBtn = Not objRecipe.ShowRefreshBtn  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

ShowSaveBtn, RecipeWndCmdTarget Property

Syntax ShowSaveBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante "Salva" nella finestra di visualizzazione ricetta. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result Boolean

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.ShowSaveBtn = Not objRecipe.ShowSaveBtn  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

SubItemDescription, RecipeWndCmdTarget Property

Syntax SubItemDescription= _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato come titolo della colonna "Descrizione" (se presente) della finestra di visualizzazione della ricetta. Se questo testo non è specificato, verrà utilizzato il testo di default. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.SubItemDescription = "Description"  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

SubItemDescriptionPos, RecipeWndCmdTarget Property

Syntax SubItemDescriptionPos= _String

Description Questa proprietà imposta o restituisce la posizione della colonna "Descrizione" all'interno della Finestra Gestione Ricetta. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.SubItemDescriptionPos = 1  
    Set objRecipe = Nothing  
End Sub
```


SubItemDescriptionWidth, RecipeWndCmdTarget Property

Syntax SubItemDescriptionWidth = _Integer

Description Questa proprietà imposta o restituisce la dimensione in pixel della colonna all'interno della finestra di visualizzazione della ricetta. Se si modifica la dimensione della colonna nella finestra è necessario richiamare il metodo RecalcLayout affinché diventi effettiva la modifica.

Parameter	Description
None	None

Result Integer

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.SubItemDescriptionWidth = 100  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

SubItemMax, RecipeWndCmdTarget Property

Syntax SubItemMax = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato come titolo della colonna "Max" (se presente) della finestra di visualizzazione della ricetta. Se questo testo non è specificato, verrà utilizzato il testo di default. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.SubItemMax = "Max"  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

SubItemMaxPos, RecipeWndCmdTarget Property

Syntax SubItemMaxPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Max" all'interno della Finestra Gestione Ricetta. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
        objRecipe.SubItemMaxPos = 5  
    Set objRecipe = Nothing  
End Sub
```

SubItemMaxWidth, RecipeWndCmdTarget Property

Syntax SubItemMaxWidth = _Integer

Description Questa proprietà imposta o restituisce la dimensione in pixel della colonna all'interno della finestra di visualizzazione della ricetta. Se si modifica la dimensione della colonna nella finestra è necessario richiamare il metodo RecalcLayout affinché diventi effettiva la modifica.

Parameter	Description
None	None

Result Integer

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
        objRecipe.SubItemMaxWidth = 100  
        objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

SubItemMin, RecipeWndCmdTarget Property

Syntax SubItemMin = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato come titolo della colonna "Min" (se presente) della finestra di visualizzazione della ricetta. Se questo testo non è specificato, verrà utilizzato il testo di default. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.SubItemMin = "Min"  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

SubItemMinPos, RecipeWndCmdTarget Property

Syntax SubItemMinPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Min" all'interno della Finestra Gestione Ricetta. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.SubItemMinPos = 4  
    Set objRecipe = Nothing  
End Sub
```

SubItemMinWidth, RecipeWndCmdTarget Property

Syntax SubItemMinWidth = _Integer

Description Questa proprietà imposta o restituisce la dimensione in pixel della colonna all'interno della finestra di visualizzazione della ricetta. Se si modifica la dimensione della colonna nella finestra è necessario richiamare il metodo RecalcLayout affinché diventi effettiva la modifica.

Parameter	Description
None	None

Result Integer

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.SubItemMinWidth = 100  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

SubItemUnits, RecipeWndCmdTarget Property

Syntax SubItemUnits = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato come titolo della colonna "Unità" (se presente) della finestra di visualizzazione della ricetta. Se questo testo non è specificato, verrà utilizzato il testo di default. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.SubItemUnits = "Units"  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

SubItemUnitsPos, RecipeWndCmdTarget Property

Syntax SubItemUnitsPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Unità" all'interno della Finestra Gestione Ricetta. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
        objRecipe.SubItemUnitsPos = 3  
    Set objRecipe = Nothing  
End Sub
```

SubItemUnitsWidth, RecipeWndCmdTarget Property

Syntax SubItemUnitsWidth = _Integer

Description Questa proprietà imposta o restituisce la dimensione in pixel della colonna all'interno della finestra di visualizzazione della ricetta. Se si modifica la dimensione della colonna nella finestra è necessario richiamare il metodo RecalcLayout affinché diventi effettiva la modifica.

Parameter	Description
None	None

Result Integer

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
        objRecipe.SubItemUnitsWidth = 100  
        objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

SubItemValue, RecipeWndCmdTarget Property

Syntax SubItemValue = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato come titolo della colonna "Valore" (se presente) della finestra di visualizzazione della ricetta. Se questo testo non è specificato, verrà utilizzato il testo di default. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.SubItemValue = "Value"  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

SubItemValuePos, RecipeWndCmdTarget Property

Syntax SubItemValuePos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Valore" all'interno della Finestra Gestione Ricetta. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.SubItemValuePos = 2  
    Set objRecipe = Nothing  
End Sub
```

SubItemValueWidth, RecipeWndCmdTarget Property

Syntax SubItemValueWidth = _Integer

Description Questa proprietà imposta o restituisce la dimensione in pixel della colonna all'interno della finestra di visualizzazione della ricetta. Se si modifica la dimensione della colonna nella finestra è necessario richiamare il metodo RecalcLayout affinché diventi effettiva la modifica.

Parameter	Description
None	None

Result Integer

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.SubItemValueWidth = 100  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

SubItemVariable, RecipeWndCmdTarget Property

Syntax SubItemVariable = _String

Description Questa proprietà imposta o restituisce il testo che verrà visualizzato come titolo della colonna "Variabile" (se presente) della finestra di visualizzazione della ricetta. Se questo testo non è specificato, verrà utilizzato il testo di default. Per rendere effettiva la modifica occorre chiamare il metodo RecalcLayout.

Parameter	Description
None	None

Result String

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
    objRecipe.SubItemVariable = "Variable"  
    objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```

SubItemVariablePos, RecipeWndCmdTarget Property

Syntax SubItemVariablePos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Variabile" all'interno della Finestra Gestione Ricetta. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
        objRecipe.SubItemVariablePos = 0  
    Set objRecipe = Nothing  
End Sub
```

SubItemVariableWidth, RecipeWndCmdTarget Property

Syntax SubItemVariableWidth = _Integer

Description Questa proprietà imposta o restituisce la dimensione in pixel della colonna all'interno della finestra di visualizzazione della ricetta. Se si modifica la dimensione della colonna nella finestra è necessario richiamare il metodo RecalcLayout affinché diventi effettiva la modifica.

Parameter	Description
None	None

Result Integer

Example:

```
Public sub Click()  
    Dim objRecipe As RecipeWndCmdTarget  
    Set objRecipe =  
        GetSynopticObject.GetSubObject("RecipeWindow").GetObjectInterface  
        objRecipe.SubItemVariableWidth = 100  
        objRecipe.RecalcLayout  
    Set objRecipe = Nothing  
End Sub
```


1.17.3. ScalingCmdTarget

Functions

GetXMLSettings, ScalingCmdTarget Function

Syntax GetXMLSettings()

Description Questa funzione restituisce una stringa con il contenuto XML del file di progetto relativo all'oggetto normalizzatore referenziato.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ScalObj As ScalingCmdTarget
    Set ScalObj = GetScaling("Scal1")
    If Not ScalObj Is Nothing Then
        MsgBox ScalObj.GetXMLSettings ,vbOkOnly,""
    End If
    Set ScalObj = Nothing
End Sub
```

Reinit, ScalingCmdTarget Function

Syntax Reinit()

Description Questo metodo reinizializza un oggetto normalizzatore e va utilizzato in runtime per rendere attive le eventuali modifiche apportate ai valori delle sue proprietà.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim ScalObj As ScalingCmdTarget
    Set ScalObj = GetScaling("Scal1")
    ScalObj.RawMaxValue = 10
    ScalObj.Reinit
    Set ScalObj = Nothing
End Sub
```

Properties

DeadBandValue, ScalingCmdTarget Property

Syntax DeadBandValue = _Double

Description Questa proprietà imposta o restituisce il valore della "banda morta" nel fattore di conversione. La banda morta stabilisce il valore a cui la variabile normalizzata viene posta nel caso in cui il valore della variabile non normalizzata esca dalla tolleranza di conversione impostata. Il valore di default impostato da Movicon è "-1".

Parameter	Description
None	None

Result Double

Example:

```
Option Explicit
Public Sub Click()
    Dim ScalObj As ScalingCmdTarget
    Set ScalObj = GetScaling("Scal1")
    If Not ScalObj Is Nothing Then
        Debug.Print ScalObj.DeadBandValue
    End If
    Set ScalObj = Nothing
End Sub
```

Enabled, ScalingCmdTarget Property

Syntax Enabled = _Boolean

Description Questa proprietà abilita o disabilita l'oggetto normalizzatore referenziato. Lasciando il valore della proprietà a False le operazioni di conversione non saranno eseguite.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim ScalObj As ScalingCmdTarget
    Set ScalObj = GetScaling("Scal1")
    If Not ScalObj Is Nothing Then
        Debug.Print ScalObj.Enabled
    End If
    Set ScalObj = Nothing
End Sub
```

Name, ScalingCmdTarget Property

Syntax Name = _String

Description Questa funzione restituisce una stringa con il nome dell'oggetto normalizzatore referenziato.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ScalObj As ScalingCmdTarget
    Set ScalObj = GetScaling("Scal1")
    If Not ScalObj Is Nothing Then
        MsgBox "Scaling Name Is " & ScalObj.Name,vbOkOnly,""
    End If
    Set ScalObj = Nothing
End Sub
```

RawMaxValue, ScalingCmdTarget Property

Syntax RawMaxValue = _Double

Description Questa proprietà imposta o restituisce il valore massimo della variabile non normalizzata, ovvero il valore in ingresso. Sulla base del valore minimo e massimo di ingresso verrà calcolato il valore minimo e massimo normalizzato in uscita secondo una funzione lineare.

Parameter	Description
None	None

Result Double

Example:

```
Option Explicit
Public Sub Click()
    Dim ScalObj As ScalingCmdTarget
    Set ScalObj = GetScaling("Scal1")
    If Not ScalObj Is Nothing Then
        Debug.Print ScalObj.RawMaxValue
    End If
    Set ScalObj = Nothing
End Sub
```

RawMinValue, ScalingCmdTarget Property

Syntax RawMinValue = _Double

Description Questa proprietà imposta o restituisce il valore minimo della variabile non normalizzata, ovvero il valore in ingresso. Sulla base del valore minimo e massimo di ingresso verrà calcolato il valore minimo e massimo normalizzato in uscita secondo una funzione lineare.

Parameter	Description
None	None

Result Double

Example:

```
Option Explicit
Public Sub Click()
    Dim ScalObj As ScalingCmdTarget
    Set ScalObj = GetScaling("Scal1")
    If Not ScalObj Is Nothing Then
        Debug.Print ScalObj.RawMinValue
    End If
    Set ScalObj = Nothing
End Sub
```

RawVariableName, ScalingCmdTarget Property

Syntax RawVariableName = _String

Description Questa funzione restituisce una stringa con il nome della variabile che contiene il valore da normalizzare.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim ScalObj As ScalingCmdTarget
    Set ScalObj = GetScaling("Scal1")
    If Not ScalObj Is Nothing Then
        MsgBox ScalObj.RawVariableName, vbOkOnly, ""
    End If
    Set ScalObj = Nothing
End Sub
```

ScaledMaxValue, ScalingCmdTarget Property

Syntax ScaledMaxValue = _Double

Description Questa proprietà imposta o restituisce il valore massimo della variabile normalizzata, ovvero il valore in uscita corrispondente alla grandezza fisica reale.

Parameter	Description
None	None

Result Double

Example:

```
Option Explicit
Public Sub Click()
    Dim ScalObj As ScalingCmdTarget
    Set ScalObj = GetScaling("Scal1")
    If Not ScalObj Is Nothing Then
        Debug.Print ScalObj.ScaledMaxValue
    End If
    Set ScalObj = Nothing
End Sub
```

ScaledMinValue, ScalingCmdTarget Property

Syntax ScaledMinValue = _Double

Description Questa proprietà imposta o restituisce il valore minimo della variabile normalizzata, ovvero il valore in uscita corrispondente alla grandezza fisica reale.

Parameter	Description
None	None

Result Double

Example:

```
Option Explicit
Public Sub Click()
    Dim ScalObj As ScalingCmdTarget
    Set ScalObj = GetScaling("Scal1")
    If Not ScalObj Is Nothing Then
        Debug.Print ScalObj.ScaledMinValue
    End If
    Set ScalObj = Nothing
End Sub
```

ScaleVariableName, ScalingCmdTarget Property

Syntax ScaleVariableName = _String

Description Questa funzione restituisce una stringa con il nome della variabile che conterrà il valore normalizzato, ovvero il risultato del calcolo di conversione sulla base dei fattori di conversione impostati nella proprietà descritte a seguito.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
```

```

Dim ScalObj As ScalingCmdTarget
Set ScalObj = GetScaling("Scal1")
If Not ScalObj Is Nothing Then
    MsgBox ScalObj.ScaleVariableName,vbOkOnly,""
End If
Set ScalObj = Nothing
End Sub

```

1.17.4. SchedulerCmdTarget

Functions

AddHoliday, SchedulerCmdTarget Function

Syntax AddHoliday(_dt)

Description Questa funzione consente di aggiungere una festività nell'oggetto schedulatore. La data viene aggiunta soltanto se non è già presente, altrimenti la funzione ritorna "false". Occorre passare alla funzione un parametro di tipo "data" di cui però viene considerato solo il giorno e il mese.

Parameter	Description
_dt as date	Data che dovrà essere gestita come festività

Result Boolean

Example:

```

Option Explicit
Public Sub Click()
    Dim SchedObj As SchedulerCmdTarget
    Set SchedObj = GetScheduler("Sched1")
    If Not SchedObj Is Nothing Then
        SchedObj.AddHoliday(now)
    End If
    Set SchedObj = Nothing
End Sub

```

GetHolidaysString, SchedulerCmdTarget Function

Syntax GetHolidaysString(_lpszSep)

Description Questa funzione restituisce una stringa divisa dal carattere di separazione prescelto (parametro della funzione) con l'elenco di tutte le date di festività impostate nell'oggetto schedulatore. Le date vengono espresse soltanto con il giorno e mese.

Parameter	Description
lpszSep as string	Carattere separatore tra le date restituite nella stringa dalla funzione

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim SchedObj As SchedulerCmdTarget
    Set SchedObj = GetScheduler("Sched1")
    If Not SchedObj Is Nothing Then
        MsgBox "Holidays Date = " &
            CStr(SchedObj.GetHolidaysString(",")),vbOkOnly,GetProjectTitle
    End If
    Set SchedObj = Nothing
End Sub

```

GetXMLSettings, SchedulerCmdTarget Function

Syntax GetXMLSettings()

Description Questa funzione restituisce una stringa con il contenuto XML del file di progetto relativo all'oggetto schedulatore referenziato.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim SchedObj As SchedulerCmdTarget
    Set SchedObj = GetScheduler("Sched1")
    If Not SchedObj Is Nothing Then
        MsgBox SchedObj.GetXMLSettings ,vbOkOnly,""
    End If
    Set SchedObj = Nothing
End Sub

```

IsHoliday, SchedulerCmdTarget Function

Syntax IsHoliday(_dt)

Description Questa funzione consente di verificare se una data è impostata come festività nell'oggetto schedulatore. Occorre passare alla funzione un parametro di tipo "data" di cui però viene considerato solo il giorno e il mese.

Parameter	Description
_dt as date	Data che dovrà essere eliminata dalla lista delle festività

Result Boolean

Example:

```

Option Explicit
Public Sub Click()
    Dim SchedObj As SchedulerCmdTarget
    Set SchedObj = GetScheduler("Sched1")
    If Not SchedObj Is Nothing Then

```

```

        MsgBox "Is Holiday = " &
        CStr(SchedObj.IsHoliday(Now)),vbOkOnly,GetProjectTitle
    End If
    Set SchedObj = Nothing
End Sub

```

RemoveHoliday, SchedulerCmdTarget Function

Syntax RemoveHoliday(_dt)

Description Questa funzione consente di eliminare una festività nell'oggetto schedulatore. La data viene eliminata soltanto se è presente, altrimenti la funzione ritorna "false". Occorre passare alla funzione un parametro di tipo "data" di cui però viene considerato solo il giorno e il mese.

Parameter	Description
_dt as date	Data che dovrà essere eliminata dalla lista delle festività

Result Boolean

Example:

```

Option Explicit
Public Sub Click()
    Dim SchedObj As SchedulerCmdTarget
    Set SchedObj = GetScheduler("Sched1")
    If Not SchedObj Is Nothing Then
        SchedObj.RemoveHoliday(now)
    End If
    Set SchedObj = Nothing
End Sub

```

Reset, SchedulerCmdTarget Function

Syntax Reset()

Description Resetta il comportamento dello schedulatore. Permette l'esecuzione di un comando inserito in un periodo successivo a quello appena eseguito, senza dover attendere il termine del periodo di base definito nel tipo di schedulazione.

Parameter	Description
None	None

Result Boolean

Example:

```

Dim myObj As SchedulerCmdTarget
Public Sub Click()
    Set myObj = GetScheduler("Schedulatore")
    myObj.Reset
End Sub

```


SaveRetentive, SchedulerCmdTarget Function

Syntax SaveRetentive()

Description Questa funzione consente di salvare le pianificazioni orarie (normale e festiva) e i giorni di festività sul file esterno per la persistenza di questi dati anche dopo un riavvio dell'applicazione. Il file salvato è lo stesso che si può salvare con l'oggetto "Finestra Scheduler" per gli schedulatori di tipo "piano ore". La funzione viene gestita solo per gli schedulatori di tipo "piano ore" e "data fissa", cioè per quelli che gestiscono le festività.

Il file viene salvato nella sotto cartella "DATA" del progetto con la seguente forma:

<NomeProgetto>_<NomeScheduler>.shp

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim SchedObj As SchedulerCmdTarget
    Set SchedObj = GetScheduler("Sched1")
    If Not SchedObj Is Nothing Then
        SchedObj.SaveRetentive
    End If
    Set SchedObj = Nothing
End Sub
```

Properties

CommandList, SchedulerCmdTarget Property

Syntax CommandList = _String

Description Questa proprietà restituisce la stringa XML di progetto contenente la definizione dei comandi associati all'oggetto scheduler referenziato quando la condizione associata è attivata.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim SchedObj As SchedulerCmdTarget
    Set SchedObj = GetScheduler("Sched1")
    If Not SchedObj Is Nothing Then
        MsgBox SchedObj.CommandList,vbOkOnly,""
    End If
    Set SchedObj = Nothing
End Sub
```

CommandListOff, SchedulerCmdTarget Property

Syntax CommandListOff = _String

Description Questa proprietà restituisce la stringa XML di progetto contenente la definizione dei comandi associati all'oggetto schedulatore referenziato quando la condizione associata è disattivata.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim SchedObj As SchedulerCmdTarget
    Set SchedObj = GetScheduler("Sched1")
    If Not SchedObj Is Nothing Then
        MsgBox SchedObj.CommandListOff,vbOkOnly,""
    End If
    Set SchedObj = Nothing
End Sub
```

Enabled, SchedulerCmdTarget Property

Syntax Enabled = _Boolean

Description Questa proprietà abilita o disabilita l'oggetto schedulatore referenziato. Lasciando il valore della proprietà a False le operazioni schedulate non saranno eseguite.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim SchedObj As SchedulerCmdTarget
    Set SchedObj = GetScheduler("Sched1")
    If Not SchedObj Is Nothing Then
        MsgBox SchedObj.Enabled ,vbOkOnly,""
    End If
    Set SchedObj = Nothing
End Sub
```

EnableVariable, SchedulerCmdTarget Property

Syntax EnableVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile di abilitazione per l'oggetto schedatore referenziato. Se la proprietà contiene una stringa vuota Movicon considera per l'abilitazione dell'oggetto la sola proprietà 'Abilita', in caso contrario considera l'oggetto abilitato se la variabile identificata da questa proprietà assume un valore diverso da zero.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim SchedObj As SchedulerCmdTarget
    Set SchedObj = GetScheduler("Sched1")
    If Not SchedObj Is Nothing Then
        MsgBox SchedObj.EnableVariable,vbOkOnly,""
    End If
    Set SchedObj = Nothing
End Sub
```

HasHolidays, SchedulerCmdTarget Property

Syntax HasHolidays = _Boolean

Description Questa proprietà consente di capire se l'oggetto schedatore è impostato per gestire le festività. La proprietà può anche essere modificata runtime. Se un oggetto schedatore non è impostato per gestire le festività, tutti i metodi inerenti ("AddHoliday", "RemoveHoliday", ecc.) non hanno effetto.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim SchedObj As SchedulerCmdTarget
    Set SchedObj = GetScheduler("Sched1")
    If Not SchedObj Is Nothing Then
        MsgBox SchedObj.HasHolidays,vbOkOnly, GetProjectTitle
    End If
    Set SchedObj = Nothing
End Sub
```

HolidaysPlan, SchedulerCmdTarget Property

Syntax HolidaysPlan = _Variant

Description Questa proprietà consente di impostare o leggere la pianificazione settimanale oraria corrente di uno schedatore di tipo "piano ore". La pianificazione oraria gestita da questa funzione è quella dei giorni festivi, mentre la pianificazione normale è gestita con il metodo "Plan".

Attenzione. Se lo Scheduler è visualizzato tramite l'oggetto "Visualizzatore Schedaltore", dopo aver modificato il file di programmazione è necessario utilizzare il metodo "Cancel" dell'oggetto "Visualizzatore Schedaltore" per avere un refresh dei dati visualizzati.

Parameter	Description
None	None

Result Variant

Example 1:

```
Option Explicit
Public Sub Click()
    Dim objScheduler As SchedulerCmdTarget
    Dim bHours() As Byte

    If objScheduler Is Nothing Then Set objScheduler = GetScheduler("Scheduler")

    Erase bHours
    For i = 0 To UBound(objScheduler.HolidaysPlan)
        ReDim Preserve bHours(i)
        SetVariableValue "Plan:Byte" & CStr(i), 255
        bHours(i) = 255
    Next
    objScheduler.HolidaysPlan = CVar(bHours)
End Sub
```

Example 2:

```
Option Explicit
Public Sub Click()
    Dim objScheduler As SchedulerCmdTarget
    Dim sPlan As String

    If objScheduler Is Nothing Then Set objScheduler = GetScheduler("Scheduler")

    For i = 0 To UBound(objScheduler.HolidaysPlan)
        If sPlan <> "" Then sPlan = sPlan & ", "
        sPlan = sPlan & objScheduler.HolidaysPlan(i)
    Next
    MsgBox sPlan, vbOkOnly, "Plan"
End Sub
```

Name, SchedulerCmdTarget Property

Syntax Name = _String

Description Questa proprietà restituisce una stringa con il nome dell'oggetto schedatore referenziato.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim SchedObj As SchedulerCmdTarget
    Set SchedObj = GetScheduler("Sched1")
End Sub
```

```

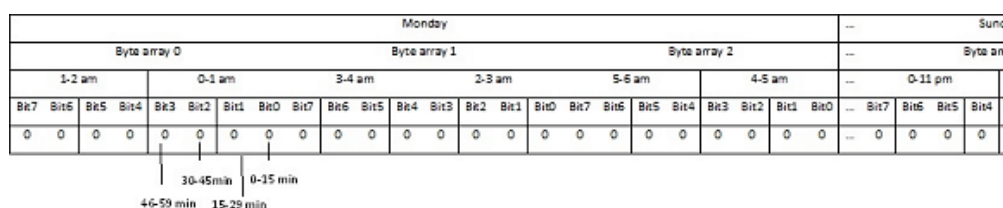
        If Not SchedObj Is Nothing Then
            MsgBox SchedObj.Name ,vbOkOnly,""
        End If
        Set SchedObj = Nothing
    End Sub

```

Plan, SchedulerCmdTarget Property

Syntax Plan = _Variant

Description Questa proprietà restituisce un array di 84 byte (0-83). La struttura prevede 1 byte per ora di planning, a partire da domenica. In pratica il primo byte contiene le informazioni dell'orario di domenica, il byte 1 l'orario 2-4 am di domenica e così via. Ci sono quindi 12 byte per ogni settimana. Pertanto per ogni ora di programmazione sono dedicati 4 bit. Quando la proprietà viene scritta è necessario compilare tutta l'array di byte prima di scrivere la proprietà, come mostrato nell'esempio sotto. La codifica risulta essere fatta in questo modo:



Il primo byte codifica l'orario da mezzanotte alle due di domenica. I 4 bit meno significativi codificano l'ora da mezzanotte all'una, mentre i più significativi codificano l'ora da una alle due. A partire dal bit 0 viene codificato il primo quarto d'ora e poi di seguito.

Attenzione. Se lo Schedulatore è visualizzato tramite l'oggetto "Visualizzatore Schedulatore", modificare il file di programmazione è necessario utilizzare il metodo "Cancel" dell'oggetto "Visualizzatore Schedulatore" per avere un refresh dei dati visualizzati.

Parameter	Description
None	None

Result Variant

Example:

```

Option Explicit
Public Sub Click()
    Dim objScheduler As SchedulerCmdTarget
    Dim i As Integer
    Dim sPlan As String
    Dim arrPlan(83) As Byte

    Set objScheduler = GetScheduler("Schedulatore")
    'Read actual Plan
    sPlan = ""
    For i = 0 To UBound(objScheduler.Plan)
        sPlan = sPlan & " " & CStr(objScheduler.Plan(i))
        arrPlan(i) = objScheduler.Plan(i)
    Next i

    Debug.Print "Old Plan = " & sPlan

    'Set new Plan
    sPlan = ""
    For i = 0 To 83
        arrPlan(i) = 51
    Next i

```

```

        sPlan = sPlan & " " & CStr(arrPlan(i))
    Next i

    'Set Plan property
    objScheduler.Plan() = arrPlan
    Debug.Print "New Plan = " & sPlan
    Set objScheduler = Nothing
End Sub

```

TimeAndDate, SchedulerCmdTarget Property

Syntax TimeAndDate = _Date

Description Questa proprietà imposta o restituisce l'orario di attivazione del comando, se si è selezionato un giorno o un mese nella proprietà "Tipo", se invece è stata selezionata la voce "Data Fissa", imposta o restituisce la data in cui dovrà essere attivato il comando.

Parameter	Description
None	None

Result Date

Example:

```

Option Explicit
Public Sub Click()
    Dim objScheduler As SchedulerCmdTarget

    Set objScheduler = GetScheduler("Scheduler1")
    If Not objScheduler Is Nothing Then
        MsgBox objScheduler.TimeAndDate, vbInformation, GetProjectTitle
    End If
    Set objScheduler = Nothing
End Sub

```

Type, SchedulerCmdTarget Property

Syntax Type = _Integer

Description Questa proprietà imposta o restituisce il "Tipo" di schedulazione.

Le impostazioni possibili sono:

```

enum_ST_DATE
enum_ST_DAY
enum_ST_FRIDAY
enum_ST_HOUR
enum_ST_MINUTE
enum_ST_MONDAY
enum_ST_MONTH
enum_ST_PLAN
enum_ST_SATURDAY
enum_ST_SUNDAY
enum_ST_THURSDAY
enum_ST_TUESDAY
enum_ST_WEDNESDAY

```

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim objScheduler As SchedulerCmdTarget

    Set objScheduler = GetScheduler("Scheduler1")
    If Not objScheduler Is Nothing Then
        objScheduler.Type = enum_ST_DATE
        objScheduler.TimeAndDate = CDate("11/04/2006 11:00:00")
    End If
    Set objScheduler = Nothing
End Sub
```

TreatHolidaysAsSunday, SchedulerCmdTarget Property

Syntax TreatHolidaysAsSunday = _Boolean

Description Questa proprietà consente di capire se l'oggetto schedulatore è impostato per gestire le festività con il piano ore normale della Domenica. La proprietà può essere modificata runtime. La proprietà viene considerata soltanto se lo schedulatore ha anche l'opzione di gestione delle festività attiva.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim SchedObj As SchedulerCmdTarget
    Set SchedObj = GetScheduler("Sched1")
    If Not SchedObj Is Nothing Then
        MsgBox SchedObj.TreatHolidaysAsSunday, vbOkOnly, GetProjectTitle
    End If
    Set SchedObj = Nothing
End Sub
```

1.17.5. ScriptMEInterface

Events

Loading, ScriptMEInterface Event

Description Evento notificato quando la risorsa Basic Script viene caricata in memoria ed eseguita la prima volta. Dopo la prima volta la routine "Loading" non verrà più eseguita a meno che non sia stato impartito un comando di "Scarica" del Basic Script.

Parameter	Description
None	None

Main, ScriptMEInterface Event

Description Evento che viene eseguito ogni volta che la risorsa Basic Script viene messa in esecuzione da un comando del progetto. La prima volta che lo Script viene richiamato verrà eseguita prima la routine "Loading" e poi la routine "Main". Dopo la prima volta soltanto la routine "Main" verrà eseguita a meno che non sia stato impartito un comando di "Scarica" del Basic Script.

Parameter	Description
None	None

Unloading, ScriptMEInterface Event

Description Evento notificato quando la risorsa Basic Script viene scaricata dalla memoria a fronte di un comando di "Scarica" oppure all'arresto del progetto.

Parameter	Description
None	None

Functions

EnterGlobalCriticalSection, ScriptMEInterface Function

Syntax EnterGlobalCriticalSection(_nTimeout)

Description Quesa istruzione abilita l'entrata in modalità di elaborazione critica per la sincronizzazione tra basic in thread separati. Significa che il codice seguente all'istruzione verrà considerato prioritario e determinerà la sincronizzazione "bloccando" eventuali altri basic che eseguiranno la medesima istruzione, fino al raggiungimento dell'istruzione "LeaveCriticalSection". Il tempo di timeout in millisecondi sarà pari al valore passato con nTimeout.

Parameter	Description
nTimeout As Long	Valore di timeout.

Result Boolean

Example:

```
Sub Main
    Dim vResult As Long
    vResult = This.EnterGlobalCriticalSection(5000)
    MsgBox "EnterGlobalCriticalSection = " &
    CBool(vResult), vbInformation, GetProjectTitle
    vResult = This.LeaveGlobalCriticalSection
    MsgBox "LeaveGlobalCriticalSection = " &
    CBool(vResult), vbInformation, GetProjectTitle
End Sub
```

GetCurrentUser, ScriptMEInterface Function

Syntax GetCurrentUser()

Description Restituisce l'utente attualmente loggato.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo UserCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Sub Main
    Dim vResult As Object
    Set vResult = This.GetCurrentUser
    'do something
    Set vResult = Nothing
End Sub
```

GetInstanceNumber, ScriptMEInterface Function

Syntax GetInstanceNumber()

Description Questa funzione restituisce il numero di istanze relative al basic script referenziato.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    Dim vResult As Long
    vResult = This.GetInstanceNumber()
    MsgBox "GetInstanceNumber = " & vResult,vbInformation,GetProjectTitle
End Sub
```

GetParameter, ScriptMEInterface Function

Syntax GetParameter(_nParam)

Description Questa funzione restituisce il parametro di indice uguale a nParam con cui è stato chiamato il basic script.

Parameter	Description
nParam As Integer	Indice del parametro.

Result String

Example:

```
Sub Main
    Dim vResult As String
    vResult = This.GetParameter(0)
    MsgBox "Parameter0 = " & vResult,vbInformation,GetProjectTitle
End Sub
```

HasPreviousInstance, ScriptMEInterface Function

Syntax HasPreviousInstance()

Description Questa funzione restituisce l'informazione relativa al fatto che il basic script possa essere già stato istanziato.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim vResult As Boolean
    vResult = This.HasPreviousInstance
    MsgBox "HasPreviousInstance = " & vResult,vbInformation,GetProjectTitle
End Sub
```

IsStopping, ScriptMEInterface Function

Syntax IsStopping()

Description Questa funzione restituisce true se qualcuno sta cercando di far terminare il basic script.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim vResult As Boolean
    Do
        vResult = This.IsStopping
    If vResult Then
        '...
        'Do something
        MsgBox "IsStopping = " & vResult,vbInformation,GetProjectTitle
        '...
    End If
    DoEvents
    Loop Until IsInStoppingMode Or vResult
End Sub
```

LeaveGlobalCriticalSection, ScriptMEInterface Function

Syntax LeaveGlobalCriticalSection()

Description Questa istruzione lascia la modalità di elaborazione critica per la sincronizzazione di basic multithread. Significa che il codice seguente all'istruzione non verrà più considerato prioritario, ma normale. Questa istruzione annulla l'istruzione "EnterCriticalSection" e rilascerà l'esecuzione al primo Basic Script pendente sull'istruzione 'EnterCriticalSection'.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim vResult As Long
    vResult = This.EnterGlobalCriticalSection(5000)
    MsgBox "EnterGlobalCriticalSection = " &
    CBool(vResult),vbInformation,GetProjectTitle
    vResult = This.LeaveGlobalCriticalSection
    MsgBox "LeaveGlobalCriticalSection = " &
    CBool(vResult),vbInformation,GetProjectTitle
End Sub
```

RunningOnServerSide, ScriptMEInterface Function

Syntax RunningOnServerSide()

Description Questa restituisce true se lo script sta girando lato server.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim vResult As Boolean
    vResult = This.RunningOnServerSide()
    MsgBox "RunningOnServerSide = " & vResult,vbInformation,GetProjectTitle
End Sub
```

ShowDebuggerWnd, ScriptMEInterface Function

Syntax ShowDebuggerWnd()

Description Questa istruzione permette di visualizzare la finestra di debug del basic script.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    Dim vResult As Boolean
    vResult = This.ShowDebuggerWnd
    MsgBox "ShowDebuggerWnd = " & vResult,vbInformation,GetProjectTitle
End Sub
```

Properties

MaxInstances, ScriptMEInterface Property

Syntax MaxInstances = _Long

Description Questa proprietà imposta o restituisce il valore relativo alla proprietà numero massimo contemporaneo di istanze di un basic script.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    MsgBox "MaxInstances = " & This.MaxInstances , vbOkOnly, GetProjectTitle
End Sub
```

ModalDialog, ScriptMEInterface Property

Syntax ModalDialog = _Boolean

Description Questa proprietà imposta o restituisce il valore di abilitazione per la proprietà 'finestre di dialogo modali'. Se impostata a vero fa in modo che le finestre di dialogo e le message box richiamate all'interno dello script siano di tipo modale.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    MsgBox "ModalDialog = " & This.ModalDialog, vbOkOnly, GetProjectTitle
End Sub
```

RunAtServer, ScriptMEInterface Property

Syntax RunAtServer = _Boolean

Description Questa proprietà imposta o restituisce il valore relativo alla proprietà 'Esegui lato Server' di un basic script. Il valore impostato risulta true se lo script viene eseguito lato server.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    MsgBox "RunAtServer = " & This.RunAtServer, vbOkOnly, GetProjectTitle
End Sub
```

SeparateThread, ScriptMEInterface Property

Syntax SeparateThread = _Boolean

Description Questa proprietà imposta o restituisce il valore relativo alla proprietà 'Esegui in Thread Separato' di un basic script. Il valore impostato risulta true se nello script viene eseguito in thread separato.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    MsgBox "SeparateThread = " & This.SeparateThread, vbOkOnly,
    GetProjectTitle
End Sub
```

SleepExecution, ScriptMEInterface Property

Syntax SleepExecution = _Long

Description Questa proprietà imposta o restituisce il valore (in millisecondi) relativo alla proprietà sleep di un basic script. Più alto sarà il tempo di sleep più il basic script impegnerà meno il processore e di conseguenza rallenterà la sua esecuzione.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    MsgBox "SleepExecution = " & This.SleepExecution, vbOkOnly, GetProjectTitle
End Sub
```

StatusVariable, ScriptMEInterface Property

Syntax StatusVariable = _String

Description Questa proprietà imposta o restituisce il nome della variabile relativa alla proprietà variabile di stato di un basic script.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    MsgBox "StatusVariable = " & This.StatusVariable , vbOkOnly, GetProjectTitle
End Sub
```

SyncroScriptTimeout, ScriptMEInterface Property

Syntax SyncroScriptTimeout = _Long

Description Questa proprietà imposta o restituisce il valore (in millisecondi) relativo alla proprietà SincroTimeout di un basic script. Il valore impostato si riferisce ai millisecondi necessari prima di terminare uno script sincrono.

Parameter	Description
None	None

Result Long

Example:

```
Sub Main
    MsgBox "SyncroScriptTimeout = " & This.SyncroScriptTimeout, vbOkOnly,
    GetProjectTitle
End Sub
```

ThreadPriority, ScriptMEInterface Property

Syntax ThreadPriority = _Byte

Description Questa proprietà imposta o restituisce il valore relativo alla proprietà priorità di esecuzione di un basic script.

La proprietà può assumere i seguenti valori:

- 0 Al di sotto del normale
- 1 Normale
- 2 Al di sopra del normale.

Parameter	Description
None	None

Result Byte

Example:

```
Sub Main
    MsgBox "ThreadPriority = " & This.ThreadPriority, vbOkOnly, GetProjectTitle
End Sub
```

UseOwnTrace, ScriptMEInterface Property

Syntax UseOwnTrace = _Boolean

Description Questa proprietà permette di registrare i messaggi di debug in una tabella apposita di log del basic script.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    MsgBox "UseOwnTrace = " & This.UseOwnTrace, vbOkOnly, GetProjectTitle
End Sub
```

UseUIInterface, ScriptMEInterface Property

Syntax UseUIInterface = _Boolean

Description Questa proprietà imposta o restituisce il valore relativo alla proprietà 'Inserisci la UI Interface' al basic script. Il valore impostato risulta true se nello script viene inserita la UI Interface. Se inserita è possibile accedere all'interno del basic script ai metodi e alle proprietà della stessa definiti nell'interfaccia UIInterface.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    MsgBox "UseUIInterface = " & This.UseUIInterface, vbOkOnly, GetProjectTitle
End Sub
```

1.17.6. SynopticCmdTarget

Events

Click, Generic Event

Description Evento notificato quando viene premuto il tasto sinistro o destro del mouse all'interno dell'area del disegno.

Parameter	Description
Nessuno	Nessuno

DbClick, Generic Event

Description Evento generato quando viene premuto per due volte in rapida successione il tasto sinistro o destro del mouse all'interno dell'area del disegno. I tempi del doppio click sono stabiliti nelle impostazioni del sistema operativo.

Parameter	Description
Nessuno	Nessuno

KeyDown, Generic Event

Description Evento notificato quando viene premuto un tasto della tastiera. L'evento restituisce le variabili, di tipo integer, KeyCode e Shift. L'evento viene generato indipendentemente dal focus.

Parameter	Description
KeyCode As Integer	Codice VBA del tasto premuto. Il codice VBA e' un insieme di costanti che, oltre ai normali caratteri alfanumeri, senza pero' distinzione di maiuscolo o minuscolo, contempla anche gli altri tasti della tastiera come, ad esempio i tasti funzione, il Caps Lock, ecc.
Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT

KeyPress, Generic Event

Description Evento notificato quando viene premuto e rilasciato un tasto della tastiera. L'evento restituisce la variabile di tipo integer KeyAscii contenente il codice ASCII del tasto premuto. L'evento viene generato solo se il focus è fissato sul disegno.

Parameter	Description
Keyascii As Integ	Codice ASCII del tasto premuto

KeyUp, Generic Event

Description Evento generato quando viene rilasciato un tasto (dopo essere stato premuto) della tastiera. L'evento restituisce le variabili di tipo integer, KeyCode e Shift. L'evento viene generato indipendentemente dal focus.

Parameter	Description
KeyCode As Integer	Codice VBA del tasto premuto. Il codice VBA e' un insieme di costanti che, oltre ai normali caratteri alfanumeri, senza pero' distinzione di

	maiuscolo o minuscolo, contempla anche gli altri tasti della tastiera come, ad esempio i tasti funzione, il Caps Lock, ecc.
Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT

KillFocus, Generic Event

Description Evento generato quando l'oggetto in questione viene deselezionato oppure perde il focus.

Parameter	Description
None	None

MouseDown, Generic Event

Description Evento notificato sia nel codice del sinottico sia nel codice degli oggetti quando viene premuto un qualsiasi pulsante del mouse nel sinottico, indipendentemente dalla posizione o dal simbolo. L'evento restituisce le variabili, di tipo integer, Button e Shift e le variabili, di tipo single, X e Y.
Per poter gestire questo evento soltanto all'interno di un oggetto del sinottico è necessario utilizzare la funzione "IsCursorOnObject".

Parameter	Description
Button As Integer	Indice del tasto mouse premuto: 1 = Sinistro 2 = Destro 4 = Centrale
Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT
X As Single	Coordinata orizzontale riferita alla posizione del cursore al momento dell'evento
Y As Single	Coordinata verticale riferita alla posizione del cursore al momento dell'evento

MouseMove, Generic Event

Description Evento notificato sia nel codice del sinottico sia nel codice degli oggetti quando il cursore del mouse cambia la propria posizione nel sinottico, indipendentemente dalla posizione o dal simbolo. L'evento restituisce le variabili, di tipo integer, Button e Shift e le variabili, di tipo single, X e Y.
Per poter gestire questo evento soltanto all'interno di un oggetto del sinottico è necessario utilizzare la funzione "IsCursorOnObject".

Parameter	Description
Button As Integer	Indice del tasto mouse premuto: 1 = Sinistro 2 = Destro 4 = Centrale

Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT
X As Single	Coordinata orizzontale riferita alla posizione del cursore al momento dell'evento
Y As Single	Coordinata verticale riferita alla posizione del cursore al momento dell'evento

MouseUp, Generic Event

Description Evento notificato sia nel codice del sinottico sia nel codice degli oggetti quando viene rilasciato un qualsiasi pulsante del mouse nel sinottico, indipendentemente dalla posizione o dal simbolo. L'evento restituisce le variabili, di tipo integer, Button e Shift e le variabili, di tipo single, X e Y. Per poter gestire questo evento soltanto all'interno di un oggetto del sinottico è necessario utilizzare la funzione "IsCursorOnObject".

Parameter	Description
Button As Integer	Indice del tasto mouse premuto: 1 = Sinistro 2 = Destro 4 = Centrale
Shift As Integer	Indica se sono premuti i tasti Shift, Ctrl e Alt: 1 = SHIFT 2 = CTRL 4 = ALT
X As Single	Coordinata orizzontale riferita alla posizione del cursore al momento dell'evento
Y As Single	Coordinata verticale riferita alla posizione del cursore al momento dell'evento

OnActivate, SynopticCmdTarget Event

Description Evento generato quando il sinottico viene reso attivo o disattivo. L'evento restituisce un parametro booleano indicante lo stato dell'attivazione, ovvero se ha il focus oppure no.

Parameter	Description
bActive As Boolean	Stato dell'attivazione: True = il sinottico è stato attivato False = il sinottico è stato disattivato

OnQueryEndSession, SynopticCmdTarget Event

Description Evento generato a seguito del comando di chiusura di un sinottico. Tramite il parametro bRet, impostandolo a false, è possibile impedire la chiusura del sinottico.

Parameter	Description
bRet As Boolean	Abilitazione alla chiusura: True = consente la chiusura del sinottico False = impedisce la chiusura del sinottico

OnSize, SynopticCmdTarget Event

Description Questo evento si verifica al ridimensionarsi della finestra del sinottico (non quando viene minimizzato)

Parameter	Description
nWidth As Integer	Parametro di uscita. Ritorna la larghezza del sinottico.
nHeight As Integer	Parametro di uscita. Ritorna l'altezza del sinottico.

Example:

```
Public Sub OnSize(ByRef nWidth As Integer, ByRef nHeight As Integer)
    MsgBox "Screen width = " & nWidth & ", Screen height = " & nHeight
End Sub
```

OnStartSynapsisExecution, SynopticCmdTarget Event

Description Evento generato all'inizio dell'esecuzione della logica di tipo sinapsi sul sinottico.

Parameter	Description
None	None

OnStopSynapsisExecution, SynopticCmdTarget Event

Description Evento generato al termine dell'esecuzione della logica di tipo sinapsi sul sinottico.

Parameter	Description
None	None

OnTimer, Generic Event

Description Evento notificato con un periodo di 1/2 di secondo circa (tempo non garantito) durante la fase di run-time. Durante il Test il periodo è proporzionale alla velocità di test impostata. Il tempo di esecuzione dell'evento può comunque essere personalizzato tramite la chiave di registro TimerEventFrequence.

Parameter	Description
None	None

SetFocus, Generic Event

Description Evento notificato quando l'oggetto disegno riceve il focus o viene selezionato.

Parameter	Description
None	None

SynopticLoading, SynopticCmdTarget Event

Description Evento notificato quando il Sinottico viene caricato in memoria.

Parameter	Description
None	None

SynopticUnloading, SynopticCmdTarget Event

Description Evento notificato quando il Sinottico viene scaricato dalla memoria.

Parameter	Description
None	None

Functions

CloseSynoptic, SynopticCmdTarget Function

Syntax CloseSynoptic()

Description Questa funzione permette di chiudere il sinottico. La funzione non ha alcun effetto se richiamata nel sinottico di avvio. Tipicamente viene utilizzata per chiudere sinottici aperti in maniera modale o in un frame separato. Una volta chiamata si ritorna sul sinottico di avvio.

Parameter	Description
None	None

Result None

Example:

```
Public Sub Click()  
    CloseSynoptic()  
End Sub
```

CreateNewSymbol, SynopticCmdTarget Function

Syntax CreateNewSymbol(_lpszSymbolName, _lpszSymbolCode, nSymbolType)

Description Questa funzione consente di creare un nuovo simbolo nel sinottico. Accetta due parametri string contenenti rispettivamente il nome nel simbolo e il codice basic interno al simbolo stesso e un parametro integer indicante il tipo di simbolo. Tutte le caratteristiche del nuovo simbolo potranno essere configurate mediante le opportune proprietà basic. Sebbene non ci siano limiti alla capacità di una variabile stringa, può essere utile appoggiarsi a un file di testo esterno per poter scrivere in modo più semplice l'intero codice.

0 = rettangolo
1 = rettangolo smussato
2 = arco
3 = corda
4 = torta
5 = ellisse
6 = linea
7 = testo
8 = sinottico incastrato
9 = Trend



Questa proprietà è supportata solo in parte su Windows CE. (non ammessa la creazione di un arco, corda, torta)

Parameter	Description
lpszSymbolName As String	Nome dell'oggetto.
lpszSymbolCode As String	Codice basic legato all'oggetto.
nSymbolType As Integer	Tipo di oggetto.

Result Object
Viene restituito un oggetto di tipo DrawCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub Click()  
    Dim lpszSymbolName As String  
    Dim lpszSymbolCode As String  
    Dim nSymbolType As Integer  
    Dim sType As String  
    Dim obj As Object  
    Dim bErr As Boolean  
    ' Ask the type (Chide il tipo)  
    sType = InputBox("Symbol's type:", "CreateNewSymbol", "0", 100, 100)  
    If sType <> "" And IsNumeric(sType) Then  
        lpszSymbolName = "MySymbol"  
        nSymbolType = CInt(sType)  
        lpszSymbolCode = ""#Uses ""Codice.txt""  
        ' Destroy the symbol (Distugge il simbolo)  
        DestroySymbol(lpszSymbolName)  
        ' Create the symbol (Crea il simbolo)  
        bErr = False  
        On Error GoTo NoObj  
        Set obj = CreateNewSymbol(lpszSymbolName, "", nSymbolType)  
        ' Show the symbol (Visualizza il simbolo)  
        obj.Height = 100  
        obj.Width = 100  
        obj.Xpos = 500  
        obj.Ypos = 300  
        obj.BackColor = RGB(192,192,192)  
        On Error GoTo 0
```

```

        Set obj = Nothing
    End If
Exit Sub
NoObj:
If Not bErr Then
    Debug.Print "Object is Nothing: " & nSymbolType
    bErr = True
End If
Resume Next
End Sub

```

DestroySymbol, SynopticCmdTarget Function

Syntax DestroySymbol(_IpszSymbolName)

Description Questa funzione consente di eliminare un simbolo dal sinottico. Accetta un parametro string contenente il nome del simbolo da eliminare. Questa funzione puo' agire sia su simboli introdotti durante la programmazione che su simboli precedentemente creati mediante la funzione CreateNewSymbol. La funzione ritorna un valore boolean che sara' True nel caso di eliminazione con successo e False nel caso contrario. Un motivo di fallimento puo' essere cercare di eliminare un simbolo non esistente.

Parameter	Description
IpszSymbolName As String	Nome dell'oggetto.

Result Boolean

Example:

```

Public Sub Click()
    Dim IpszSymbolName As String
    Dim IpszSymbolCode As String
    Dim nSymbolType As Integer
    Dim sType As String
    Dim obj As Object
    Dim bErr As Boolean
    ' Ask the type (Chide il tipo)
    sType = InputBox("Symbol's type:", "CreateNewSymbol", "0", 100, 100)
    If sType <> "" And IsNumeric(sType) Then
        IpszSymbolName = "MySymbol"
        nSymbolType = CInt(sType)
        IpszSymbolCode = ""#Uses ""Codice.txt""
        ' Destroy the symbol (Distrugge il simbolo)
        DestroySymbol(IpszSymbolName)
        ' Create the symbol (Crea il simbolo)
        bErr = False
        On Error GoTo NoObj
        Set obj = CreateNewSymbol(IpszSymbolName, "", nSymbolType)
        ' Show the symbol (Visualizza il simbolo)
        obj.Height = 100
        obj.Width = 100
        obj.Xpos = 500
        obj.Ypos = 300
        obj.BackColor = RGB(192,192,192)
        On Error GoTo 0
        Set obj = Nothing
    End If
    'Wait 1 second (aspetta un secondo)
    Wait 5
    DestroySymbol(IpszSymbolName)
Exit Sub
NoObj:
If Not bErr Then
    Debug.Print "Object is Nothing: " & nSymbolType

```

```

        bErr = True
    End If
    Resume Next
End Sub

```

GetAbsoluteSubObject, SynopticCmdTarget Function

Syntax GetAbsoluteSubObject(_IpszName)

Description Questa funzione permette di accedere ai metodi e alle proprietà di un oggetto all'interno del sinottico anche se l'oggetto è contenuto in un simbolo. Il parametro IpszName è il nome dell'oggetto.



Questa funzione non è supportata in Windows CE.(se utilizzata restituisce sempre null')

Parameter	Description
IpszName As String	Nome dell'oggetto.

Result Object
Viene restituito un oggetto di tipo DrawCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```

Public Sub Click()
    Dim obj As Object
    ' Create object (Crea oggetto)
    Set obj = GetAbsoluteSubObject("Object1")
    obj.BackColor = RGB(192,192,192)
    Set obj = Nothing
End Sub

```

GetActiveUserObject, SynopticCmdTarget Function

Syntax GetActiveUserObject()

Description Questa funzione permette di reperire l'oggetto utente attivo per il sinottico. Con questo metodo è possibile sapere con quale utente un Web Client si è autenticato al Server. Se nessun utente ha effettuato il login, la funzione restituisce un oggetto Nothing.
L'omonimo metodo dell'interfaccia basic 'UserAndGroupCmdTarget' non può essere utilizzato allo scopo precedentemente descritto. Infatti questo metodo, se usato per un Web Client, restituisce l'ultimo utente che si è autenticato sul Server e non quello attualmente autenticato sul Web Client.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo UserCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```

Option Explicit
Public Sub Click()
    Dim objUser As UserCmdTarget

    Set objUser = GetSynopticObject.GetActiveUserObject
    If Not objUser Is Nothing Then
        MsgBox(objUser.Name & " is Logged In", vbOkOnly, GetProjectTitle)
        Set objUser = Nothing
    End If
End Sub

```

GetAlias, SynopticCmdTarget Function

Syntax GetAlias(_IpszAlias)

Description Questa funzione restituisce il valore definito per l'Alias passato come parametro "IpszAlias" definito nella Tabella Alias del sinottico.

Parameter	Description
IpszAlias As String	Nome dell'Alias per il quale recuperare il valore.

Result String

Example:

```

Public Sub Click()
    Dim objScreen As SynopticCmdTarget
    Set objScreen = GetSynopticObject
    MsgBox "Alias <<TsetAlais>> = "
    objScreen.GetAlias("TsetAlais"),vbInformation, GetProjectTitle
    Set objScreen = Nothing
End Sub

```

GetAliasListName, SynopticCmdTarget Function

Syntax GetAliasListName()

Description Questa funzione restituisce la lista degli Alias definiti nel sinottico. Viene restituita una stringa dove i nomi degli Alias sono separati dal carattere "|" (pipe).

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim objScreen As SynopticCmdTarget
    Set objScreen = GetSynopticObject
    MsgBox "Alias List = " & objScreen.GetAliasListName(),vbInformation,
    GetProjectTitle

```

```
        Set objScreen = Nothing
End Sub
```

GetAliasListValue, SynopticCmdTarget Function

Syntax GetAliasListValue()

Description Questa funzione restituisce la lista dei valori associati agli Alias definiti nel sinottico. Viene restituita una stringa dove i valori degli Alias sono separati dal carattere "|" (pipe).

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim objScreen As SynopticCmdTarget
    Set objScreen = GetSynopticObject
    MsgBox "Alias Value List = " & objScreen.GetAliasListValue(), vbInformation,
    GetProjectTitle
    Set objScreen = Nothing
End Sub
```

GetFocus, SynopticCmdTarget Function

Syntax GetFocus()

Description Questa funzione permette di accedere alle proprietà e i metodi dell'oggetto disegno che possiede il focus all'interno del sinottico. Se nessun disegno possiede il focus la variabile oggetto è Nothing.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo DrawCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
    Dim obj As Object

    ' Create object (Crea l'oggetto)
    Set obj = GetFocus
    ' Print title (Stampa titolo)
    On Error Resume Next
    Debug.Print obj.title
    On Error GoTo 0
    ' Delete object (Distrugge l'oggetto)
    Set obj = Nothing
End Sub
```

GetImage, SynopticCmdTarget Function

Syntax GetImage(_nImageType)

Description Questa funzione permette di prelevare l'immagine in memoria con una determinata qualità. La qualità dell'immagine viene definita dal parametro nImageType. Il risultato contiene l'immagine.

Il parametro nImageType può assumere i seguenti valori:
0 alta qualità
1 medio-alta qualità
2 medio-bassa qualità
3 bassa qualità



Questa funzione non è pienamente supportata in Windows CE .(L'immagine dovrebbe tornare sempre e solo come bitmap')



La funzione eseguirà il caricamento dell'immagine soltanto se il sinottico è aperto. Non è infatti sufficiente che l'opzione "Mantieni in memoria" del sinottico sia marcata in quanto anche se il sinottico viene mantenuto in memoria in realtà la sua immagine, una volta chiusa la pagina, viene comunque scaricata, pertanto la funzione GetImage non sarà più in grado di caricare l'immagine. La funzione GetImage ritornerà un variant contenente l'immagine se l'esecuzione è andata a buon fine, mentre ritornerà un variant vuoto se l'immagine non è stata caricata.

Parameter	Description
nImageType As Integer	Qualità dell'immagine.

Result Variant

Example:

```
Public Sub Click()  
    Dim vResult As Variant  
    Dim nImageType As Integer  
    nImageType = 1  
    vResult = GetImage(nImageType)  
End Sub
```

GetInstanceNumber, SynopticCmdTarget Function

Syntax GetInstanceNumber()

Description Questa funzione ritorna l'indice assegnato all'istanza corrente con cui il sinottico è stato aperto: ogni istanza presenta pertanto un numero d'indice differente.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    Dim objScreen As SynopticCmdTarget  
  
    Set objScreen = GetSynopticObject  
    Debug.Print objScreen.GetInstanceNumber()  
    Set objScreen = Nothing  
End Sub
```

GetObjectByUniqueID, SynopticCmdTarget Function

Syntax GetObjectByUniqueID(_ID)

Description Questa funzione permette di accedere alle proprietà e metodi di un oggetto all'interno del sinottico. L'oggetto deve contenere al suo interno delle sinapsi per poter essere referenziato con il parametro ID. Movicon in automatico assegna ad ogni oggetto contenente delle sinapsi un identificatore numerico univoco e il parametro ID serve appunto per indicare l'oggetto al quale si vuole accedere. Nella libreria SymbolInterface è presente la funzione GetUniqueObjectID che restituisce l'identificatore numerico di un disegno.

Parameter	Description
ID As Long	ID dell'oggetto.

Result Object
Viene restituito un oggetto di tipo DrawCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Public Sub OnStartSynapsisExecution()  
    Dim obj as Object  
  
    'Create object (Crea oggetto)  
    Set obj = GetObjectByUniqueID(3)  
    ' Changhe color (Cambia colore)  
    On Error GoTo NoObj  
    obj.BackColor = RGB(192,192,192)  
    On Error GoTo 0  
    Set obj = Nothing  
    Exit Sub  
  
    NoObj:  
    Debug.Print "Not object ID: 3"  
    Resume Next  
End Sub
```

GetParameterVariable, SynopticCmdTarget Function

Syntax GetParameterVariable(_IpszVariableName)

Description Questa funzione restituisce il nome della variabile che sostituisce la variabile referenziata col parametro IpszVariableName nel file di parametrizzazione del sinottico. Per parametrizzazione di un sinottico si intende la possibilità di sostituire, durante la fase di runtime, le variabili utilizzate nelle proprietà di esecuzione dei disegni con altre specificate appunto nel file di testo

in oggetto. Per l'impostazione del file di parametrizzazione vedere la proprietà `ParameterFileName`.

Parameter	Description
<code>IpszVariableName</code> As String	Nome della variabile.

Result String

Example:

```
Public Sub Click()  
    Dim IpszVariableName As String  
    Dim sResult As String  
  
    ParameterFileName = "Parametri.txt"  
    IpszVariableName = InputBox("Variable name?", "", "VAR00001", 100, 100)  
    sResult = GetParameterVariable(IpszVariableName)  
    MsgBox "GetParameterVariable= " & sResult , vbOkOnly, GetProjectTitle  
End Sub
```

GetSubObject, SynopticCmdTarget Function

Syntax `GetSubObject(_IpszObjectName)`

Description Questa funzione consente di referenziare un simbolo contenuto nel sinottico. Accetta un parametro string contenente il nome dell'oggetto a cui fare riferimento. Si tenga conto che nel caso di oggetti semplici deve essere passato il contenuto della proprietà "Nome" dell'oggetto, mentre nel caso di simboli composti deve essere passato il contenuto della proprietà "Descrizione" dell'oggetto.

Parameter	Description
<code>IpszObjectName</code> As String	Nome dell'oggetto.

Result Object
Viene restituito un oggetto di tipo `DrawCmdTarget` se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà `Nothing`.

Example:

```
Public Sub Click()  
    Dim vResult As DrawCmdTarget  
    Set vResult = GetSubObject("Object1")  
    vResult.BackColor = vResult.BackColor + 10  
    Set vResult = Nothing  
End Sub
```

GetTimeZone, SynopticCmdTarget Function

Syntax `GetTimeZone()`

Description Questa funzione consente di conoscere il time zone in minuti impostato nel computer in cui il sinottico è visualizzato. Nel caso in cui il sinottico venga visualizzato da una postazione Web Client il valore indicherà il time zone del computer in cui risiede il browser che sta gestendo la pagina html.
Il time zone restituito tiene sempre conto dell'eventuale ora legale.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    Dim sMsg As String
    sMsg = CStr(GetSynopticObject.GetTimeZone) & " min."
    MsgBox "Time Zone : " & sMsg, vbOkOnly + vbInformation, GetProjectTitle
End Sub
```

GetWindowPos, SynopticCmdTarget Function

Syntax GetWindowPos(_nX, _nY, _nWidth, _nHeight, _nShow)

Description Questa funzione è di sola lettura e permette di leggere informazioni sulla dimensione e posizione del sinottico referenziato.



Questa funzione è supportata solo in parte in Windows CE. (nShow può assumere solo i valori '0'(SW_HIDE) o '5'(SW_SHOW))

Nel caso del desktop possiamo avere i seguenti valori:

SW_HIDE = 0

nasconde la finestra corrente e attiva un'altra finestra

SW_SHOWNORMAL = 1

attiva e visualizza la finestra ,se la finestra era minimizzata o massimizzata ,il sistema la riporterà alle sue dimensioni originali (un'applicazione richiederà la specifica di questo valore quando apparirà nella finestra per la prima volta)

SW_SHOWMINIMIZED = 2

attiva la finestra e la visualizza se era minimizzata

SW_SHOWMAXIMIZED = 3

Attiva la finestra e la visualizza se era massimizzata

SW_SHOWNOACTIVATE = 4

visualizza una finestra mantenendo dimensioni e posizione. Questo valore è simile al SW_SHOWNORMAL, tranne per il fatto che la finestra non viene attivata.

SW_SHOW = 5

Attiva la finestra e la visualizza nelle sue dimensioni e posizione attuali.

SW_MINIMIZE = 6

Minimizza la finestra specificata e attiva la finestra successiva nell'ordine di tabulazione

SW_SHOWMINNOACTIVE = 7

Visualizza la finestra come finestra minimizzata .

questo valore è simile al SW_SHOWMINIMIZED, tranne per il fatto che la finestra non viene attivata

SW_SHOWNA = 8

Visualizza la finestra nelle sue attuali posizioni e dimensione. Questo valore è simile al SW_SHOW, tranne per il fatto che la finestra non viene attivata

SW_RESTORE = 9

Attiva e visualizza la finestra. Se la finestra era minimizzata o massimizzata ,il sistema la riporterà alle sue dim.e posizione originali.(un'applicazione richiederà la specifica di questo valore quando apparirà nella finestra per la prima volta)

SW_SHOWDEFAULT = 10

SW_FORCEMINIMIZE = 11

Parameter	Description
nX As Integer	Variabile di appoggio per la coordinata X angolo alto sinistra.
nY As Integer	Variabile di appoggio per la coordinata Y angolo alto sinistra.

nWidth As Integer	Variabile di appoggio per la dimensione larghezza.
nHeight As Integer	Variabile di appoggio per la dimensione altezza.
nShow As Integer	Variabile di appoggio per la modalità di visualizzazione del sinottico.

Result Boolean

Example:

```
Public Sub Click()
    Dim nX As Variant
    Dim nY As Variant
    Dim nWidth As Variant
    Dim nHeight As Variant
    Dim nShow As Variant
    GetWindowPos(nX , nY , nWidth , nHeight , nShow )
    MsgBox "nX = " & nX & " nY = " & nY & "nWidth = " & nWidth & "nHeight = " &
    nHeight & "nShow = " & nShow , vbOkOnly, GetProjectTitle
End Sub
```

IsRemoteClientView, SynopticCmdTarget Function

Syntax IsRemoteClientView()

Description Restituisce True se il codice basic e' stato invocato da un oggetto WebClient di Movicon.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    MsgBox "IsRemoteClientView = " & CStr(IsRemoteClientView), vbOkOnly,
    GetProjectTitle
End Sub
```

PrintSynoptic, SynopticCmdTarget Function

Syntax PrintSynoptic(_nMode, _bKeepPrintProportions)

Description Questa funzione effettua la stampa andando a ricaricare in background il Sinottico rieseguendo pertanto la SynopticLoading(). Per questo motivo eventuali modifiche ad oggetti dinamici, effettuate dopo il caricamento del Sinottico stesso, non vengono riportate nella stampa. Accetta il parametro nMode indicante il modo di stampa.

Il parametro può assumere i seguenti valori:
0=seleziona stampante
1=stampa diretta
2=Anteprima

Parameter	Description
nMode As Integer	Modalità di stampa.
bKeepPrintProportions as boolean	Parametro Opzionale. Se impostato a "True" permette di stampare mantenendo le proporzioni che si vedono sullo schermo, mentre se impostato a "False" (valore di default) vengono adattate sia l'altezza che la larghezza all'intera dimensione della pagina.

Result Boolean

Example:

```
Public Sub Click()
    PrintSynoptic(2)
End Sub
```

RemoveAlias, SynopticCmdTarget Function

Syntax RemoveAlias(_IpszAlias)

Description Questa funzione consente di eliminare l'Alias passato come parametro "IpszAlias".

Parameter	Description
IpszAlias As String	Nome dell'Alias da eliminare.

Result Boolean

Example:

```
Public Sub Click()
    Dim objScreen As SynopticCmdTarget
    Set objScreen = GetSynopticObject
    Debug.Print objScreen.RemoveAlias("TsetAlais")
    Set objScreen = Nothing
End Sub
```

RemoveAllAliases, SynopticCmdTarget Function

Syntax RemoveAllAliases()

Description Questa funzione consente di eliminare tutti gli Alias definiti nella Tabella del sinottico.

Parameter	Description
None	None

Result None

Example:

```
Public Sub Click()
    Dim objScreen As SynopticCmdTarget
    Set objScreen = GetSynopticObject
    Debug.Print objScreen.RemoveAllAliases()
    Set objScreen = Nothing
End Sub
```

SaveImageToFile, SynopticCmdTarget Function

Syntax

SaveImageToFile(_lpszFileName, _nImageType)

Description

Questa funzione permette di salvare come immagine il sinottico visualizzato sullo schermo. La qualità di stampa viene definita dal parametro nImageType. I valori che può assumere vanno da 0 a 3.

0 = alta qualità
 1 = medio-alta qualità
 2 = medio-bassa qualità
 3 = bassa qualità

Il parametro nImageType oltre a definire la qualità indica anche il tipo di formato con il quale verrà salvata l'immagine, ovvero:

0 = imf_ImageFormatBMP
 1 = imf_ImageFormatPNG
 2 = imf_ImageFormatJPEG
 3 = imf_ImageFormatGIF



Questa funzione non è supportata anche su Windows CE, ma in questo caso verrà sempre generata un'immagine in formato ".bmp".



La funzione eseguirà il caricamento dell'immagine soltanto se il sinottico è aperto. Non è infatti sufficiente che l'opzione "Mantieni in memoria" del sinottico sia marcata in quanto anche se il sinottico viene mantenuto in memoria in realtà la sua immagine, una volta chiusa la pagina, viene comunque scaricata, pertanto la funzione GetImage non sarà più in grado di caricare l'immagine. La funzione GetImage ritornerà un variant contenente l'immagine se l'esecuzione è andata a buon fine, mentre ritornerà un variant vuoto se l'immagine non è stata caricata.

Parameter	Description
lpszFileName As String	Nome del file.
nImageType As Integer	Modalità con cui salvare l'immagine.

Result

Boolean

Example:

```
Public Sub Click()
    Dim bResult As Boolean
    Dim lpszFileName As String
    Dim nImageType As Integer
    lpszFileName = "Print.jpg"
    nImageType = 1
    bResult = SaveImageToFile(lpszFileName, nImageType)
```

End Sub

SetAlias, SynopticCmdTarget Function

Syntax SetAlias(_IpszAlias, _IpszValue)

Description Questa funzione imposta il valore definito per l'Alias passato come parametro "IpszAlias". Il nuovo valore sarà quello passato nel parametro "IpszValue ", e potrà essere il nome di una variabile o un valore numerico o stringa.
Se l'Alias non esiste nella Tabella del sinottico verrà aggiunto come nuovo Alias.

Parameter	Description
IpszAlias As String	Nome dell'Alais per il quale impostare il valore. Se non esiste verrà aggiunto come nuovo Alias.
IpszValue As String	Valore con cui impostare l'Alias

Result Boolean

Example:

```
Public Sub Click()  
    Dim objScreen As SynopticCmdTarget  
    Set objScreen = GetSynopticObject  
    Debug.Print objScreen.SetAlias("TsetAlais", "VAR00001")  
    Set objScreen = Nothing  
End Sub
```

SetFocusTo, SynopticCmdTarget Function

Syntax SetFocusTo(_IpszObjectName)

Description Questa funzione consente di fissare il fuoco del sistema su un determinato disegno nel sinottico. Accetta il parametro IpszObjectName contenente il nome del simbolo su cui puntare. Il valore di ritorno True indica che il simbolo ha ricevuto il fuoco altrimenti se il disegno non viene trovato nel sinottico la funzione ritorna con False.

Parameter	Description
IpszObjectName As String	Nome dell'oggetto.

Result Boolean

Example:

```
Public Sub MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)  
    Dim IpszObjectName As String  
    Dim bResult  
  
    ' Ask the symbol (Chiede il simbolo)  
    IpszObjectName = InputBox("Wath symbol do you want set focus ?",  
    "SetFocusTo", "Symbol1", 100, 100)  
    ' Set focus (Imposta il fuoco)  
    bResult = SetFocusTo(IpszObjectName)  
    If Not bResult Then MsgBox("Symbols not found !", , "SetFocusTo")  
End Sub
```

SetRedraw, SynopticCmdTarget Function

Syntax SetRedraw(_bSet)

Description Questa funzione permette di abilitare o disabilitare la gestione grafica della pagina sinottico nella quale viene chiamata. Il parametro bSet identifica uno dei due modi. L'utilizzo è molto delicato in quanto la disabilitazione comporta anche la perdita da parte dell'utente della possibilità di interagire con i vari oggetti o simboli presenti nella pagina. La funzione è utile per disabilitare la visualizzazione di lunghe operazioni grafiche sul progetto e renderle attive solo al momento che sono state ultimate.

Parameter	Description
bSet As Boolean	Valore di abilitazione.

Result None

Example:

```
Public Sub Click()  
    If MsgBox("Do you suspend redraw on the synoptic ?", vbYesNo, "SetRedraw") =  
        vbYes Then  
        SetRedraw(False)  
        Wait 5  
    End If  
    SetRedraw(True)  
End Sub
```

SetSynopsisVisible, SynopticCmdTarget Function

Syntax SetSynopsisVisible(_bVisible)

Description Questa funzione permette di abilitare o disabilitare la visualizzazione delle sinapsi e dei collegamenti (oggetti connettore) presenti nel sinottico. Il parametro bVisible se impostato a false determinerà la non visualizzazione delle sinapsi e collegamenti. Il valore di default all'aperuta di un sinottico è true.

Parameter	Description
bVisible As Boolean	Valore di abilitazione.

Result None

Example:

```
Public Sub SynopticLoading()  
    ' Doesn't show synopsis (Non visualizza le sinapsi)  
    SetSynopsisVisible(False)  
End Sub
```

SetWindowPos, SynopticCmdTarget Function

Syntax SetWindowPos(_nX, _nY, _nWidth, _nHeight, _nShow)

Description Permette di spostare e ridimensionare la finestra contenente il sinottico.

Parameter	Description
nX As Integer	Coordinata X angolo alto sinistra.
nY As Integer	Coordinata Y angolo alto sinistra.
nWidth As Integer	Dimensione larghezza.
nHeight As Integer	Dimensione altezza.
nShow As Integer	Definizione della modalità di visualizzazione: 0= nasconde 1= ripristina 2= riduce a icona 3= massimizza

Result Boolean

Example:

```
Public Sub Click()  
    SetWindowPos(20, 100, 500, 400, 1)  
End Sub
```

ZoomIn, SynopticCmdTarget Function

Syntax ZoomIn()

Description Aumenta il fattore di zoom del sinottico.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim bResult As Boolean  
    bResult = ZoomIn  
    MsgBox "ZoomIn = " & CStr(bResult), vbOkOnly, GetProjectTitle  
End Sub
```

ZoomOut, SynopticCmdTarget Function

Syntax ZoomOut()

Description Diminuisce il fattore di zoom del sinottico.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim bResult As Boolean  
    bResult = ZoomOut  
    MsgBox "ZoomOut = " & CStr(bResult), vbOkOnly, GetProjectTitle  
End Sub
```

ZoomTo, SynopticCmdTarget Function

Syntax ZoomTo()

Description Ritorna al fattore di zoom di default del sinottico.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()  
    Dim bResult As Boolean  
    bResult = ZoomTo  
    MsgBox "ZoomTo = " & CStr(bResult), vbOkOnly, GetProjectTitle  
End Sub
```

Properties

BackColor, SynopticCmdTarget Property

Syntax BackColor = _Long

Description Questa proprietà imposta o restituisce il colore di sfondo del sinottico. Il valore passato o restituito contiene il codice (R,G,B, su ogni byte) del colore di sfondo. Può essere utile utilizzare la funzione RGB per identificare in modo più semplice il colore.

Parameter	Description
None	None

Result Long

Example:

```
'Ambito sinottico
Public Sub Click()
    Dim IRet As Long
    BackColor = RGB(255,0,0)
    IRet = BackColor
    MsgBox "BackColor = " & CStr(IRet), vbOkOnly, "Test BackColor"
    BackColor = RGB(0,255,0)
    IRet = BackColor
    MsgBox "BackColor = " & CStr(IRet), vbOkOnly, "Test BackColor"
    BackColor = RGB(0,0,255)
    IRet = BackColor
    MsgBox "BackColor = " & CStr(IRet), vbOkOnly, "Test BackColor"
End Sub

'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Dim IRet As Long
    Set ObjSyn = GetSynopticObject
    ObjSyn.BackColor= RGB(255,0,0)
    IRet = ObjSyn.BackColor
    MsgBox "BackColor= " & CStr(IRet), vbOkOnly, "Test BackColor"
    ObjSyn.BackColor= RGB(0,255,0)
    IRet = ObjSyn.BackColor
    MsgBox "BackColor = " & CStr(IRet), vbOkOnly, "Test BackColor"
    ObjSyn.BackColor= RGB(0,0,255)
    IRet = ObjSyn.BackColor
    MsgBox "BackColor = " & CStr(IRet), vbOkOnly, "Test BackColor"
    Set ObjSyn = Nothing
End Sub
```

BackColorFileBitmap, SynopticCmdTarget Property

Syntax BackGroundFileBitmap = _String

Description Questa proprietà imposta o restituisce l'immagine di sfondo del sinottico. Il valore passato o restituito contiene l'indirizzo fisico per esteso dell'immagine.

Parameter	Description
None	None

Result String

Example:

```
'Ambito sinottico
Public Sub Click()
    MsgBox "BackColorFileBitmap = " & BackColorFileBitmap ,
    vbOkOnly,GetProjetTitle
    BackColorFileBitmap = "Immagini\sfondo.bmp" 'supponendo di avere
    un'immagine sfondo.bmp nella directory specificata
End Sub
```

```
'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Set ObjSyn = GetSynopticObject
    MsgBox "BackGroundFileBitmap = " & ObjSyn.BackGroundFileBitmap ,
    vbOkOnly,GetProjetTitle
    'supponendo di avere un'immagine sfondo.bmp nella directory specificata
    ObjSyn.BackGroundFileBitmap = "Immagini\sfondo.bmp"
    Set ObjSyn = Nothing
End Sub
```

BackGroundFileBitmapTile, SynopticCmdTarget Property

Syntax BackGroundFileBitmapTile = _Boolean

Description Questa proprietà imposta o restituisce l'abilitazione alla ripetizione del file bitmap di sfondo fino a coprire l'intera superficie del sinottico. Accetta un valore Boolean.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre false')

Parameter	Description
None	None

Result Boolean

Example:

```
'Ambito sinottico
Public Sub Click()
    MsgBox "BackGroundFileBitmapTile = " & BackGroundFileBitmapTile ,
    vbOkOnly, GetProjetTitle
    BackGroundFileBitmapTile = False
End Sub
'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Set ObjSyn = GetSynopticObject
    MsgBox "BackGroundFileBitmapTile = " & ObjSyn.BackGroundFileBitmapTile ,
    vbOkOnly, GetProjetTitle
    ObjSyn.BackGroundFileBitmapTile = False
    Set ObjSyn = Nothing
End Sub
```

CXBackImage, SynopticCmdTarget Property

Syntax CXBackImage = _Integer

Description Questa proprietà imposta o restituisce la larghezza dell'immagine di sfondo del sinottico.

Parameter	Description
None	None

Result Integer

Example:

```
'Ambito sinottico
Public Sub Click()
    Dim IRet As Integer
    IRet = CXBackImage
    MsgBox "CXBackImage = " & CStr(IRet), vbOkOnly, "Test CXBackImage "
    CXBackImage = 100
    IRet = CXBackImage
    MsgBox "CXBackImage = " & CStr(IRet), vbOkOnly, "Test CXBackImage "
End Sub

'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Dim IRet As Long
    Set ObjSyn = GetSynopticObject
    IRet = ObjSyn.CXBackImage
    MsgBox "CXBackImage = " & CStr(IRet), vbOkOnly, "Test CXBackImage "
    ObjSyn.CXBackImage = 100
    IRet = ObjSyn.CXBackImage
    MsgBox "CXBackImage = " & CStr(IRet), vbOkOnly, "Test CXBackImage "
    Set ObjSyn = Nothing
End Sub
```

CYBackImage, SynopticCmdTarget Property

Syntax CYBackImage = _Integer

Description Questa proprietà imposta o restituisce l'altezza dell'immagine di sfondo del sinottico.

Parameter	Description
None	None

Result Integer

Example:

```
'Ambito sinottico
Public Sub Click()
    Dim IRet As Integer
    IRet = CYBackImage
    MsgBox "CYBackImage = " & CStr(IRet), vbOkOnly, "Test CYBackImage "
    CYBackImage = 100
    IRet = CYBackImage
    MsgBox "CYBackImage = " & CStr(IRet), vbOkOnly, "Test CYBackImage "
End Sub

'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Dim IRet As Long
    Set ObjSyn = GetSynopticObject
    IRet = ObjSyn.CYBackImage
    MsgBox "CYBackImage = " & CStr(IRet), vbOkOnly, "Test CYBackImage "
    ObjSyn.CYBackImage = 100
    IRet = ObjSyn.CYBackImage
    MsgBox "CYBackImage = " & CStr(IRet), vbOkOnly, "Test CYBackImage "
    Set ObjSyn = Nothing
End Sub
```


CyclicExecution, SynopticCmdTarget Property

Syntax CyclicExecution = _Boolean

Description L'abilitazione di questa proprietà permette di eseguire in maniera ciclica le eventuali logiche sinapsi quando il Sinottico è attivo.

Parameter	Description
None	None

Result Boolean

Example:

```
'Ambito sinottico
Public Sub Click()
    Dim IRet As Integer
    IRet = CyclicExecution
    MsgBox "CyclicExecution = " & CStr(IRet), vbOkOnly, "Test CyclicExecution "
End Sub

'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Dim IRet As Long
    Set ObjSyn = GetSynopticObject
    IRet = ObjSyn.CyclicExecution
    MsgBox "CyclicExecution = " & CStr(IRet), vbOkOnly, "Test CyclicExecution "
    Set ObjSyn = Nothing
End Sub
```

EnableScrollBars, SynopticCmdTarget Property

Syntax EnableScrollBars = _Boolean

Description L'abilitazione di questa proprietà permette di visualizzare le barre di scorrimento nella finestra che contiene il sinottico. Questa proprietà ha senso se il sinottico ha dimensioni maggiori della finestra che lo contiene e se non viene impostata la proprietà FitInWindow a True.



Questa proprietà non è supportata in Windows CE. (Se impostata restituisce sempre 'false')

Parameter	Description
None	None

Result Boolean

Example:

```
'Ambito sinottico
```

```

Public Sub Click()
    Dim IRet As Integer
    IRet = EnableScrollBars
    MsgBox "EnableScrollBars = " & CStr(IRet), vbOkOnly, "Test EnableScrollBars "
End Sub
'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Dim IRet As Long
    Set ObjSyn = GetSynopticObject
    IRet = ObjSyn.EnableScrollBars
    MsgBox "EnableScrollBars = " & CStr(IRet), vbOkOnly, "Test EnableScrollBars "
    Set ObjSyn = Nothing
End Sub

```

FastTickCounter, SynopticCmdTarget Property

Syntax FastTickCounter = _Long

Description Questa proprietà permette di impostare o visualizzare il numero di loop per tick nelle elaborazioni pendenti di tipo sinapsi o nelle animazioni gestite a polling associate al sinottico. Si consiglia comunque di non modificare i valori di default. In particolare la FastTickCounter riguarda le operazioni più prioritarie.

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
    Dim sRet As Long
    sRet = FastTickCounter
    MsgBox "FastTickCounter = " & sRet, vbOkOnly, GetProjectTitle
End Sub

```

FastTickFrequency, SynopticCmdTarget Property

Syntax FastTickFrequency = _Long

Description Questa proprietà permette di impostare o visualizzare la frequenza di Tick per la gestione delle animazioni gestite a polling e per esempio per l'elaborazione della logica delle sinapsi associate al sinottico. Si consiglia comunque di non modificare i valori di default. In particolare la FastTickFrequency riguarda le operazioni più prioritarie.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
    Dim sRet As Long
    sRet = FastTickFrequency
    MsgBox "FastTickFrequency = " & sRet, vbOkOnly, GetProjectTitle
End Sub
```

FitInWindow, SynopticCmdTarget Property

Syntax FitInWindow = _Boolean

Description L'abilitazione di questa proprietà permette di forzare le dimensioni del sinottico a quelle della finestra che lo contiene.

Parameter	Description
None	None

Result Boolean

Example:

```
'Ambito sinottico
Public Sub Click()
    Dim IRet As Integer
    IRet = FitInWindow
    MsgBox "FitInWindow = " & CStr(IRet), vbOkOnly, "Test FitInWindow"
End Sub

'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Dim IRet As Long
    Set ObjSyn = GetSynopticObject
    IRet = ObjSyn.FitInWindow
    MsgBox "FitInWindow = " & CStr(IRet), vbOkOnly, "Test FitInWindow "
    Set ObjSyn = Nothing
End Sub
```

FrameTitle, SynopticCmdTarget Property

Syntax FrameTitle = _String

Description Questa proprietà imposta o restituisce il nome del frame che ha caricato il sinottico. Questa proprietà non ha quindi significato se usata da WebClient, e in questo caso restituisce una stringa vuota."

Parameter	Description
None	None

Result String

Example:

```
'Ambito sinottico
Public Sub Click()
    MsgBox "FrameTitle = " & FrameTitle , vbOkOnly,GetProjetTitle
```

```

End Sub
'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Set ObjSyn = GetSynopticObject
    MsgBox "FrameTitle = " & ObjSyn.FrameTitle , vbOkOnly,GetProjetTitle
    Set ObjSyn = Nothing
End Sub

```

GlobalContainerName, SynopticCmdTarget Property

Syntax GlobalContainerName = _Boolean

Description L'abilitazione di questa proprietà permette di rendere pubblico sinottico per ogni suo oggetto figlio. Ciò significa che gli oggetti ereditano le funzioni specifiche del sinottico.

Parameter	Description
None	None

Result Boolean

Example:

```

'Ambito sinottico
Public Sub Click()
    Dim IRet As Integer
    IRet = GlobalContainerName
    MsgBox "GlobalContainerName = " & CStr(IRet), vbOkOnly, "Test
    GlobalContainerName "
End Sub
'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Dim IRet As Long
    Set ObjSyn = GetSynopticObject
    IRet = ObjSyn.GlobalContainerName
    MsgBox "GlobalContainerName = " & CStr(IRet), vbOkOnly, "Test
    GlobalContainerName "
    Set ObjSyn = Nothing
End Sub

```

GradientColor, SynopticCmdTarget Property

Syntax GradientColor = _Long

Description Questa proprietà imposta o restituisce il colore di sfumatura dell'oggetto. Accetta un parametro Long contenente il codice (R,G,B, su ogni byte) per il colore dell'ombra. Può risultare utile l'utilizzo della funzione RGB di Movicon.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()  
    GradientColor = RGB(255,255,255)  
    sRet = GradientColor  
    MsgBox "GradientColor = " & sRet, vbOkOnly, GetProjectTitle 'Return->  
    16777215  
End Sub
```

GradientFill, SynopticCmdTarget Property

Syntax GradientFill = _Integer

Description Questa proprietà imposta o restituisce, mediante un codice numerico, la direzione di sfumatura del colore di sfondo del sinottico. Tale proprietà può assumere i seguenti valori:

0 = nessuno,
1 = da destra a sinistra,
2 = dal centro verso le estremità in orizzontale
3 = da sinistra a destra
4 = dal basso verso l'alto
5 = dal centro verso le estremità in verticale
6 = dall'alto verso il basso
7 = dal centro verso le estremità
8 = dall'angolo in alto a sinistra in diagonale
9 = dall'angolo in alto a destra in diagonale
10 = dall'angolo in basso a destra in diagonale
11 = dall'angolo in basso a sinistra in diagonale

Parameter	Description
None	None

Result Integer

Example:

```
'Ambito sinottico  
Public Sub Click()  
    For i = 0 To 11 Step 1  
        GradientFill = i  
        sRet = GradientFill  
        MsgBox "GradientFill = " & sRet, vbOkOnly, GetProjectTitle  
    Next i  
End Sub  
  
'Ambito oggetto nel sinottico  
Public Sub Click()  
    Dim ObjSyn As SynopticCmdTarget  
    Set ObjSyn = GetSynopticObject  
    For i = 0 To 11 Step 1  
        ObjSyn.GradientFill = i  
        sRet = ObjSyn.GradientFill  
        MsgBox "GradientFill = " & sRet, vbOkOnly, GetProjectTitle  
    Next i  
    Set ObjSyn = Nothing  
End Sub
```

KeepAspectRatio, SynopticCmdTarget Property

Syntax KeepAspectRatio = _Boolean

Description Permette di leggere o impostare la proprietà del Sinottico "Mantieni Proporzionale (Web Client HTML5)" che permette il ridimensionamento proporzionale delle pagine visualizzate da la sessione Web Client HTML5. Quando è attiva permette di mantenere proporzionale il rapporto di visualizzazione del sinottico indipendentemente dalla proprietà "Fit In Windows" impostata sul sinottico stesso.

Parameter	Description
None	None

Result boolean

Example:

```
'Nell'Ambiente Sinottico
Public Sub Click()
Dim IRet As Integer
    IRet = KeepAspectRatio
    MsgBox "KeepAspectRatio = " & CStr(IRet), vbOkOnly, "Test KeepAspectRatio "
End Sub
'in un Oggetto Grafico del Sinottico
Public Sub Click()
Dim ObjSyn As SynopticCmdTarget
    Set ObjSyn = GetSynopticObject
    ObjSyn.KeepAspectRatio = Not ObjSyn.KeepAspectRatio
    Debug.Print "KeepAspectRatio = " & CStr(CBool(ObjSyn.KeepAspectRatio))
    Set ObjSyn = Nothing
End Sub
```

LayerVariable, SynopticCmdTarget Property

Syntax LayerVariable = _String

Description Questa proprietà restituisce il nome della variabile del Real Time DB di Movicon che determinerà il numero del livello di visualizzazione (Layer) per i controlli o i simboli del sinottico.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sRet As String
    sRet = LayerVariable
    MsgBox "LayerVariable = " & sRet, vbOkOnly, GetProjectTitle
End Sub
```

MaxInstances, SynopticCmdTarget Property

Syntax MaxInstances = _Integer

Description Questa proprietà consente di leggere il valore del numero massimo di istanze del sinottico.

Nota: l'apertura di ogni istanze del sinottico causa la riletture del file XML di descrizione del sinottico e quindi MaxInstances è impostato al valore in esso definito.

Parameter	Description
None	None

Result Integer

Example:

```
Dim objScreen As SynopticCmdTarget
Set objScreen = GetSynopticObject
Debug.Print objScreen.MaxInstances()
Set objScreen = Nothing
End Sub
```

NonDestroyable, SynopticCmdTarget Property

Syntax NonDestroyable = _Integer

Description Questa opzione imposta o restituisce la proprietà per cui il sinottico non viene mai definitivamente chiuso. Ne consegue che, anche se non visualizzato, il sinottico continuerà ad occupare memoria e risorse di sistema, con il vantaggio di una rapida visualizzazione a video.

Parameter	Description
None	None

Result Integer

Example:

```
'Ambito sinottico
Public Sub Click()
    Dim IRet As Integer
    IRet = NonDestroyable
    MsgBox "NonDestroyable = " & CStr(IRet), vbOkOnly, "Test NonDestroyable "
End Sub

'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Dim IRet As Long
    Set ObjSyn = GetSynopticObject
    IRet = ObjSyn.NonDestroyable
    MsgBox "NonDestroyable = " & CStr(IRet), vbOkOnly, "Test NonDestroyable "
    Set ObjSyn = Nothing
End Sub
```

NumColors, SynopticCmdTarget Property

Syntax NumColors = _Integer

Description Questa proprietà consente di impostare o leggere il numero di colori utilizzati per gestire la grafica nel sinottico. I valori ammessi sono

0 = Default
1 = Bianco e Nero

Attualmente impostare la proprietà con un nuovo valore non ha molto senso, perchè manca un metodo per aggiornare la pagina con i nuovi valori.

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()  
    If GetSynopticObject.NumColors <> 0 Then  
        GetSynopticObject.NumColors = 0  
    Else  
        GetSynopticObject.NumColors = 1  
    End If  
End Sub
```

ParameterFileName, SynopticCmdTarget Property

Syntax ParameterFileName = _String

Description Questa proprietà restituisce una stringa contenente il nome del file per la parametrizzazione del sinottico. Il file di testo di parametrizzazione deve essere salvato in formato **UNICODE**. Un sinottico puo' essere aperto in modo parametrizzato mediante gli opportuni comandi della Movicon Interface. Per parametrizzazione di un sinottico si intende la possibilità di sostituire, durante la fase di runtime, le variabili utilizzate nelle proprietà di esecuzione dei disegni con altre specificate appunto nel file di testo in oggetto. La struttura che tale file deve rispettare e': <variabile originale>,<nuova variabile>. La parametrizzazione non influisce sugli oggetti del sinottico o sul codice basic contenuto nei disegni o simboli. Un esempio di file di parametrizzazione puo' essere il seguente:

```
VAR00001,NUOVA01  
VAR00002,NUOVA02  
VAR00003,NUOVA03
```

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()  
    Dim sRet As String  
    sRet = ParameterFileName
```



```

MsgBox "ParameterFileName = " & sRet, vbOkOnly, GetProjectTitle
End Sub

```

ScrollPositionX, SynopticCmdTarget Property

Syntax ScrollPositionX = _Boolean

Description Questa proprietà imposta o restituisce la posizione X di scroll di un sinottico. Questa funzionalità ha effetto soltanto se sono abilitate le barre di scorrimento del sinottico.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre zero)

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub Click()
    GetSynopticObject.ScrollPositionX = GetSynopticObject.ScrollPositionX + 5
End Sub

```

ScrollPositionY, SynopticCmdTarget Property

Syntax ScrollPositionY = _Boolean

Description Questa proprietà imposta o restituisce la posizione Y di scroll di un sinottico. Questa funzionalità ha effetto soltanto se sono abilitate le barre di scorrimento del sinottico.



Questa proprietà non è supportata in Windows CE.(se utilizzata restituisce sempre zero)

Parameter	Description
None	None

Result Boolean

Example:

```

Public Sub Click()
    GetSynopticObject.ScrollPositionY = GetSynopticObject.ScrollPositionY + 5
End Sub

```

SeparateThread, SynopticCmdTarget Property

Syntax SeparateThread = _Boolean

Description L'abilitazione di questa proprietà permette di eseguire in un thread separato il sinottico, indipendentemente dall'esecuzione degli altri thread del progetto. E' utile questa funzione in caso di sinottici contenenti significative elaborazioni logiche o sinapsi. In tal caso le elaborazioni logiche saranno eseguite in un thread separato senza penalizzare l'interfaccia grafica, richiedendo tuttavia un maggiore impiego di risorse di memoria.

Parameter	Description
None	None

Result Boolean

Example:

```
'Ambito sinottico
Public Sub Click()
    Dim IRet As Integer
    IRet = SeparateThread
    MsgBox "SeparateThread = " & CStr(IRet), vbOkOnly, "Test SeparateThread "
End Sub

'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Dim IRet As Long
    Set ObjSyn = GetSynopticObject
    IRet = ObjSyn.SeparateThread
    MsgBox "SeparateThread = " & CStr(IRet), vbOkOnly, "Test SeparateThread "
    Set ObjSyn = Nothing
End Sub
```

ShowOnMDITabsFlag, SynopticCmdTarget Property

Syntax ShowOnMDITabsFlag = _Boolean

Description L'abilitazione di questa proprietà permette di visualizzare i Tab MDI dei Sinottici anche in Runtime.



Questa proprietà non è supportata in Windows CE. (Se impostata restituisce sempre 'false')

Parameter	Description
None	None

Result Boolean

Example:

```
'Ambito sinottico
Public Sub Click()
```

```

Dim IRet As Integer
IRet = ShowOnMDITabsFlag
MsgBox "ShowOnMDITabsFlag = " & CStr(IRet), vbOkOnly, "Test
ShowOnMDITabsFlag "
End Sub
'Ambito oggetto nel sinottico
Public Sub Click()
Dim ObjSyn As SynopticCmdTarget
Dim IRet As Long
Set ObjSyn = GetSynopticObject
IRet = ObjSyn.ShowOnMDITabsFlag
MsgBox "ShowOnMDITabsFlag = " & CStr(IRet), vbOkOnly, "Test
ShowOnMDITabsFlag "
Set ObjSyn = Nothing
End Sub

```

SlowTickCounter, SynopticCmdTarget Property

Syntax SlowTickCounter = _Long

Description Questa proprietà permette di impostare o visualizzare il numero di loop per tick nelle elaborazioni pendenti di tipo sinapsi o nelle animazioni gestite a polling associate al sinottico. Si consiglia comunque di non modificare i valori di default. In particolare la SlowTickCounter riguarda le operazioni meno prioritarie.

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
Dim sRet As Long
sRet = SlowTickCounter
MsgBox "SlowTickCounter = " & sRet, vbOkOnly, GetProjectTitle
End Sub

```

SlowTickFrequency, SynopticCmdTarget Property

Syntax SlowTickFrequency = _Long

Description Questa proprietà permette di impostare o visualizzare la frequenza di Tick per la gestione delle animazioni gestite a polling e per esempio per l'elaborazione della logica delle sinapsi associate al sinottico. Si consiglia comunque di non modificare i valori di default. In particolare la SlowTickFrequency riguarda le operazioni meno prioritarie.

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
    Dim sRet As Long
    sRet = SlowTickFrequency
    MsgBox "SlowTickFrequency = " & sRet, vbOkOnly, GetProjectTitle
End Sub
```

SynapsisExecution, SynopticCmdTarget Property

Syntax SynapsisExecution = _Boolean

Description Questa proprietà imposta o restituisce lo stato di esecuzione delle sinapsi all'interno del sinottico. Impostando la proprietà a True è eseguita la logica di tipo sinapsi contenuta nel sinottico. Al termine la proprietà ritorna al valore False.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub SynopticLoading()
    ' Run the synapsis (Esegue le sinapsi)
    SynapsisExecution = True
End Sub
```

SynopticHeight, SynopticCmdTarget Property

Syntax SynopticHeight = _Integer

Description Questa proprietà imposta o restituisce l'altezza del sinottico.

Parameter	Description
None	None

Result Integer

Example:

```
'Ambito sinottico
Public Sub Click()
    MsgBox "SynopticHeight = " & SynopticHeight , vbOkOnly, GetProjectTitle
End Sub
'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Set ObjSyn = GetSynopticObject
    MsgBox "SynopticHeight = " & ObjSyn.SynopticHeight , vbOkOnly,
    GetProjectTitle
    Set ObjSyn = Nothing
```

End Sub

SynopticID, SynopticCmdTarget Property

Syntax SynopticID = _Long

Description Questa proprietà imposta o restituisce l'ID del sinottico.

Parameter	Description
None	None

Result Long

Example:

```
'Ambito sinottico
Public Sub Click()
    MsgBox "FrameTitle = " & SynopticID , vbOkOnly, GetProjetTitle
End Sub
'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Set ObjSyn = GetSynopticObject
    MsgBox "FrameTitle = " & ObjSyn.SynopticID , vbOkOnly, GetProjetTitle
    Set ObjSyn = Nothing
End Sub
```

SynopticPublicSource, SynopticCmdTarget Property

Syntax SynopticPublicSource = _String

Description Questa proprietà restituisce il nome del Sinottico di riferimento per i Simboli Pubblici. Se nel progetto sono stati inseriti alcuni simboli con lo stesso "Nome Pubblico", e uno di questi è presente nel Sinottico "Contenitore Simboli Pubblici", all'avvio del Runtime tutti i simboli pubblici verranno modificati con le proprietà del simbolo di riferimento, ovvero quello contenuto nel Sinottico specificato nella proprietà "Contenitore Simboli Pubblici".

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sRet As String
    sRet = SynopticPublicSource
    MsgBox "SynopticPublicSource = " & sRet, vbOkOnly, GetProjectTitle
End Sub
```

SynopticWidth, SynopticCmdTarget Property

Syntax SynopticWidth = _Integer

Description Questa proprietà imposta o restituisce la larghezza del sinottico.

Parameter	Description
None	None

Result Integer

Example:

```
'Ambito sinottico
Public Sub Click()
    MsgBox "SynopticWidth = " & SynopticWidth, vbOkOnly, GetProjetTitle
End Sub
'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Set ObjSyn = GetSynopticObject
    MsgBox "SynopticWidth = " & ObjSyn.SynopticWidth, vbOkOnly, GetProjetTitle
    Set ObjSyn = Nothing
End Sub
```

UseAntialiasing, SynopticCmdTarget Property

Syntax UseAntialiasing= _Boolean

Description Questa proprietà attiva o disattiva la modalità grafica Anti Alaising sui disegni appartenenti al sinottico.



Attenzione: su sistemi Windos CE non ha alcun effetto.

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim oSyn As SynopticCmdTarget
    Set oSyn = GetSynopticObject
    oSyn.UseAntialiasing = Not(oSyn.UseAntialiasing)
End Sub
```

UseIMDB,DLRCmdTarget property

Syntax UseIMDB = _Boolean

Description Questa proprietà di sola lettura consente di sapere se il data logger è impostato per registrare i valori attraverso il motore InMemoryDataBase.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
Dim obj As DLRCmdTarget

Set obj = GetDataLoggerRecipe("DataLogger")
If Not obj Is Nothing Then
MsgBox "UseIMDB = " & obj.UseIMDB
End If
End Sub
```

XBackImage, SynopticCmdTarget Property

Syntax XBackImage = _Integer

Description Questa proprietà imposta o restituisce il valore relativo alla coordinata sull'asse x dell'immagine all'interno del sinottico.

Parameter	Description
None	None

Result Integer

Example:

```
'Ambito sinottico
Public Sub Click()
    Dim IRet As Integer
    IRet = XBackImage
    MsgBox "XBackImage = " & CStr(IRet), vbOkOnly, "Test XBackImage "
    XBackImage = 100
    IRet = XBackImage
    MsgBox "XBackImage = " & CStr(IRet), vbOkOnly, "Test XBackImage "
End Sub

'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Dim IRet As Long
    Set ObjSyn = GetSynopticObject
    IRet = ObjSyn.XBackImage
    MsgBox "XBackImage = " & CStr(IRet), vbOkOnly, "Test XBackImage "
    ObjSyn.XBackImage = 100
    IRet = ObjSyn.XBackImage
    MsgBox "XBackImage = " & CStr(IRet), vbOkOnly, "Test XBackImage "
    Set ObjSyn = Nothing
End Sub
```

YBackImage, SynopticCmdTarget Property

Syntax YBackImage = _Integer

Description Questa proprietà imposta o restituisce il valore relativo alla coordinata sull'asse y dell'immagine all'interno del sinottico.

Parameter	Description
None	None

Result Integer

Example:

```
'Ambito sinottico
Public Sub Click()
    Dim IRet As Integer
    IRet = YBackImage
    MsgBox "YBackImage = " & CStr(IRet), vbOkOnly, "Test YBackImage "
    YBackImage = 100
    IRet = YBackImage
    MsgBox "YBackImage = " & CStr(IRet), vbOkOnly, "Test YBackImage "
End Sub

'Ambito oggetto nel sinottico
Public Sub Click()
    Dim ObjSyn As SynopticCmdTarget
    Dim IRet As Long
    Set ObjSyn = GetSynopticObject
    IRet = ObjSyn.YBackImage
    MsgBox "YBackImage = " & CStr(IRet), vbOkOnly, "Test YBackImage "
    ObjSyn.YBackImage = 100
    IRet = ObjSyn.YBackImage
    MsgBox "YBackImage = " & CStr(IRet), vbOkOnly, "Test YBackImage "
    Set ObjSyn = Nothing
End Sub
```

ZoomFactorX, SynopticCmdTarget Property

Syntax ZoomFactorX = _Double

Description Questa proprietà o imposta restituisce il valore di zoom lungo l'asse x del sinottico all'interno della finestra.

Parameter	Description
None	None

Result Double

Example:

```
Public Sub Click()
    Dim sRet As Double
    sRet = ZoomFactorX
    MsgBox "ZoomFactorX = " & sRet, vbOkOnly, GetProjectTitle
    ZoomFactorX = 0.5
```



```

sRet = ZoomFactorX
MsgBox "ZoomFactorX = " & sRet, vbOkOnly, GetProjectTitle
End Sub

```

ZoomFactorY, SynopticCmdTarget Property

Syntax ZoomFactorY = _Double

Description Questa proprietà restituisce o imposta il valore di zoom lungo l'asse Y del sinottico all'interno della finestra.

Parameter	Description
None	None

Result Double

Example:

```

Public Sub Click()
    Dim sRet As Double
    sRet = ZoomFactorY
    MsgBox "ZoomFactorY = " & sRet, vbOkOnly, GetProjectTitle
    ZoomFactorY = 0.5
    sRet = ZoomFactorY
    MsgBox "ZoomFactorY = " & sRet, vbOkOnly, GetProjectTitle
End Sub

```

1.17.7. TraceDBWndCmdTarget

Events

OnFilter, TraceDBWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di applicare un filtro per l'estrazione dei dati dal file di trace.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

OnPrint, TraceDBWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di effettuare la stampa dei dati caricati nella finestra di visualizzazione.



Questo evento non è supportato in Windows CE.

Parameter	Description
bRet As Boolean	Abilitazione all'invio della stampa.

OnRefresh, TraceDBWndCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di effettuare un rinfresco dei dati caricati nella finestra di visualizzazione.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

Functions

EditCopy, TraceDBWndCmdTarget Function

Syntax EditCopy()

Description Questa proprietà esegue una copia negli appunti del contenuto della riga selezionata.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWnd =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd.EditCopy
    End If
    Set TraceDBWnd = Nothing
End Sub
```

EditLayout, TraceDBWndCmdTarget Function

Syntax EditLayout()

Description Questa funzione apre la finestra di configurazione dei campi da visualizzare nella finestra di visualizzazione del Trace.



La funzione verrà eseguita soltanto se nell'oggetto Finestra è stata abilitata la proprietà "Mostra Finestra di Controllo". In caso contrario la "Finestra Scelta Campo" non verrà aperta e la funzione ritornerà il valore "False"

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd.EditLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

LoadExtSettings, TraceDBWndCmdTarget Function

Syntax LoadExtSettings

Description Questa funzione permette di caricare la configurazione dell'oggetto dal file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Boolean

Example:

```
Public Sub Click()
    Dim objSymbol As TraceDBWndCmdTarget
    Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
    If objSymbol Is Nothing Then Exit Sub
    objSymbol.ExtSettingsFile = "test.sxml"
    objSymbol.LoadExtSettings
    Set objSymbol = Nothing
End Sub
```

RecalcLayout, TraceDBWndCmdTarget Function

Syntax RecalcLayout()

Description Questa funzione esegue un'aggiornamento grafico dell'oggetto. E' necessario eseguire tale funzione dopo che e' stata modificata una proprietà il cui impatto grafico sia particolarmente consistente, come ad esempio la modifica della dimensione di una colonna.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd .AutoLayout = Not TraceDBWnd .AutoLayout
        TraceDBWnd .RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

Refresh, TraceDBWndCmdTarget Function

Syntax Refresh()

Description Questa funzione esegue un'aggiornamento dei dati nell'oggetto, utile ad esempio quando viene modificata la query di estrazione dei dati dal database del Log.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd.Query = "SELECT * FROM VAR00001 ORDER BY
        VAR00001.TimeCol DESC"
        TraceDBWnd.Refresh
    End If
    Set TraceDBWnd = Nothing
End Sub
```

Properties

AutoLayout, TraceDBWndCmdTarget Property

Syntax AutoLayout = _Boolean

Description Abilitando questa proprietà il layout verrà impostato in modo automatico. Questo significa che le colonne verranno automaticamente ridimensionate in modo che risultino visibili tutte quante nell'area della **Finestra DB Trace**. Disabilitando invece questa proprietà all'apertura della finestra le colonne avranno la dimensione definita in programmazione, con la possibilità che le ultime sulla destra non siano visibili se non attivando la barra di scorrimento orizzontale.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd.AutoLayout = Not TraceDBWnd.AutoLayout
        TraceDBWnd.RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

ButtonPos, TraceDBWndCmdTarget Property

Syntax ButtonPos = _Integer

Description Questa impostazione restituisce la posizione in cui dovranno apparire i pulsanti nella finestra di Trace.

Le possibilità sono:
 0 = sinistra
 1 = sopra
 2 = destra
 3 = sotto

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim objWnd As TraceDBWndCmdTarget
    Set objWnd = GetSynopticObject.GetSubObject("TraceLog").GetObjectInterface
    If Not objWnd Is Nothing Then
        MsgBox "objWnd 's ButtonPos is " & objWnd.ButtonPos, vbInformation, GetProjectTitle
        objWnd.ButtonPos = 2
        objWnd.RecalcLayout
    Else
        MsgBox "objWnd is nothing", vbInformation, GetProjectTitle
    End If
End Sub
```

ButtonSize, TraceDBWndCmdTarget Property

Syntax ButtonSize = _Integer

Description Questa impostazione restituisce la dimensione dei pulsanti che verranno poi visualizzati nella finestra di Trace.

Le possibilità sono:

0 = piccolo

1 = medio

2 = grande

Parameter	Description
None	None

Result Integer

Example:

Option Explicit

Public Sub Click()

Dim objWnd As TraceDBWndCmdTarget

Set objWnd = GetSynopticObject.GetSubObject("TraceLog").GetObjectInterface

If Not objWnd Is Nothing Then

MsgBox "objWnd 's ButtonSize is " & objWnd

.**ButtonSize**,vbInformation,GetProjectTitle

objWnd.**ButtonSize**= 2

objWnd.RecalcLayout

Else

MsgBox "objWnd is nothing",vbInformation,GetProjectTitle

End If

End Sub

Clickable, TraceDBWndCmdTarget Property

Syntax Clickable = _Boolean

Description Questa proprietà permette di definire se l'operatore potrà interagire con la finestra di visualizzazione del Trace. Disabilitando questa proprietà il controllo non potrà infatti essere gestito nè con il mouse nè con la tastiera.

Parameter	Description
None	None

Result Boolean

Example:

Option Explicit

Public Sub Click()

Dim TraceDBWnd As TraceDBWndCmdTarget

Set TraceDBWnd

GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface

If Not TraceDBWnd Is Nothing Then

TraceDBWnd.**Clickable** = Not TraceDBWnd.**Clickable**

TraceDBWnd.RecalcLayout

```

        End If
        Set TraceDBWnd = Nothing
    End Sub

```

ExtSettingsFile, TraceDBWndCmdTarget Property

Syntax ExtSettingsFile = _String

Description Questa proprietà imposta o restituisce il file di configurazione esterna per l'oggetto referenziato. Il file può essere specificato anche in fase di design nella proprietà "File di Configurazione" dell'oggetto. L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Long

Example:

```

Public Sub Click()
    Dim objSymbol As TraceDBWndCmdTarget
    Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
    If objSymbol Is Nothing Then Exit Sub
    objSymbol.ExtSettingsFile = "test.sxml"
    objSymbol.SaveExtSettings
    Set objSymbol = Nothing
End Sub

```

FileReport, TraceDBWndCmdTarget Property

Syntax FileReport = _String

Description Questa proprietà imposta o restituisce il nome del file di report da utilizzare per la stampa dei dati visualizzati nella finestra di Trace delle Variabili. Il file deve essere creato con Report Designer o Crystal Report© (.rpt). Se il campo viene lasciato vuoto Movicon utilizza il file di report di default creato da Progea in formato Report Designer.



Questa proprietà non è supportata in Windows CE. (Se impostata restituisce sempre una stringa vuota)

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim objWnd As TraceDBWndCmdTarget
    Set objWnd = GetSynopticObject.GetSubObject("TraceLog").GetObjectInterface
    If Not objWnd Is Nothing Then

```

```

        MsgBox "objWnd 'sFileReport is " & objWnd.FileReport
        ,vbInformation,GetProjectTitle
        objWnd.FileReport = "C:\Report1.rpt"
        objWnd.RecalcLayout
    Else
        MsgBox "objWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub

```

FilterBtnText, TraceDBWndCmdTarget Property

Syntax FilterBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando di filtro dei dati nella finestra di Trace. Se non specificato nulla, Movicon utilizza il testo di default.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim objWnd As TraceDBWndCmdTarget
    Set objWnd = GetSynopticObject.GetSubObject("TraceLog").GetObjectInterface
    If Not objWnd Is Nothing Then
        MsgBox "objWnd 's FilterBtnText is " &
        objWnd.FilterBtnText,vbInformation,GetProjectTitle
        objWnd.FilterBtnText = "Filter options"
        objWnd.RecalcLayout
    Else
        MsgBox "objWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub

```

FilterFromDate, TraceDBWndCmdTarget Property

Syntax FilterFromDate = _Date

Description Questa proprietà imposta o restituisce il filtro 'Data iniziale' per la visualizzazione dei messaggi nella finestra di Trace di Movicon.

Parameter	Description
None	None

Result Date

Example:

Option Explicit


```

Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        MsgBox "TraceDBWnd's FilterFromDate is " & TraceDBWnd
        .FilterFromDate, vbInformation, GetProjectTitle
        TraceDBWnd.FilterFromDate = Now()
        TraceDBWnd.Refresh
    Else
        MsgBox "TraceDBWnd is nothing", vbInformation, GetProjectTitle
    End If
    Set TraceDBWnd = Nothing
End Sub

```

FilterToDate, TraceDBWndCmdTarget Property

Syntax FilterToDate = _Date

Description Questa proprietà imposta o restituisce il filtro 'Data Iniziale' per la visualizzazione dei messaggi nella finestra di Log Storico di Movicon.

Parameter	Description
None	None

Result Date

Example:

```

Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        MsgBox "TraceDBWnd's FilterToDate is " & TraceDBWnd
        .FilterToDate, vbInformation, GetProjectTitle
        TraceDBWnd.FilterToDate = Now()
        TraceDBWnd.Refresh
    Else
        MsgBox "TraceDBWnd is nothing", vbInformation, GetProjectTitle
    End If
    Set TraceDBWnd = Nothing
End Sub

```

FilterUser, TraceDBWndCmdTarget Property

Syntax FilterUser = _String

Description Questa proprietà imposta o restituisce il filtro 'Utente' per la visualizzazione dei messaggi nella finestra di Trace di Movicon.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        MsgBox "TraceDBWnd's FilterUser is " &
            TraceDBWnd.FilterUser, vbInformation, GetProjectTitle
        TraceDBWnd.FilterUser = "User00001"
        TraceDBWnd.Refresh
    Else
        MsgBox "TraceDBWnd is nothing", vbInformation, GetProjectTitle
    End If
    Set TraceDBWnd = Nothing
End Sub
```

GraphicButtons, TraceDBWndCmdTarget Property

Syntax GraphicButtons = _Boolean

Description Abilitando questa proprietà i pulsanti della Finestra DB Trace verranno disegnati usando un'icona invece del testo. Il testo opportuno verrà invece visualizzato come tooltip posizionando il mouse sopra ad un pulsante.



Il tooltip non viene gestito nella versione Windows CE.



Questa proprietà non è gestita dall'oggetto "Banner Allarmi".

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Click()
    GraphicButtons = True
    RecalcLayout
End Sub
```

IncludeMilliseconds, TraceDBWndCmdTarget Property

Syntax IncludeMilliseconds = _Boolean

Description Questa proprietà permette di definire se visualizzare o meno i millisecondi nella colonna tempo della finestra.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd.IncludeMilliseconds = Not
        TraceDBWnd.IncludeMilliseconds
        TraceDBWnd.Refresh
    End If
    Set TraceDBWnd = Nothing
End Sub
```

MaxCount, TraceDBWndCmdTarget Property

Syntax MaxCount = _Integer

Description Questa proprietà permette di definire il numero massimo di righe che la **Finestra DB Trace** potrà visualizzare.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd.MaxCount = 150
        TraceDBWnd.Refresh
    End If
    Set TraceDBWnd = Nothing
End Sub
```

NetworkBackupServerName, TraceDBWndCmdTarget Property

Syntax NetworkBackupServerName = _String

Description Questa proprietà imposta o restituisce il nome dell'eventuale Server di Networking di Backup usato per prelevare i dati da visualizzare nella Finestra TraceDB quando il Server Primario, ovvero quello impostato nella proprietà "NetowrkServerName", è in timeout.

Parameter	Description
None	None

Result String

Example:

```
Dim TraceDBWnd As TraceDBWndCmdTarget

Public Sub Click()
    Debug.Print objTraceDBWnd.NetworkBackupServerName
End Sub
Public Sub SymbolLoading()
    Set objTraceDBWnd=
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
End Sub
```

NetworkServerName, TraceDBWndCmdTarget Property

Syntax NetworkServerName = _String

Description Questa proprietà restituisce il nome dell'eventuale Server di Networking dal quale dovranno essere prelevati i dati da visualizzare [nella Finestra TraceDB](#).

Parameter	Description
None	None

Result String

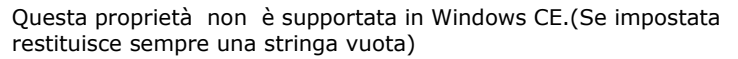
Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd.NetworkServerName = "PERSONAL11"
        TraceDBWnd.Refresh
    End If
    Set TraceDBWnd = Nothing
End Sub
```

PrintBtnText, TraceDBWndCmdTarget Property

Syntax PrintBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando di stampa dei dati nella finestra di Trace. Se non specificato nulla, Movicon utilizza il testo di default.



Result	String
--------	--------

```
Option Explicit
Public Sub Click()
    Dim objWnd As TraceDBWndCmdTarget
    Set objWnd = GetSynopticObject.GetSubObject("TraceLog").GetObjectInterface
    If Not objWnd Is Nothing Then
        MsgBox "objWnd 's PrintBtnText is " & objWnd.PrintBtnText
        ,vbInformation,GetProjectTitle
        objWnd.PrintBtnText = "Print data"
        objWnd.RecalcLayout
    Else
        MsgBox "objWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub
```

Syntax Project = _String

In questa proprietà va inserito esclusivamente il nome di un eventuale progetto figlio del progetto corrente.

Result	String
--------	--------

927

Query, TraceDBWndCmdTarget Property

Syntax Query = _String

Description Questa proprietà permette di definire una query di selezione in linguaggio SQL per l'estrazione dei dati contenuti nel database del TraceDB. Questa query verrà eseguita per default ad ogni rinfresco dei dati nella finestra, sia esso eseguito automaticamente che su comando da operatore.

Per ogni variabile sulla quale viene abilitato il "Trace" viene creata una tabella nel database.

Le colonne appartenenti alla struttura delle tabelle sono:

ActionCol	Action
ValueCol	Value
TimeCol	Event Time
UserCol	User
BeforeCol	Before
AfterCol	After
QualityCol	Quality

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd .Query = "SELECT * FROM VAR00001 ORDER BY
        VAR00001.TimeCol DESC"
        TraceDBWnd.Refresh
    End If
    Set TraceDBWnd = Nothing
End Sub
```

RefreshBtnText, TraceDBWndCmdTarget Property

Syntax RefreshBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando di rinfresco dei dati nella finestra di Trace. Se non specificato nulla, Movicon utilizza il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim objWnd As TraceDBWndCmdTarget
    Set objWnd = GetSynopticObject.GetSubObject("TraceLog").GetObjectInterface
    If Not objWnd Is Nothing Then
        MsgBox "objWnd 's RefreshBtnText is " &
            objWnd.RefreshBtnText, vbInformation, GetProjectTitle
        objWnd.RefreshBtnText = "Refresh all"
        objWnd.RecalcLayout
    Else
        MsgBox "objWnd is nothing", vbInformation, GetProjectTitle
    End If
End Sub
```

SaveExtSettings, TraceDBWndCmdTarget Function

Syntax SaveExtSettings

Description Questa funzione permette di salvare la configurazione dell'oggetto nel file di configurazione esterna relativo. Il file può essere specificato in fase di design nella proprietà "File di Configurazione", oppure mediante la proprietà dell'interfaccia "ExtSettingsFile". L'estensione prevista per il file è ".SXML".

Parameter	Description
None	None

Result Long

Example:

```
Public Sub Click()
    Dim objSymbol As TraceDBWndCmdTarget
    Set objSymbol = GetSynopticObject.GetSubObject("TestObject").GetObjectInterface
    If objSymbol Is Nothing Then Exit Sub
    objSymbol.ExtSettingsFile = "test.sxml"
    objSymbol.SaveExtSettings
    Set objSymbol = Nothing
End Sub
```

ShowFilterBtn, TraceDBWndCmdTarget Property

Syntax ShowFilterBtn = _Boolean

Description Questa proprietà permette di visualizzare il pulsante di comando per l'operazione di filtro nella finestra di Trace.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Public Sub Click()
    Dim objTraceWnd As HisLogWndCmdTarget
    Set objTraceWnd =
    GetSynopticObject.GetSubObject("TraceLog").GetObjectInterface
    If Not objTraceWnd Is Nothing Then
        objTraceWnd .ShowFilterBtn = Not objTraceWnd .ShowFilterBtn
        objTraceWnd .RecalcLayout
    End If
    Set objTraceWnd = Nothing
End Sub

```

ShowPrintBtn, TraceDBWndCmdTarget Property

Syntax ShowPrintBtn = _Boolean

Description Questa proprietà permette di visualizzare il pulsante di comando per l'operazione di stampa dei dati visualizzati finestra di Trace.
La stampa viene eseguita utilizzando il file di report che dovrà essere specificato nella proprietà "File di Report". Movicon "passa" al report le stesse impostazioni di filtro per la stampa dei dati, che coincidono con ciò che è visualizzato nella finestra di Visualizzazione in questione.



Questa proprietà non è supportata in Windows CE. (Se impostata restituisce sempre 'false')

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Public Sub Click()
    Dim objTraceWnd As HisLogWndCmdTarget
    Set objTraceWnd =
    GetSynopticObject.GetSubObject("TraceLog").GetObjectInterface
    If Not objTraceWnd Is Nothing Then
        objTraceWnd .ShowPrintBtn = Not objTraceWnd .ShowPrintBtn
        objTraceWnd .RecalcLayout
    End If
    Set objTraceWnd = Nothing
End Sub

```

ShowRefreshBtn, TraceDBWndCmdTarget Property

Syntax ShowRefreshBtn = _Boolean

Description Questa proprietà permette di visualizzare il pulsante di comando per l'operazione di rinfresco dei dati nella finestra di Trace.

Parameter	Description
-----------	-------------

None	None
------	------

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim objTraceWnd As HisLogWndCmdTarget
    Set objTraceWnd = GetSynopticObject.GetSubObject("TraceLog").GetObjectInterface
    If Not objTraceWnd Is Nothing Then
        objTraceWnd.ShowRefreshBtn = Not objTraceWnd.ShowRefreshBtn
        objTraceWnd.RecalcLayout
    End If
    Set objTraceWnd = Nothing
End Sub
```

SortBy, TraceDBWndCmdTarget Property

Syntax SortBy = _String

Description Questa proprietà imposta o restituisce il filtro 'Ordina Per' per la visualizzazione dei messaggi nella finestra di Trace di Movicon.

I campi possibili sono:

ActionCol
AfterCol
BeforeCol
LocalCol
MSecCol
QualityCol
TimeCol
UserCol
ValueCol

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim objTraceWnd As TraceDBWndCmdTarget
    Set objTraceWnd = GetSynopticObject.GetSubObject("TraceLog").GetObjectInterface
    If Not objTraceWnd Is Nothing Then
        MsgBox "objTraceWnd 's SortBy is " & objTraceWnd.SortBy
        ,vbInformation,GetProjectTitle
        objTraceWnd.SortBy = "ActionCol"
        objTraceWnd.Refresh
    Else
        MsgBox "objTraceWnd is nothing",vbInformation,GetProjectTitle
    End If
End Sub
```

SubItemAction, TraceDBWndCmdTarget Property

Syntax SubItemAction = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Action". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface =
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd .SubItemAction = "Type action"
        TraceDBWnd .RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemActionPos, TraceDBWndCmdTarget Property

Syntax SubItemActionPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Azione" all'interno della Finestra DB Trace. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface =
    If Not TraceDBWnd Is Nothing Then
        Debug.Print TraceDBWnd .SubItemActionPos
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemActionWidth, TraceDBWndCmdTarget Property

Syntax SubItemActionWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione dei trace. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd .SubItemActionWidth = 20
        TraceDBWnd .RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemAfter, TraceDBWndCmdTarget Property

Syntax SubItemAfter = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "After". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd .SubItemAfter = "After that"
        TraceDBWnd .RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemAfterPos, TraceDBWndCmdTarget Property

Syntax SubItemAfterPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Dopo" all'interno della Finestra DB Trace. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        Debug.Print TraceDBWnd .SubItemAfterPos
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemAfterWidth, TraceDBWndCmdTarget Property

Syntax SubItemAfterWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione dei trace. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd .SubItemAfterWidth = 20
        TraceDBWnd .RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemBefore, TraceDBWndCmdTarget Property

Syntax SubItemBefore = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "After". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd .SubItemBefore = "Before that"
        TraceDBWnd .RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemBeforePos, TraceDBWndCmdTarget Property

Syntax SubItemBeforePos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Prima" all'interno della Finestra DB Trace. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        Debug.Print TraceDBWnd .SubItemBeforePos
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemBeforeWidth, TraceDBWndCmdTarget Property

Syntax SubItemBeforeWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione dei trace. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd .SubItemBeforeWidth = 20
        TraceDBWnd .RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemQuality, TraceDBWndCmdTarget Property

Syntax SubItemQuality = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Quality". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd .SubItemQuality = "With Quality"
        TraceDBWnd .RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemQualityPos, TraceDBWndCmdTarget Property

Syntax SubItemQualityPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Qualità" all'interno della Finestra DB Trace. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWnd
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        Debug.Print TraceDBWnd .SubItemQualityPos
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemQualityWidth, TraceDBWndCmdTarget Property

Syntax SubItemQualityWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione dei trace. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWnd
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd .SubItemQualityWidth = 20
        TraceDBWnd .RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemTime, TraceDBWndCmdTarget Property

Syntax SubItemTime = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Time". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd .SubItemTime = "At Time"
        TraceDBWnd .RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemTimePos, TraceDBWndCmdTarget Property

Syntax SubItemTimePos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Tempo Evento" all'interno della Finestra DB Trace. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        Debug.Print TraceDBWnd .SubItemTimePos
    End If
    Set TraceDBWnd = Nothing
End Sub
```


SubItemTimeStamp, TraceDBWndCmdTarget Property

Syntax SubItemTimeStamp = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "TimeStamp". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWnd
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface =
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd.SubItemTimeStamp = "Tag TimeStamp"
        TraceDBWnd.RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemTimeStampPos, TraceDBWndCmdTarget Property

Syntax SubItemTimeStampPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "TimeStamp" all'interno della Finestra DB Trace. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWnd
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface =
    If Not TraceDBWnd Is Nothing Then
        Debug.Print TraceDBWnd.SubItemTimeStampPos
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemTimeStampWidth, TraceDBWndCmdTarget Property

Syntax SubItemTimeStampWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna "TimeStamp" all'interno della finestra di visualizzazione di tracce. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd.SubItemTimeStampWidth = 20
        TraceDBWnd.RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemTimeWidth, TraceDBWndCmdTarget Property

Syntax SubItemTimeWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione dei tracce. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd.SubItemTimeWidth = 20
        TraceDBWnd.RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemUser, TraceDBWndCmdTarget Property

Syntax SubItemUser = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "User". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd .SubItemUser = "Logon User"
        TraceDBWnd .RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemUserPos, TraceDBWndCmdTarget Property

Syntax SubItemUserPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Utente" all'interno della Finestra DB Trace. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        Debug.Print TraceDBWnd .SubItemUserPos
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemUserWidth, TraceDBWndCmdTarget Property

Syntax SubItemUserWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione dei trace. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd .SubItemUserWidth = 20
        TraceDBWnd .RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemValue, TraceDBWndCmdTarget Property

Syntax SubItemValue = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "User". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd .SubItemValue = "Var. Value"
        TraceDBWnd .RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemValuePos, TraceDBWndCmdTarget Property

Syntax SubItemValuePos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Valore" all'interno della Finestra DB Trace. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWnd
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        Debug.Print TraceDBWnd .SubItemValuePos
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemValueWidth, TraceDBWndCmdTarget Property

Syntax SubItemValueWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna all'interno della finestra di visualizzazione dei trace. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWnd
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd .SubItemValueWidth = 20
        TraceDBWnd .RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemVarDesc, TraceDBWndCmdTarget Property

Syntax SubItemVarDesc = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Descrizione Tag". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd.SubItemVarDesc= "Tag Description"
        TraceDBWnd.RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemVarDescPos, TraceDBWndCmdTarget Property

Syntax SubItemVarDescPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Descrizione Tag" all'interno della Finestra DB Trace. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget =
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        Debug.Print TraceDBWnd.SubItemVarDescPos
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemVarDescWidth, TraceDBWndCmdTarget Property

Syntax SubItemVarDescWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna "Descrizione Tag" all'interno della finestra di visualizzazione di tracce. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWnd
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd.SubItemVarDescWidth = 20
        TraceDBWnd.RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemVarGroup, TraceDBWndCmdTarget Property

Syntax SubItemVarGroup = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Gruppo Tag". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWnd
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd.SubItemVarGroup = "Tag Group"
        TraceDBWnd.RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemVarGroupPos, TraceDBWndCmdTarget Property

Syntax SubItemVarGroupPos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Gruppo Tag" all'interno della Finestra DB Trace. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWnd
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        Debug.Print TraceDBWnd.SubItemVarGroupPos
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemVarGroupWidth, TraceDBWndCmdTarget Property

Syntax SubItemVarGroupWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna "Gruppo Tag" all'interno della finestra di visualizzazione di trace. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWnd
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd.SubItemVarGroupWidth = 20
        TraceDBWnd.RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```


SubItemVarName, TraceDBWndCmdTarget Property

Syntax SubItemVarName = _String

Description Permette di definire il testo che dovrà apparire come nome della colonna "Nome Tag". Se il campo viene lasciato vuoto verrà utilizzato il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd.SubItemVarName = "Tag Name"
        TraceDBWnd.RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemVarNamePos, TraceDBWndCmdTarget Property

Syntax SubItemVarNamePos = _Integer

Description Questa proprietà imposta o restituisce la posizione della colonna "Nome Tag" all'interno della Finestra DB Trace. Impostando un nuovo valore le altre colonne vengono automaticamente riposizionate nel layout della finestra. Inoltre impostando il valore "-1" la colonna verrà nascosta. Con il valore "0" si indica la posizione della prima colonna a sinistra della finestra.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        Debug.Print TraceDBWnd.SubItemVarNamePos
    End If
    Set TraceDBWnd = Nothing
End Sub
```

SubItemVarNameWidth, TraceDBWndCmdTarget Property

Syntax SubItemVarNameWidth = _Integer

Description Questa proprietà indica la dimensione in pixel della colonna "Nome Tag" all'interno della finestra di visualizzazione di trace. Se la colonna non è visualizzata, restituisce il valore -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        TraceDBWnd.SubItemVarNameWidth = 20
        TraceDBWnd.RecalcLayout
    End If
    Set TraceDBWnd = Nothing
End Sub
```

Variable, TraceDBWndCmdTarget Property

Syntax Variable = _String

Description Questa proprietà imposta o restituisce il nome della tabella di trace da associare alla finestra di visualizzazione. Se il nome della tabella di trace è uguale al nome della variabile associata allora pure questa proprietà riporta il nome della variabile.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim TraceDBWnd As TraceDBWndCmdTarget
    Set TraceDBWnd = TraceDBWndCmdTarget
    GetSynopticObject.GetSubObject("TraceDB").GetObjectInterface
    If Not TraceDBWnd Is Nothing Then
        Debug.Print TraceDBWnd.Variable
    End If
    Set TraceDBWnd = Nothing
End Sub
```

1.17.8. TrendCmdTarget

Events

OnChangingState, TrendCmdTarget Event

Description Evento notificato ogni volta che il Trend passa da Run a Stop e viceversa. La variabile booleana bRet abilita il cambiamento di stato, cioè se tale variabile viene impostata a False, il cambiamento di stato viene inibito.

Parameter	Description
bRet As Boolean	Abilitazione al cambiamento di stato.

OnCursorPosChanged, TrendCmdTarget Event

Description Evento notificato ogni volta che viene modificata la posizione del cursore, quando il Trend è in 'Stop'. La variabile nPos restituisce la posizione del cursore nel recordset dei valori del Trend (cioè nel numero dei campionamenti).

Parameter	Description
nPos As Long	Posizione del cursore nel recordset.

OnErrorRecordset, TrendCmdTarget Event

Description Evento notificato quando si verifica un errore durante la creazione del recordset dei valori del Trend. Un possibile motivo di errore può essere la sintassi della query associata al Trend sbagliata.



Quando la proprietà di esecuzione "Leggi Dati in Background" dell'oggetto Trend o Data Analysis è abilitata l'evento verrà automaticamente disabilitato.

Parameter	Description
RecordsetError As String	Messaggio di errore.

OnExpand, TrendCmdTarget Event

Description Evento notificato ogni volta che viene comandata l'espansione dell'area del Trend/Data Analysis e viceversa.

Parameter	Description
bRet As Boolean	Impostato a "false" consente di annullare l'operazione.

Example:

```

Public Sub OnExpand(ByRef bRet As Boolean)
    If MsgBox ("Do you want Expand Trend Area?", vbYesNo + vbQuestion,
        GetProjectTitle) = vbYes Then
        Debug.Print "Expanding Trend Area..."
    Else
        Debug.Print "Deleting Expand Trend Area..."
        bRet = False
    End If
End Sub

```

OnExportEnd, TrendCmdTarget Event

Description Evento notificato al termine dell'esportazione dei valori selezionati mediante apposito comando.

Parameter	Description
None	None

OnExportNext, TrendCmdTarget Event

Description Evento notificato ogni volta che viene esportato un successivo record dei valori selezionati mediante apposito comando.

Parameter	Description
NumRecord As Long	Numero del record in esportazione
bRet As Boolean	Abilitazione al proseguimento dell'esportazione

OnExportStart, TrendCmdTarget Event

Description Evento notificato all'inizio dell'esportazione dei valori selezionati mediante apposito comando.

Parameter	Description
None	None

OnFailedCreatingThread, TrendCmdTarget Event

Description Evento notificato se il sistema non riesce ad allocare le risorse per creare il thread necessario all'esecuzione del Trend. Viene quindi segnalata una condizione di anomalia molto grave in cui le risorse di sistema sono al di sotto dei minimi consentiti.

Parameter	Description

None	None
------	------

OnImportEnd, TrendCmdTarget Event

Description Evento notificato al termine dell'importazione dei valori selezionati mediante apposito comando.



Quando la proprietà di esecuzione "Leggi Dati in Background" dell'oggetto Trend o Data Analysis è abilitata l'evento verrà automaticamente disabilitato.

Parameter	Description
None	None

OnImportNext, TrendCmdTarget Event

Description Evento notificato ogni volta che viene importato un successivo record dei valori selezionati mediante apposito comando.



Quando la proprietà di esecuzione "Leggi Dati in Background" dell'oggetto Trend o Data Analysis è abilitata l'evento verrà automaticamente disabilitato.

Parameter	Description
NumRecord As Long	Numero del record in importazione
bRet As Boolean	Abilitazione al proseguimento dell'esportazione

OnImportStart, TrendCmdTarget Event

Description Evento notificato all'inizio dell'importazione dei valori selezionati mediante apposito comando.



Quando la proprietà di esecuzione "Leggi Dati in Background" dell'oggetto Trend o Data Analysis è abilitata l'evento verrà automaticamente disabilitato.

Parameter	Description
None	None

OnNext, TrendCmdTarget Event

Description Evento notificato ogni volta che viene comandato lo spostamento a sinistra del grafico (pulsante ">").

Parameter	Description
bRet As Boolean	Impostato a "false" consente di annullare l'operazione.

Example:

```
Public Sub OnNext(ByRef bRet As Boolean)
    If MsgBox ("Do you want execute Next Command ?", vbYesNo + vbQuestion,
        GetProjectTitle) = vbYes Then
        Debug.Print "Executing Next Command..."
    Else
        Debug.Print "Deleting Next Command..."
        bRet = False
    End If
End Sub
```

OnPageChanged, TrendCmdTarget Event

Description Evento notificato al cambiare di una pagina durante lo scorrimento quando il Trend è in pausa.

Parameter	Description
None	None

OnPageEnd, TrendCmdTarget Event

Description Evento notificato ogni volta che si raggiunge l'ultima pagina del buffer dei valori durante lo scorrimento quando il Trend è in pausa.

Parameter	Description
None	None

OnPageNext, TrendCmdTarget Event

Description Evento notificato ogni volta che viene comandato lo spostamento a sinistra di una pagina del grafico (pulsante ">>").

Parameter	Description
bRet As Boolean	Impostato a "false" consente di annullare l'operazione.

Example:

```
Public Sub OnPageNext(ByRef bRet As Boolean)
    If MsgBox ("Do you want execute Page Next Command ?", vbYesNo +
        vbQuestion, GetProjectTitle) = vbYes Then
        Debug.Print "Executing Page Next Command..."
    Else
        Debug.Print "Deleting Page Next Command..."
        bRet = False
    End If
End Sub
```

OnPagePrev, TrendCmdTarget Event

Description Evento notificato ogni volta che viene comandato lo spostamento a destra di una pagina del grafico (pulsante "<<").

Parameter	Description
bRet As Boolean	Impostato a "false" consente di annullare l'operazione.

Example:

```
Public Sub OnPagePrev(ByRef bRet As Boolean)
    If MsgBox ("Do you want execute Page Prev Command ?", vbYesNo +
vbQuestion, GetProjectTitle) = vbYes Then
        Debug.Print "Executing Page Prev Command..."
    Else
        Debug.Print "Deleting Page Prev Command..."
        bRet = False
    End If
End Sub
```

OnPageStart, TrendCmdTarget Event

Description Evento notificato ogni volta che si raggiunge la prima pagina del buffer dei valori durante lo scorrimento quando il Trend è in pausa.

Parameter	Description
None	None

OnPositionScrolled, TrendCmdTarget Event

Description Evento notificato ogni volta che viene eseguito uno spostamento in avanti o indietro di una posizione nel buffer dei valori quando il Trend è in pausa.

Parameter	Description
None	None

OnPrev, TrendCmdTarget Event

Description Evento notificato ogni volta che viene comandato lo spostamento a destra del grafico (pulsante "<").

Parameter	Description
bRet As Boolean	Impostato a "false" consente di annullare l'operazione.

Example:

```
Public Sub OnPrev(ByRef bRet As Boolean)
```

```

    If MsgBox ("Do you want execute Previous Command ?", vbYesNo + vbQuestion,
    GetProjectTitle) = vbYes Then
        Debug.Print "Executing Previous Command..."
    Else
        Debug.Print "Deleting Previous Command..."
    bRet = False
    End If
End Sub

```

OnPrint, TrendCmdTarget Event

Description Evento notificato ogni volta che viene richiesto di effettuare la stampa dei dati caricati nella finestra Trend.



Questo evento non è supportato in Windows CE.

Parameter	Description
bRet As Boolean	Abilitazione all'invio della stampa.

OnRecordsetMoveNext, TrendCmdTarget Event

Description Evento notificato durante lo scorrimento del set di valori che soddisfano la query di selezione, ogni volta che il sistema acquisisce un nuovo valore.



Quando la proprietà di esecuzione "Leggi Dati in Background" dell'oggetto Trend o Data Analysis è abilitata l'evento verrà automaticamente disabilitato.

Parameter	Description
NumRecord As Long	Indice del record attualmente puntato
bRet As Boolean	Abilitazione al proseguimento dello scorrimento

OnRecordsetQueryEnd, TrendCmdTarget Event

Description Evento notificato al termine dell'acquisizione dei valori che soddisfano la query di selezione.



Quando la proprietà di esecuzione "Leggi Dati in Background" dell'oggetto Trend o Data Analysis è abilitata l'evento verrà automaticamente disabilitato.

Parameter	Description
None	None

OnRecordsetQueryStart, TrendCmdTarget Event

Description Evento notificato all'inizio dell'acquisizione dei valori che soddisfano la query di selezione.



Quando la proprietà di esecuzione "Leggi Dati in Background" dell'oggetto Trend o Data Analysis è abilitata l'evento verrà automaticamente disabilitato.

Parameter	Description
None	None

OnResetZoom, TrendCmdTarget Event

Description L'evento viene richiamato in uscita da uno zoom mode, cioè quando si preme il tasto ESC o si richiama il metodo "ResetZoom".

Nota: l'evento "OnResetZoom" non viene più richiamato se dall'evento "OnStartZoom" si è usciti con bRet = False. Questo comportamento è By-Design.

Parameter	Description
bRet As Boolean	Impostando questo parametro a "False" è possibile annullare l'uscita dallo Zoom mode.

OnStartRecording, TrendCmdTarget Event

Description Evento notificato ogni volta che si avvia la registrazione dei valori del Trend.

Parameter	Description
None	None

OnStartZoom, TrendCmdTarget Event

Description L'evento viene richiamato dopo che è stata selezionata l'area di zoom ma prima che venga applicato lo zoom dell'area selezionata. Grazie a questo metodo è possibile eseguire delle operazioni sui dati selezionati all'interno dell'area di zoom, ad esempio effettuare un requery per vedere i soli dati selezionati.

Parameter	Description
FromRecord As Long	Posizione del primo valore selezionato all'interno dell'area di zoom. La posizione è riferita al buffer di lavoro del Trend e con altre funzioni (tipo "SampleDateTime") è possibile recuperare altre informazioni, come ad esempio la data e l'ora di quella registrazione.
ToRecord As Long	Posizione dell'ultimo valore selezionato all'interno dell'area di zoom. La posizione è riferita al buffer di lavoro del Trend e con altre funzioni (tipo "SampleDateTime") è possibile recuperare altre

	informazioni, come ad esempio la data e l'ora di quella registrazione.
bRet As Boolean	Impostando questo parametro a "False" è possibile impedire che venga effettuato uno zoom dell'area selezionata.

OnStopRecording, TrendCmdTarget Event

Description Evento notificato ogni volta che si interrompe la registrazione dei valori del Trend.

Parameter	Description
None	None

OnUpdateData, TrendCmdTarget Event

Description Evento notificato ogni volta che avviene un aggiornamento a video dei valori del Trend.

Parameter	Description
None	None

OnZoomAreaChanged, TrendCmdTarget Event

Description L'evento viene richiamato dopo che è stata selezionata l'area di zoom ma prima che venga applicato lo zoom dell'area selezionata. L'evento restituisce le informazioni riguardanti la dimensione e la posizione dell'area di Zoom selezionata.

Parameter	Description
X As Integer	Coordinata orizzontale del vertice in alto a sinistra del rettangolo che rappresenta l'area di Zoom selezionata.
Y As Integer	Coordinata verticale del vertice in alto a sinistra del rettangolo che rappresenta l'area di Zoom selezionata.
nWidth As Integer	Dimensione in pixel della larghezza del rettangolo che rappresenta l'area di Zoom selezionata.
nHeight As Integer	Dimensione in pixel dell'altezza del rettangolo che rappresenta l'area di Zoom selezionata.

Functions

AddPen, TrendCmdTarget Function

Syntax AddPen(IpszName, IpszVariable, dMinValue, dMaxValue, PenColor, BrushColor, nPlotType)

Description Questa funzione consente di aggiungere, durante l'esecuzione del progetto, una penna all'oggetto Trend. Prima di eseguire la funzione è necessario interrompere la registrazione mediante la funzione "Recording". Essendo il Data Analysis uno strumento per l'analisi "Off-Line" dei dati presenti nel Data Base, deve necessariamente essere collegato un Data Logger e, a differenza dell'oggetto Trend, non è possibile aggiungere una penna legata ad una variabile che non è già contemplata nel Data Logger.

Parameter	Description
IpszName As String	Nome della penna da aggiungere
IpszVariable As String	Variabile associata alla penna
dMinValue As Double	Valore minimo della penna
dMaxValue As Double	Valore massimo della penna
PenColor As Long	Colore della penna
BrushColor As Long	Colore del pennello
nPlotType As Integer	Indice del tipo di linea per rappresentare la penna.

Result Boolean

Example:

```
Option Explicit
Const DLColName = "SimCosInt"
Public Sub Click()
Dim strVar As String
GetVariableNameFromList(strVar)
If Len(strVar)<>0 Then
objDataAnalysis.AddPen("MyPen", strVar, -100, 100, RGB(255,0,0), RGB(0,255,0), 0)
objDataAnalysis.Refresh
objDataAnalysis.PenDLCColumnName("MyPen") = DLColName
End If
End Sub
```

ClearAllSavedValues, TrendCmdTarget Function

Syntax ClearAllSavedValues()

Description Pulisce l'area Trend da tutti i valori delle curve precedentemente salvati. I valori delle penne possono essere salvati sull'area Trend con la funzione SaveAllCurrentValue e sono visibili sul Trend con delle linee tratteggiate fino a quando non è richiamata la funzione ClearAllSavedValues.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.ClearAllSavedValues
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

ClearSavedValues, TrendCmdTarget Function

Syntax ClearSavedValues(_IpszPenName)

Description Pulisce l'area Trend da tutti i valori precedentemente salvati per la curva specificata con il parametro IpszPenName. I valori di una penna possono essere salvati sull'area Trend con la funzione SaveCurrentValue, e sono visibili sul Trend con una linea tratteggiata fino a quando non è richiamata la funzione ClearSavedValues.

Parameter	Description
IpszPenName As String	Nome della penna di cui salvare i valori

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.ClearSavedValues(objTrend.GetPenNameFromList(0))
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

CloseBackupLink, TrendCmdTarget Function

Syntax CloseBackupLink()

Description Questa funzione chiude la connessione al database di backup impostata con la proprietà BackupLink e ritorna a prelevare i dati dal Data Logger originariamente impostato sul Trend. L'aggiornamento dei dati deve essere forzato con le funzioni Requery e Refresh oppure cambiando stato del Trend.



Questa proprietà non è supportata in Windows CE.

Parameter	Description
-----------	-------------

None	None
------	------

Result None

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.CloseBackupLink
    objTrend.Requery
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set                                objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

CopyLegendToClipboard, TrendCmdTarget Function

Syntax CopyLegendToClipboard()

Description Questa funzione copia negli appunti di Windows, in formato testo, i campi e i valori presenti nella legenda dell'oggetto Trend. In questo modo e' possibile avere a disposizione i valori istantanei del Trend per poterli riportare su altro file mediante il comando Windows di "Incolla".



Questa funzione non è supportata in Windows CE.(Se impostata restituisce sempre ' false')



Questa funzione non è gestita dall'oggetto "Data Analysis".

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.CopyLegendToClipboard
End Sub
Public Sub SymbolLoading()
    Set                                objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

EditPenProperties, TrendCmdTarget Function

Syntax EditPenProperties(_IpszPenName)

Description Questa funzione consente di modificare durante l'esecuzione del progetto le impostazioni generali di una penna. Viene cioè visualizzata la maschera di configurazione della penna stessa.



Questa funzione non è supportata in Windows CE. (Se impostata restituisce sempre 'false')

Parameter	Description
IpszPenName As String	Nome della penna da editare

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.EditPenProperties(objTrend.GetPenNameFromList(0))
End Sub
Public Sub SymbolLoading()
    Set                                objTrend                                =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

ExportToClipboard, TrendCmdTarget Function

Syntax ExportToClipboard(_IStart, _IEnd)

Description Questa funzione esegue l'esportazione dei record specificati negli appunti di Windows. Accetta due parametri di tipo Long contenenti gli indici del primo e dell'ultimo record da esportare. Se tali parametri assumono rispettivamente i valori 0 e -1, vengono esportati tutti i record dimensionati nel buffer del Trend. Se invece tali parametri assumono i valori 0 e 0, viene abilitata la selezione mediante mouse dei record da esportare; in tal modo è possibile selezionare i record trascinando il puntatore del mouse sull'area del Trend. Tale funzione è operativa solamente quando il Trend è in Stop.

Nota: I record più recenti in ordine di tempo hanno indice più basso.



Questa funzione non è supportata in Windows CE (Se impostata restituisce sempre 'false').



Questa funzione non è gestita dall'oggetto "Data Analysis".

Parameter	Description
IStart As Long	Numero del primo record da copiare
IEnd As Long	Numero dell'ultimo record da copiare

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.ExportToClipboard(12,154)
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

ExportToFile, TrendCmdTarget Function

Syntax ExportToFile(_IpszFileName, _IStart, _IEnd)

Description Questa funzione esegue l'esportazione dei record specificati su file di testo ".csv". Accetta un parametro di tipo String contenente il nome del file di destinazione e due parametri di tipo Long contenenti gli indici del primo e dell'ultimo record da salvare. Se tali parametri assumono rispettivamente i valori 0 e -1, vengono esportati tutti i record dimensionati nel buffer del Trend. Se invece tali parametri assumono i valori 0 e 0, viene abilitata la selezione mediante mouse dei record da esportare; in tal modo è possibile selezionare i record trascinando il puntatore del mouse sull'area del Trend. Tale funzione è operativa solamente quando il Trend è in Stop.



I record più recenti in ordine di tempo hanno indice più basso.



Questa funzione non è gestita dall'oggetto "Data Analysis".

Parameter	Description
IpszFileName	Nome del file di destinazione
IStart As Long	Numero del primo record da copiare
IEnd As Long	Numero dell'ultimo record da copiare

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.ExportToFile(GetDataLoggerRecipePath & "FileData.csv", 0, -1)
End Sub

Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

GetCurrentDataLoggerName, TrendCmdTarget Function

Syntax GetCurrentDataLoggerName()

Description Questa funzione restituisce il nome del Data Logger utilizzato per estrarre i dati. Nel caso al Data Analysis siano associate penne appartenenti a Data Logger differenti, le funzioni "OnRecordsetQueryStart" e "OnRecordsetQueryEnd" vengono chiamate per ognuno dei Data Logger e la funzione "GetCurrentDataLoggerName" può essere usata per sapere qual'è il Data Logger corrente durante il caricamento dati.



Questa funzione non è gestita dall'oggetto "Trend".

Parameter	Description
None	None

Result String

Example:

Option Explicit

Dim sDataLoggerName As String

Public Sub OnRecordsetQueryStart()

 sDataLoggerName = **GetCurrentDataLoggerName**()

 Debug.Print "OnRecordsetQueryStart Executed: DataLogger Name = " &

 sDataLoggerName

End Sub

Public Sub OnRecordsetQueryEnd()

 sDataLoggerName = **GetCurrentDataLoggerName**()

 Debug.Print "OnRecordsetQueryEnd Executed: DataLogger Name = " &

 sDataLoggerName

End Sub

GetCursorDateTime, TrendCmdTarget Function

Syntax GetCursorDateTime()

Description Questa funzione restituisce, in formato Date, la data e l'ora corrispondenti alla posizione del cursore nel Trend e Data Analysis. Restituisce un valore nullo quando il Trend è in run.

Parameter	Description
None	None

Result Date

Example:

Option Explicit

Dim objTrend As TrendCmdTarget

Public Sub Click()


```

        Dim strVar As String
        If objTrend Is Nothing Then Exit Sub
        objTrend.Freezed = True
        Debug.Print objTrend.GetCursorDateTime
    End Sub
    Public Sub SymbolLoading()
        Set                                     objTrend
        GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
    End Sub

```

GetCursorDateTimeMsec, TrendCmdTarget Function

Syntax GetCursorDateTimeMsec()

Description Questa funzione restituisce il numero di millisecondi, se registrati, abbinati alla data e ora corrispondenti alla posizione del cursore del Trend e Data Analysis. Restituisce un valore nullo quando il Trend è in run.

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.Freezed = True
    Debug.Print objTrend.GetCursorDateTimeMsec
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

GetCursorDateTimeString, TrendCmdTarget Function

Syntax GetCursorDateTimeString()

Description Questa funzione restituisce, in formato Stringa, la data e l'ora corrispondenti alla posizione del cursore del Trend e Data Analysis. Restituisce un valore nullo quando il Trend è in run.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.Freezed = True
    Debug.Print objTrend.GetCursorDateTimeString
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

GetCursorPenValue, TrendCmdTarget Function

Syntax GetCursorPenValue(_IpszPenName)

Description Questa funzione restituisce, in formato double, il valore corrispondente all'intersezione della penna indicata dal parametro passato e il cursore impostato. Nel caso di un Trend XY restituisce la posizione del cursore X o Y in riferimento al nome della penna passato come parametro.

Parameter	Description
IpszPenName As String	Nome della penna per la quale recuperare il valore.

Result Double

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Dim valPen As Double
Public Sub Click()
    Set objTrend = GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    valPen = objTrend.GetCursorPenValue("CurveName")
    MsgBox "GetCursorPenValue = " &
    GetCursorPenValue,vbInformation,GetProjectTitle
    Set objTrend = Nothing
End Sub

```

GetCursorPosInLegendArea, TrendCmdTarget Function

Syntax GetCursorPosInLegendArea(_pnX, _pnY)

Description Questa funzione restituisce le coordinate X e Y indicanti la posizione del cursore del mouse rispetto all'origine dell'area della legenda.

Parameter	Description
pnX As Integer	Coordinata X del cursore

pnY As Integer	Coordinata Y del cursore
----------------	--------------------------

Result Double

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    objTrend.Freezed = True
    objTrend.GetCursorPosInLegendArea(pnX, pnY)
    Debug.Print pnX & " " & pnY
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

GetCursorPosInPenArea, TrendCmdTarget Function

Syntax GetCursorPosInPenArea(_pnX, _pnY)

Description Questa funzione restituisce le coordinate X e Y indicanti la posizione del cursore del mouse rispetto all'origine dell'area delle penne.

Parameter	Description
pnX As Integer	Coordinata X del cursore
pnY As Integer	Coordinata Y del cursore

Result Double

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    objTrend.Freezed = True
    objTrend.GetCursorPosInLegendArea(pnX, pnY)
    Debug.Print pnX & " " & pnY
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

GetCursorPosInScaleArea, TrendCmdTarget Function

Syntax GetCursorPosInScaleArea(_lpzPenName, _pnX, _pnY)

Description	Questa funzione restituisce le coordinate X e Y indicanti la posizione del cursore del mouse rispetto all'origine dell'area della scala della penna specificata.
--------------------	--

Parameter	Description
lpzPenName As String	Nome della penna
pnX As Integer	Coordinata X del cursore
pnY As Integer	Coordinata Y del cursore

Result	Double
---------------	--------

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    objTrend.Freeze = True
    objTrend.GetCursorPosInScaleArea(objTrend.GetPenNameFromList(0),pnX,
    pnY)
    Debug.Print pnX & " " & pnY
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

GetCursorPosInTimeArea, TrendCmdTarget Function

Syntax GetCursorPosInTimeArea(_pnX, _pnY)

Description	Questa funzione restituisce le coordinate X e Y indicanti la posizione del cursore del mouse rispetto all'origine dell'area dell'orologio.
--------------------	--

Parameter	Description
pnX As Integer	Coordinata X del cursore
pnY As Integer	Coordinata Y del cursore

Result	Double
---------------	--------

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
```

```

Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    objTrend.Freezed = True
    objTrend.GetCursorPosInTimeArea(pnX, pnY)
    Debug.Print pnX & " " & pnY
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
=

```

GetCursorPosInTrendArea, TrendCmdTarget Function

Syntax GetCursorPosInTrendArea(_pnX, _pnY)

Description Questa funzione restituisce le coordinate X e Y indicanti la posizione del cursore del mouse rispetto all'origine dell'area della finestra del Trend.

Parameter	Description
pnX As Integer	Coordinata X del cursore
pnY As Integer	Coordinata Y del cursore

Result Double

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    objTrend.Freezed = True
    objTrend.GetCursorPosInTrendArea(pnX, pnY)
    Debug.Print pnX & " " & pnY
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
=

```

GetDateTimeColumnName, TrendCmdTarget Function

Syntax GetDateTimeColumnName

Description Questa funzione restituisce una stringa contenente il nome impostato per la colonna Tempo nel data logger associato al Trend.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    objTrend.Freezed = True
    Debug.Print objTrend.GetDateTimeColumnName
End Sub
Public Sub SymbolLoading()
    Set                                objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

GetFirstValidDateTime, TrendCmdTarget Function

Syntax GetFirstValidDateTime

Description Questa funzione restituisce, in formato Date, la data e l'ora corrispondenti alla prima registrazione utile nel buffer dei valori del Trend e del Data Analysis. Per quanto riguarda il Trend il valore viene restituito sia che questo sia in run o che sia in pausa.

Parameter	Description
None	None

Result Date

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.GetFirstValidDateTime
End Sub
Public Sub SymbolLoading()
    Set                                objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

GetFirstValidDateTimeMs, TrendCmdTarget Function

Syntax GetFirstValidDateTimeMs

Description Questa funzione restituisce il numero di millisecondi, se registrati, abbinati alla data e ora corrispondenti alla prima registrazione utile nel buffer dei valori del Trend e del Data Analysis. Per quanto riguarda il Trend il valore viene restituito sia che questo sia in run o che sia in pausa.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.GetFirstValidDateTimeMs
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

GetLastValidDateTime, TrendCmdTarget Function

Syntax GetLastValidDateTime

Description Questa funzione restituisce, in formato Date, la data e l'ora corrispondenti all'ultima registrazione, quindi la più recente, nel buffer dei valori del Trend e del Data Analysis. Per quanto riguarda il Trend il valore viene restituito sia che questo sia in run o che sia in pausa.

Parameter	Description
None	None

Result Date

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.GetLastValidDateTime
End Sub
```

```

Public Sub SymbolLoading()
    Set                                objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

GetLastValidDateTimeMs, TrendCmdTarget Function

Syntax GetLastValidDateTimeMs

Description Questa funzione restituisce il numero di millisecondi, se registrati, abbinati alla data e ora corrispondenti all'ultima registrazione, quindi la piu' recente, nel buffer dei valori del Trend e del Data Analysis. Per quanto riguarda il Trend il valore viene restituito sia che questo sia in run o che sia in pausa.

Parameter	Description
None	None

Result Long

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.GetLastValidDateTimeMs
End Sub
Public Sub SymbolLoading()
    Set                                objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

GetLastValidValuePosition, TrendCmdTarget Function

Syntax GetLastValidValuePosition

Description Questa funzione restituisce il valore del cursore nella posizione corrispondente all'ultima registrazione, quindi la più recente, nel buffer dei valori del Trend, sia che questo sia in run o che sia in pausa.

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget

```



```

Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.GetLastValidValuePosition
End Sub
Public Sub SymbolLoading()
    Set                                objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

GetMaxPage, TrendCmdTarget Function

Syntax GetMaxPage

Description Questa funzione restituisce il numero totale di pagine visualizzabili nella finestra del Trend in base al numero massimo di campionamenti e al numero di campionamenti visualizzati per pagina.

Parameter	Description
None	None

Result Long

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.GetMaxPage
End Sub
Public Sub SymbolLoading()
    Set                                objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

GetPenNameFromList, TrendCmdTarget Function

Syntax GetPenNameFromList(_nIndex)

Description Questa funzione restituisce una stringa con il nome della penna riferita al parametro 'nIndex' di tipo intero (la prima penna corrisponde all'indice 0). Se l'indice passato come parametro non si riferisce a nessuna penna, allora la funzione restituisce una stringa vuota.

Parameter	Description
nIndex As Integer	Indice di riferimento della penna

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.GetPenNameFromList(0)
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

GetPensNumber, TrendCmdTarget Function

Syntax GetPensNumber

Description Questa funzione restituisce un valore di tipo intero che indica il numero di penne configurate nel Trend.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.GetPensNumber
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

GetTimeFontOrientation, TrendCmdTarget Function

Syntax GetTimeFontOrientation

Description Questa funzione restituisce un valore di tipo intero che indica l'orientamento del font nell'area dell'orario del Trend.

Parameter	Description
-----------	-------------

None	None
------	------

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.GetTimeFontOrientation
End Sub
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

GetZoomAreaDateTimeFrom, TrendCmdTarget Function

Syntax GetZoomAreaDateTimeFrom()

Description Questa funzione restituisce, in formato Data, la data e l'ora corrispondenti al primo punto dell'area di Zoom applicato.

Parameter	Description
None	None

Result DATE

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Dim sZoomAreaDateTimeFrom As Date
Public Sub Click()
    Set objTrend = GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    sZoomAreaDateTimeFrom = objTrend.GetZoomAreaDateTimeFrom
    MsgBox "GetZoomAreaDateTimeFrom = " & Format(sZoomAreaDateTimeFrom, "YYYY/MM/DD hh:nn:ss"), vbInformation, GetProjectTitle
    Set objTrend = Nothing
End Sub
```

GetZoomAreaDateTimeMsecFrom, TrendCmdTarget Function

Syntax GetZoomAreaDateTimeMsecFrom()

Description Questa funzione restituisce, in formato numerico, il numero di millisecondi della data e ora corrispondenti al primo punto dell'area di Zoom applicato.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Dim sZoomAreaDateTimeMsecFrom As Integer
Public Sub Click()
    Set objTrend = GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    sZoomAreaDateTimeMsecFrom = objTrend.GetZoomAreaDateTimeMsecFrom
    MsgBox "GetZoomAreaDateTimeMsecFrom = " &
    CStr(sZoomAreaDateTimeMsecFrom), vbInformation, GetProjectTitle
    Set objTrend = Nothing
End Sub
```

GetZoomAreaDateTimeMsecTo, TrendCmdTarget Function

Syntax GetZoomAreaDateTimeMsecTo()

Description Questa funzione restituisce, in formato numerico, il numero di millisecondi della data e ora corrispondenti all'ultimo punto dell'area di Zoom applicato.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Dim sZoomAreaDateTimeMsecFrom As Integer
Public Sub Click()
    Set objTrend = GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    sZoomAreaDateTimeMsecFrom = objTrend.GetZoomAreaDateTimeMsecTo
    MsgBox "GetZoomAreaDateTimeMsecTo=" &
    CStr(sZoomAreaDateTimeMsecFrom), vbInformation, GetProjectTitle
    Set objTrend = Nothing
End Sub
```

GetZoomAreaPenValueFrom, TrendCmdTarget Function

Syntax GetZoomAreaPenValueFrom(_IpszPenName)

Description Questa funzione restituisce, in formato double, il valore corrispondente all'intersezione della penna con il nome passato come parametro e il primo estremo dell'area di zoom, quello meno recente sull'asse X. Funziona anche se non è applicato lo zoom.

Parameter	Description
lpszPenName as string	Nome della penna per la quale recuperare il valore.

Result Double

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Dim valPenFrom As Double
Public Sub Click()
    Set objTrend = GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    valPenFrom = objTrend.GetZoomAreaPenValueFrom("CurveName")

    MsgBox "GetZoomAreaPenValueFrom = " &
    GetZoomAreaPenValueFrom,vbInformation,GetProjectTitle
    Set objTrend = Nothing
End Sub
```

GetZoomAreaPenValueTo,TrendCmdTarget Function

Syntax GetZoomAreaPenValueTo(_lpszPenName)

Description Questa funzione restituisce, in formato double, il valore corrispondente all'intersezione della curva con il nome passato come parametro e il secondo estremo dell'area di zoom, quello più recente sull'asseX. Funziona anche se non è applicato lo zoom.

Parameter	Description
lpszPenName as string	Nome della penna per la quale recuperare il valore.

Result Double

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Dim valPenTo As Double
Public Sub Click()
    Set objTrend = GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    valPenTo = objTrend.GetZoomAreaPenValueTo("CurveName")
    MsgBox "GetZoomAreaPenValueTo = " &
    GetZoomAreaPenValueTo,vbInformation,GetProjectTitle
    Set objTrend = Nothing
End Sub
```

GetZoomAreaDateTimeStringFrom, TrendCmdTarget Function

Syntax GetZoomAreaDateTimeStringFrom()

Description Questa funzione restituisce, in formato Stringa, la data e l'ora corrispondenti al primo punto dell'area di Zoom applicato.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Dim sZoomAreaDateTimeFrom As String
Public Sub Click()
    Set objTrend = GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    sZoomAreaDateTimeFrom = objTrend.GetZoomAreaDateTimeStringFrom
    MsgBox "GetZoomAreaDateTimeStringFrom = " & sZoomAreaDateTimeFrom, vbInformation, GetProjectTitle
    Set objTrend = Nothing
End Sub
```

GetZoomAreaDateTimeStringTo, TrendCmdTarget Function

Syntax GetZoomAreaDateTimeStringTo()

Description Questa funzione restituisce, in formato Stringa, la data e l'ora corrispondenti all'ultimo punto dell'area di Zoom applicato.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Dim sZoomAreaDateTimeFrom As String
Public Sub Click()
    Set objTrend = GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    sZoomAreaDateTimeFrom = objTrend.GetZoomAreaDateTimeStringTo
    MsgBox "GetZoomAreaDateTimeStringTo = " & sZoomAreaDateTimeFrom, vbInformation, GetProjectTitle
    Set objTrend = Nothing
End Sub
```

GetZoomAreaDateTimeTo, TrendCmdTarget Function

Syntax GetZoomAreaDateTimeTo()

Description Questa funzione restituisce, in formato Data, la data e l'ora corrispondenti all'ultimo punto dell'area di Zoom applicato.

Parameter	Description
None	None

Result DATE

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Dim sZoomAreaDateTimeFrom As Date
Public Sub Click()
    Set objTrend = GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    sZoomAreaDateTimeFrom = objTrend.GetZoomAreaDateTimeTo
    MsgBox "GetZoomAreaDateTimeTo = " & Format(sZoomAreaDateTimeFrom,
    "YYYY/MM/DD hh:nn:ss"),vbInformation,GetProjectTitle
    Set objTrend = Nothing
End Sub
```

GetZoomAreaScaleValueFrom, TrendCmdTarget Function

Syntax GetZoomAreaScaleValueFrom(_IpszPenName)

Description Questa funzione restituisce, in formato numerico, il valore minimo di scala, della penna indicata, corrispondente all'area di Zoom applicato.

Parameter	Description
IpszPenName as string	Nome della penna per la quale recuperare il valore minimo di scala

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Dim sZoomAreaScaleValueFrom As Double
Public Sub Click()
    Set objTrend = GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    sZoomAreaScaleValueFrom =
    objTrend.GetZoomAreaScaleValueFrom("Var_SignWord")
    MsgBox "GetZoomAreaScaleValueFrom = " &
    CStr(sZoomAreaScaleValueFrom),vbInformation,GetProjectTitle
    Set objTrend = Nothing
End Sub
```

GetZoomAreaScaleValueTo, TrendCmdTarget Function

Syntax GetZoomAreaScaleValueTo(_IpszPenName)

Description Questa funzione restituisce, in formato numerico, il valore massimo di scala, della penna indicata, corrispondente all'area di Zoom applicato.

Parameter	Description
lpszPenName as string	Nome della penna per la quale recuperare il valore massimo di scala.

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Dim sZoomAreaScaleValueFrom As Double
Public Sub Click()
    Set objTrend = GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    sZoomAreaScaleValueFrom =
    objTrend.GetZoomAreaScaleValueTo("Var_SignWord")
    MsgBox "GetZoomAreaScaleValueTo = " &
    CStr(sZoomAreaScaleValueFrom), vbInformation, GetProjectTitle
    Set objTrend = Nothing
End Sub
```

ImportFromClipboard, TrendCmdTarget Function

Syntax ImportFromClipboard(_lStart, _lEnd)

Description Questa funzione esegue l'importazione dei valori dei record specificati dagli appunti di Windows. Accetta due parametri di tipo Long contenenti gli indici del primo e dell'ultimo record. Se tali parametri assumono rispettivamente i valori 0 e -1, vengono importati tutti i record dimensionati nel buffer del Trend. Se invece tali parametri assumono i valori 0 e 0, viene abilitata la selezione mediante mouse dei record da importare; in tal modo è possibile selezionare i record trascinando il puntatore del mouse sull'area del Trend. Tale funzione è operativa solamente quando il Trend è in Stop. Una volta chiamata la funzione è necessario eseguire un refresh del Trend mediante apposito comando. Inoltre, tutti record di indice più alto rispetto all'ultimo importato vengono eliminati dal buffer del Trend. Tutte queste operazioni hanno effetto solo sui valori temporanei del Trend ma non influenzano l'eventuale Data Logger associato.

Nota: I record più recenti in ordine di tempo hanno indice più basso.



Questa funzione non è supportata in Windows CE. (Se impostata restituisce sempre 'false')



Questa funzione non è gestita dall'oggetto "Data Analysis".

Parameter	Description
lStart As Long	Numero del primo record da copiare
lEnd As Long	Numero dell'ultimo record da copiare

Result Boolean

Example:


```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.ImportFromClipboard(12,154)
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface =
End Sub

```

ImportFromFile, TrendCmdTarget Function

Syntax ImportFromFile (IpszFileName, IStart, IEnd)

Description Questa funzione esegue l'importazione da un file di testo dei valori dei record specificati. Accetta un parametro di tipo String contenente il nome del file sorgente e due parametri di tipo Long contenenti gli indici del primo e dell'ultimo record da recuperare. Se tali parametri assumono rispettivamente i valori 0 e -1, vengono importati tutti i record dimensionati nel buffer del Trend. Se invece tali parametri assumono entrambi il valore 0, viene abilitata la selezione mediante mouse dei record da importare; è possibile in tal modo selezionare i record trascinando il puntatore del mouse sull'area del Trend. Tale funzione è operativa solamente quando il Trend è in Stop. Una volta chiamata la funzione è necessario eseguire un refresh del Trend mediante apposito comando. E' da tener presente inoltre che tutti record di indice più alto rispetto all'ultimo importato vengono eliminati dal buffer del Trend.

Tutte queste operazioni hanno effetto solo sui valori temporanei del Trend ma non influenzano l'eventuale Data Logger associato.

Nota: I record più recenti in ordine di tempo hanno indice più basso.



Questa funzione non è gestita dall'oggetto "Data Analysis".

Parameter	Description
IpszFileName As String	Nome del file da importare
IStart As Long	Numero del primo record da copiare
IEnd As Long	Numero dell'ultimo record da copiare

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.ImportFromFile ("exp.txt ",12,154)
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface =
End Sub

```

IsCursorPosInLegendArea, TrendCmdTarget Function

Syntax `IsCursorPosInLegendArea`

Description	Questa funzione restituisce il valore booleano True se il cursore del mouse si trova nell'area della legenda del Trend.
--------------------	---

Parameter	Description
None	None

Result	Boolean
---------------	---------

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.IsCursorPosInLegendArea
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

IsCursorPosInPenArea, TrendCmdTarget Function

Syntax `IsCursorPosInPenArea`

Description	Questa funzione restituisce il valore booleano True se il cursore del mouse si trova nell'area penne del Trend.
--------------------	---

Parameter	Description
None	None

Result	Boolean
---------------	---------

Example:

```

Example:
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.IsCursorPosInPenArea
End Sub
Public Sub SymbolLoading()
    Set                                objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

IsCursorPosInScaleArea, TrendCmdTarget Function

Syntax IsCursorPosInScaleArea(_IpszPenName)

Description Questa funzione restituisce il valore booleano True se il cursore del mouse si trova nell'area della scala della penna del specificata.

Parameter	Description
IpszPenName As String	Nome della penna

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    Debug.Print
    objTrend.IsCursorPosInScaleArea(objTrend.GetPenNameFromList(0))
End Sub
Public Sub SymbolLoading()
    Set                                objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface =
End Sub
```

IsCursorPosInTimeArea, TrendCmdTarget Function

Syntax IsCursorPosInTimeArea

Description Questa funzione restituisce il valore booleano True se il cursore del mouse si trova nell'area orario del Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.IsCursorPosInTimeArea
End Sub
Public Sub SymbolLoading()
```

```

Set objTrend = GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

IsCursorPosInTrendArea, TrendCmdTarget Function

Syntax IsCursorPosInTrendArea

Description Questa funzione restituisce il valore booleano True se il cursore del mouse si trova nell'area della finestra del Trend.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.IsCursorPosInTrendArea
End Sub
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

LinkToDataLogger, TrendCmdTarget Function

Syntax LinkToDataLogger(_bClean)

Description Questa funzione esegue un aggiornamento delle penne dell'oggetto Trend in base al Data Logger specificato nelle proprietà del Trend o mediante la funzione "LinkedDataLogger". Il parametro booleano "bClean" consente di specificare se eliminare o meno le penne preesistenti nel Trend.



Nel caso dell'oggetto Trend la funzione "LinkToDataLogger()" è applicabile solo se la proprietà "Leggi Dati in Background" (proprietà del gruppo "Esecuzione") è disabilitata.

Parameter	Description
bClean As Boolean	Abilitazione alla eliminazione delle penne preesistenti nel Trend.

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget

```

```

Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.LinkedDataLogger = "Trend01"
    objTrend.Recording = False
    Debug.Print objTrend.LinkToDataLogger(True)
    objTrend.Recording = True
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

LoadExtSettings, TrendCmdTarget Function

Syntax LoadExtSettings

Description Questa funzione permette di caricare la configurazione dell'oggetto Trend dal file relativo. Il file di configurazione può essere specificato nelle proprietà relativa del Trend oppure mediante la proprietà "ExtSettingsFile". Prima di eseguire la funzione il Trend deve essere messo in pausa.



Nel caso dell'oggetto Trend la funzione "LoadExtSettings" è applicabile solo se la proprietà "Leggi Dati in Background" (proprietà del gruppo "Esecuzione") è disabilitata.

Inoltre è necessario salvare i file di Configurazione con la proprietà "Leggi Dati in Background" già disabilitata altrimenti potrebbero non essere caricati a runtime dall'oggetto Trend.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.ExtSettingsFile = "Test.TSXML"
    objTrend.Recording = False
    objTrend.LoadExtSettings
    objTrend.Recording = True
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

MoveCursorToMousePos, TrendCmdTarget Function

Syntax MoveCursorToMousePos()

Description Questa funzione consente di spostare il cursore del Trend nella posizione del puntatore del mouse quando il Trend è in stop.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If IsCursorPosInTrendArea Then
        MoveCursorToMousePos
    End If
End Sub
```

PageEnd, TrendCmdTarget Function

Syntax PageEnd

Description Questa funzione esegue lo scorrimento dei valori visualizzati nella finestra del Trend fino a mostrare la pagina contenente i dati più recenti. Tale funzione è operativa solo se il Trend è in pausa.

Parameter	Description
None	None

Result None

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.Freezed = True
    objTrend.PageEnd
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

PageNext, TrendCmdTarget Function

Syntax PageNext

Description Questa funzione esegue lo scorrimento dei valori visualizzati nella finestra del Trend fino a mostrare la pagina successiva a quella attualmente visualizzata. Per pagina successiva si intende quella contenente dati immediatamente più recenti rispetto ai dati visualizzati. Tale funzione è operativa solo se il Trend è in pausa.

Parameter	Description
None	None

Result None

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.Freezed = True
    objTrend.PageNext
End Sub
Public Sub SymbolLoading()
    Set                                objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface =
End Sub
```

PagePrev, TrendCmdTarget Function

Syntax PagePrev

Description Questa funzione esegue lo scorrimento dei valori visualizzati nella finestra del Trend fino a mostrare la pagina precedente a quella attualmente visualizzata. Per pagina precedente si intende quella contenente dati immediatamente più vecchi in ordine di tempo rispetto ai dati visualizzati. Tale funzione è operativa solo se il Trend è in pausa.

Parameter	Description
None	None

Result None

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.Freezed = True
    objTrend.PagePrev
End Sub
Public Sub SymbolLoading()
    Set                                objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface =
End Sub
```

PageStart, TrendCmdTarget Function

Syntax PageStart

Description Questa funzione esegue lo scorrimento dei valori visualizzati nella finestra del Trend fino a mostrare la pagina contenente i dati più vecchi in ordine di tempo. Tale funzione è operativa solo se il Trend è in pausa.

Parameter	Description
None	None

Result None

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.Freezed = True
    objTrend.PageStart
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

PrintTrend, TrendCmdTarget Function

Syntax PrintTrend(_bDefaultPrinter, _bPrintBackground, _nStartPage, _nEndPage, _lpszTitle)

Description Questa funzione permette la stampa di un intervallo di valori del Trend calcolato in pagine. Le caratteristiche di stampa sono definite dal driver Windows, per permetterne la modifica run-time si può disabilitare l'utilizzo della stampante di default ponendo a False il primo parametro (bDefaultPrinter).



La funzione non è supportata su Windows CE. Per eseguire la stampa di un sinottico utilizzare il tool "PrnWndCE.exe".



Utilizzando l'oggetto Data Analysis verrà sempre stampata la pagina corrente, in quanto di fatto il Data Analysis carica in memoria solo la pagina in visualizzazione.

Parameter	Description
bDefaultPrinter As Boolean	Utilizzo della stampante predefinita.
bPrintBackground As Boolean	Stampa in background.
nStartPage As Integer	Numero di pagina iniziale dell'intervallo di stampa.
nEndPage As Integer	Numero di pagina finale dell'intervallo di stampa.
lpszTitle As String	Titolo da scrivere in testa alla stampa.

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
```



```

Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.PrintTrend(True, True, objTrend.Page, objTrend.Page,
    objTrend.LinkedData Logger)
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

Refresh, TrendCmdTarget Function

Syntax Refresh

Description Questa funzione esegue un'aggiornamento grafico dell'oggetto Trend. E' necessario eseguire tale funzione dopo che è stata modificata una proprietà il cui impatto grafico sia particolarmente consistente.

Parameter	Description
None	None

Result None

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

RemovePen, TrendCmdTarget Function

Syntax RemovePen(_IpszName)

Description Questa funzione consente di eliminare una penna dal Trend durante l'esecuzione del progetto. Nel momento in cui si esegue tale funzione è necessario sospendere momentaneamente la registrazione impostando la proprietà Recording a False. Può essere necessario inoltre eseguire anche la funzione Refresh dopo la rimozione della penna per aggiornare graficamente l'oggetto Trend stesso.

Parameter	Description
IpszName As String	Nome della penna

Result None

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.Recording = False
    objTrend.RemovePen(objTrend.GetPenNameFromList(0))
    objTrend.Recording = True
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend                                     =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

Requery, TrendCmdTarget Function

Syntax Requery

Description Questa funzione esegue la query di modifica o di estrazione (campi "Filtro" e "Ordina"). E' necessario utilizzare tale funzione ogni volta che vengono modificate le proprietà "DataDefaultQuery", " DataFilterBy" e "DataSortBy".

Parameter	Description
None	None

Result None

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.DataFilterBy = objTrend.GetDateTimeColumnName() & " >= " & (Data - 1)
    objTrend.DataSortBy = objTrend.GetDateTimeColumnName() & " DESC"
    objTrend.Requery
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend                                     =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

ResetZoom, TrendCmdTarget Function

Syntax ResetZoom

Description Questa funzione annulla la modalità Zoom dell'oggetto Trend. Una volta eseguita, il cursore del mouse, se tracciato nell'area Trend, non seleziona più l'area di cui eseguire lo zoom.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    If objTrend Is Nothing Then Exit Sub
    objTrend.ResetZoom
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

RestartStatistic, TrendCmdTarget Function

Syntax RestartStatistic(_IpszPenName)

Description Questa funzione reinizializza i calcoli statistici di minimo, massimo e media per i valori della penna specificata.

Parameter	Description
IpszPenName As String	Nome della penna

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    If objTrend Is Nothing Then Exit Sub
    objTrend.RestartStatistic(objTrend.GetPenNameFromList(0))
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

SaveAllCurrentValue, TrendCmdTarget Function

Syntax SaveAllCurrentValue()

Description Salva sull'area Trend i valori di tutte le curve. I valori salvati sono visualizzati sulla pagina Trend con delle linee tratteggiate fino a quando non è richiamata la funzione ClearAllSavedValues. Tale funzione ha effetto solamente se per le penne è stata abilitata la proprietà ShowSavedValues, in caso contrario la funzione non ha effetto.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Dim i As Integer
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    For i= 0 to objTrend.GetPensNumber-1 Step 1
        objTrend.ShowSavedValues(objTrend.GetPenNameFromList(i)) = True
    Next i
    objTrend.SaveAllCurrentValue
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

SaveCurrentValue, TrendCmdTarget Function

Syntax SaveCurrentValue(_IpszPenName)

Description Salva sull'area Trend i valori della curva specificata con il parametro IpszPenName. I valori salvati sono visualizzati sulla pagina Trend con una linea tratteggiata fino a quando non è richiamata la funzione ClearSavedValues. Tale funzione ha effetto solamente se per la penna è stata abilitata la proprietà ShowSavedValues, in caso contrario la funzione non ha effetto.

Parameter	Description
IpszPenName As String	Nome della penna di cui salvare i valori

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.ShowSavedValues(objTrend.GetPenNameFromList(0)) = True
    objTrend.SaveCurrentValue(objTrend.GetPenNameFromList(0))
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

SaveExtSettings, TrendCmdTarget Function

Syntax SaveExtSettings

Description Questa funzione permette di salvare la configurazione dell'oggetto Trend nel file relativo. Il file di configurazione può essere specificato nelle proprietà relativa del Trend oppure mediante la proprietà "ExtSettingsFile".

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.ExtSettingsFile = "Test.TSXML"
    objTrend.SaveExtSettings
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

ScrollPosNext, TrendCmdTarget Function

Syntax ScrollPosNext(_IStep)

Description Questa funzione esegue lo scorrimento dei valori del Trend, se questo è in pausa, del numero di posizioni specificate nella direzione dei dati più recenti.

Parameter	Description
IStep As Long	Numero di posizioni da scorrere

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.ScrollPosNext(4)
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

ScrollPosPrev, TrendCmdTarget Function

Syntax ScrollPosPrev(_IStep)

Description Questa funzione esegue lo scorrimento dei valori del Trend, se questo è in pausa, del numero di posizioni specificate nella direzione dei dati più remoti.

Parameter	Description
IStep As Long	Numero di posizioni da scorrere

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.ScrollPosNext(4)
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

SetSamplesValue, TrendCmdTarget Function

Syntax SetSamplesValue(IpszPenName, nStart, nCount, pBuffer)

Description Questa funzione consente di forzare il valore di una serie di campionamenti consecutivi di un specifica penna. Accetta un parametro di tipo String per specificare la penna a cui fare riferimento, due parametri di tipo Long per indicare la posizione iniziale nel buffer e il numero di campionamenti da modificare e un Variant a cui passare un array di Double contenente i nuovi valori per i campionamenti. Attenzione: il numero di campionamenti da modificare deve essere uguale al numero di elementi dell'array contenente i nuovi dati.



Questa funzione non è gestita dall'oggetto "Data Analysis".

Parameter	Description
IpszPenName As String	Nome della penna
nStart As Long	Numero del campionamento iniziale
nCount As Long	Numero complessivo campionamenti
pBuffer As Variant	Array di double contenente i valori

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim i As Integer
    Dim value(50) As Double

    For i = 0 To 50
        value(i) = i
    Next i

    Dim IpszPenName As String
    Dim nStart As Long
    Dim nCount As Long

    IpszPenName = "Curva1"
    nStart = 0
    nCount = 51
```

```

        objTrend.SetSamplesValue(IpszPenName, nStart, nCount, value)
End Sub
Public Sub SymbolLoading()
    Set                                objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
=

```

SetTimeFontOrientation, TrendCmdTarget Function

Syntax SetTimeFontOrientation(_nNewValue)

Description Questa funzione imposta l'orientamento in gradi della scala dei tempi del Trend. Tale valore può andare da 0 a 45.

Parameter	Description
nNewValue As Integer	Valore dell'orientamento

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim i As Integer
    Public Sub Click()
        i = i+1
        If i=5 Then i=0
        objTrend.SetTimeFontOrientation(i)
End Sub
Public Sub SymbolLoading()
Set objTrend = GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

StartPanMode, TrendCmdTarget Function

Syntax StartPanMode

Description Questa funzione, se attivata, consente di scrollare l'area del Trend muovendo semplicemente il mouse. Questo metodo controlla che il mouse sia nell' area del Trend per cominciare a fare il pan. Questa funzionalità avrà effetto solo se il Trend è in stop. Accetta un valore Boolean.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()

```

```

        objTrend.StartPanMode
End Sub
Public Sub SymbolLoading()
    Set                                objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

StartZoomMode, TrendCmdTarget Function

Syntax StartZoomMode

Description Questa funzione inizializza la selezione dell'area Trend per la visualizzazione in modalità zoom. Può essere attivata solo se il Trend è in pausa. Chiamando questa funzione e tracciando una porzione dell'area Trend mediante il mouse, al rilascio del pulsante sinistro del mouse, l'area selezionata viene ingrandita fino ad occupare tutta la finestra di Trend.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.StartZoomMode
End Sub
Public Sub SymbolLoading()
    Set                                objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

Properties

AllBtnText, TrendCmdTarget Property

Syntax AllBtnText = _String

Description Questa proprietà imposta o restituisce il testo che dovrà apparire sul "Pulsante tutti" della barra dei pulsanti per la selezione dell'intervallo di tempo dell'oggetto Data Analysis. Se non viene specificato nulla, Movicon utilizzerà il testo di default.

Parameter	Description
None	None

Result String

Example:

```

Dim objDataAnalysis As TrendCmdTarget
Public Sub Click()
    If Not objDataAnalysis Is Nothing Then

```



```

        MsgBox "objDataAnalysis's AllBtnText is " &
objDataAnalysis.AllBtnText,vbInformation,GetProjectTitle
objDataAnalysis.AllBtnText = "All"
objDataAnalysis.Refresh
    Else
        MsgBox "objDataAnalysis is nothing",vbExclamation,GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objDataAnalysis =
GetSynopticObject.GetSubObject("DataAnalysis").GetObjectInterface
End Sub

```

BackupLink, TrendCmdTarget Property

Syntax BackupLink = _String

Description Questa proprietà permette di impostare la connessione ODBC associata al Trend. Questa proprietà è utile nel caso in cui si vogliano visualizzare dati archiviati in altri file.



Questa proprietà non è supportata in Windows CE.(Se impostata restituisce sempre una stringa vuota)

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    Dim ChartWnd As ChartWndCmdTarget
    Dim sConnectionString As String

    sConnectionString = "MyProject__BackupLink" 'DSN name

    If Not objTrend Is Nothing Then
        objTrend.BackupLink = sConnectionString
        objTrend.Requery
        objTrend.Refresh
    End If
    Set objTrend = Nothing
End Sub

```

BorderLegend, TrendCmdTarget Property

Syntax BorderLegend = _Boolean

Description Questa proprietà abilita o disabilita il bordo nell'area legenda del Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.BorderLegend = Not objTrend.BorderLegend
    objTrend.Refresh
End Sub
```

BorderLegendRaised, TrendCmdTarget Property

Syntax BorderLegendRaised = _Boolean

Description Questa proprietà imposta o restituisce il tipo di bordo per l'area legenda del Trend. Se impostata al valore booleano True il bordo è rialzato.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.BorderLegendRaised = Not objTrend.BorderLegendRaised
    objTrend.Refresh
End Sub
```

BorderPen, TrendCmdTarget Property

Syntax BorderPen = _Boolean

Description Questa proprietà abilita o disabilita il bordo nell'area penne del Trend.

Parameter	Description
-----------	-------------

None	None
------	------

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.BorderPen = Not objTrend.BorderPen
    objTrend.Refresh
End Sub
```

BorderPenRaised, TrendCmdTarget Property

Syntax BorderPenRaised = _Boolean

Description Questa proprietà imposta o restituisce il tipo di bordo per l'area penne del Trend. Se impostata al valore booleano True il bordo è rialzato.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.BorderPenRaised = Not objTrend.BorderPenRaised
    objTrend.Refresh
End Sub
```

BorderTime, TrendCmdTarget Property

Syntax BorderTime = _Boolean

Description Questa proprietà abilita o disabilita il bordo nell'area tempi del Trend.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.BorderTime = Not objTrend.BorderTime
    objTrend.Refresh
End Sub

```

BorderTimeRaised, TrendCmdTarget Property

Syntax BorderTimeRaised = _Boolean

Description Questa proprietà imposta o restituisce il tipo di bordo per l'area tempi del Trend. Se impostata al valore booleano True il bordo è rialzato.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.BorderTimeRaised = Not objTrend.BorderTimeRaised
    objTrend.Refresh
End Sub

```

BorderTrend, TrendCmdTarget Property

Syntax BorderTrend = _Boolean

Description Questa proprietà abilita o disabilita il bordo nell'area grafico del Trend.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface

```

```

End Sub
Public Sub Click()
    objTrend.BorderTrend = Not objTrend.BorderTrend
    objTrend.Refresh
End Sub

```

BorderTrendRaised, TrendCmdTarget Property

Syntax BorderTrendRaised = _Boolean

Description Questa proprietà imposta o restituisce il tipo di bordo per l'area grafico del Trend. Se impostata al valore booleano True il bordo è rialzato.

Parameter	Description
None	None

Result Boolean

Example:
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
 Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
 objTrend.**BorderTrendRaised** = Not objTrend.**BorderTrendRaised**
 objTrend.Refresh
End Sub

BrushColor, TrendCmdTarget Property

Syntax BrushColor = _Long

Description Questa proprietà imposta o restituisce il colore associato al pennello (cioè l'area di sfondo) della penna indicata. Accetta un valore di tipo Long.

Parameter	Description
None	None

Result Long

Example:
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
 Dim strVar As String
 If objTrend Is Nothing Then Exit Sub
 objTrend.**BrushColor**(objTrend.GetPenNameFromList(0)) = RGB(12,56,68)
 objTrend.Refresh
End Sub

```

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

ButtonPos, TrendCmdTarget Property

Syntax ButtonPos = _Integer

Description Questa impostazione restituisce la posizione in cui dovranno apparire i pulsanti sull'oggetto Trend.

Le possibilità sono:
 0 = sinistra
 1 = sopra
 2 = destra
 3 = sotto

Parameter	Description
None	None

Result Integer

Example:

```

Dim objTrend As TrendCmdTarget
Public Sub Click()
    If Not objTrend Is Nothing Then
        MsgBox "objTrend 's ButtonSize is " & objTrend.ButtonSize, vbInformation, GetProjectTitle
        objTrend.ButtonSize = 1
        objTrend.Refresh
    Else
        MsgBox "objTrend is nothing", vbExclamation, GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub

```

ButtonSize, TrendCmdTarget Property

Syntax ButtonSize = _Integer

Description Questa impostazione restituisce la dimensione dei pulsanti che verranno poi visualizzati sull'oggetto Trend.

Le possibilità sono:
 0 = piccolo
 1 = medio
 2 = grande

Parameter	Description
None	None

Result Integer

Example:

```
Dim objTrend As TrendCmdTarget
Public Sub Click()
    If Not objTrend Is Nothing Then
        MsgBox "objTrend's ButtonSize is " &
            objTrend.ButtonSize, vbInformation, GetProjectTitle
        objTrend.ButtonSize = 2
        objTrend.Refresh
    Else
        MsgBox "objTrend is nothing", vbExclamation, GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub
```

ColumnSeparator, TrendCmdTarget Property

Syntax ColumnSeparator = _Integer

Description Questa proprietà imposta o restituisce il codice ASCII del carattere usato come separatore tra le colonne nel file di registrazione associato al Trend.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.ColumnSeparator = Asc(",")
End Sub
Public Sub SymbolLoading()
    Set objTrend =
        GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

CompareTimeFrameBtnColor, TrendCmdTarget Property

Syntax CompareTimeFrameBtnColor = _Long

Description Questa proprietà consente di leggere ed impostare il colore usato per visualizzare i pulsanti per la selezione dell'intervallo di rappresentazione dei dati di comparazione per l'oggetto Data Analysis. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Refresh" per aggiornare graficamente l'oggetto.

Parameter	Description
None	None

Result Long

Example:

```
Dim obj As TrendCmdTarget
Public Sub Click()
    Dim IColor As Long

    If obj Is Nothing Then Set obj =
        GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
    If ChooseColor(IColor) Then
        obj.CompareTimeFrameBtnColor = IColor
        obj.Refresh
    End If
End Sub
```

ComposedFileName, TrendCmdTarget Property

Syntax ComposedFileName = _String

Description Questa proprietà restituisce, in formato stringa, il nome e il percorso del file di registrazione associato all'oggetto Trend. Questa proprietà è di sola lettura.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.ComposedFileName
End Sub
Public Sub SymbolLoading()
    Set objTrend =
        GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

CompressData, TrendCmdTarget Property

Syntax CompressData = _String

Description Questa proprietà imposta o restituisce la modalità di lettura dei dati dal Data Logger. Impostando la proprietà al valore booleano a True i dati letti verranno compressi in modo da essere visualizzati possibilmente in un'unica pagina Trend.

Se il numero di records letti supera l'impostazione del numero di campi da visualizzare nella pagina Trend, viene eseguita in automatico un'approssimazione del grafico in modo da visualizzare il tutto su una o al massimo due pagine Trend.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.CompressData = True
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

CurrentMultiplier, TrendCmdTarget Property

Syntax CurrentMultiplier = _Byte

Description Questa proprietà consente di leggere o impostare il valore del Fattore di Moltiplicazione per i pulsanti di selezione dell'intervallo di data. Il valore massimo impostabile è 100, il valore minimo è 1.



Questa proprietà è gestita soltanto dall'oggetto Data Analysis e non dal Trend.

Parameter	Description
None	None

Result Byte

Example:

```
Option Explicit
Dim objDataAnalysis As TrendCmdTarget
Public Sub Click()
    If Not objDataAnalysis Is Nothing Then
        objDataAnalysis.CurrentMultiplier = VAR00001
    End If
End Sub
Public Sub SymbolLoading()
    Set objDataAnalysis =
    GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
End Sub
```

CurrentSelectedPen, TrendCmdTarget Property

Syntax CurrentSelectedPen = _Long

Description Questa proprietà consente di leggere e impostare la penna selezionata nella legenda. Il valore di ritorno corrisponde a "-1" nel caso in cui non sia stata selezionata nessuna penna in legenda. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Refresh" per aggiornare graficamente l'oggetto.

Parameter	Description
None	None

Result Long

Example:

```
Dim obj As TrendCmdTarget
Public Sub Click()
    If obj Is Nothing Then Set obj =
        GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    If obj.CurrentSelectedPen >= 0 Then
        obj.EditPenProperties(obj.GetPenNameFromList(obj.CurrentSelectedPe
n))
    Else
        MsgBox "Plese, select a pen before!", vbInformation + vbOkOnly,
            GetProjectTitle
    End If
End Sub
```

CurrentTopPen, TrendCmdTarget Property

Syntax CurrentTopPen = _Long

Description Questa proprietà consente di leggere e impostare la prima penna visualizzata in legenda. Questo consente di scrollare le penne in legenda, quando il loro numero è maggiore del numero massimo di penne visualizzabili contemporaneamente (vedi proprietà "MaxLegendVisiblePen"). Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Refresh" per aggiornare graficamente l'oggetto.

Parameter	Description
None	None

Result Long

Example:

```
Dim obj As TrendCmdTarget
Public Sub Click()
    If obj Is Nothing Then Set obj =
        GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    obj.CurrentTopPen = obj.CurrentTopPen + 1
    obj.Refresh
End Sub
```

CursorPos, TrendCmdTarget Property

Syntax CursorPos = _Long

Description Questa proprietà imposta o restituisce la posizione del cursore del Trend nel buffer dei valori memorizzati quando il Trend è in pausa.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    If objTrend Is Nothing Then Exit Sub
    objTrend.CursorPos = 100
End Sub
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

DataDefaultQuery, TrendCmdTarget Property

Syntax DataDefaultQuery = _String

Description La proprietà imposta o restituisce una query SQL personalizzata di aggiornamento per i dati del Data Logger associato all'oggetto Trend o Data Analysis. Una volta definita la query nella proprietà è necessario invocare la funzione "Requery" per poterla eseguire tramite l'ODBC impostato nel Data Logger di riferimento impostato nell'oggetto Trend o Data Analysis. Per il corretto risultato della query di select personalizzata è necessario che l'ordine delle Colonne specificate rispetti l'ordine della lista delle Penne configurate nell'oggetto Trend o nell'oggetto Data Analysis. Ad esempio se la lista Penne è "Col01, Col02, Col03" riferiti al Data Logger di nome "myDataLogger" la query personalizzata potrebbe essere del tipo:
SELECT LocalCol, Col01, Col02, Col03 FROM myDataLogger WHERE LocalCol >= {ts '2017-06-14 00:00:00.000'} AND LocalCol < {ts '2017-06-15 00:00:01.000'} ORDER BY TimeCol DESC
In ogni caso è possibile eseguire query complesse come ad esempio la seguente query che recupera un recorset di massimo 10.000 valori dove ogni valore rappresenta la media dei valori registrati in un'ora:

```
DataDefaultQuery = "SELECT TOP 10000 0 As MSecCol, MIN(LocalCol) As LocalCol, AVG(Cosine) As Cosine, AVG(Ramp) As Ramp FROM Data Logger GROUP BY DatePart (dayofyear, LocalCol ), DatePart (Hour, LocalCol ) ORDER BY LocalCol DESC"
```



I valori delle proprietà "DataFilterBy" e "DataSortBy" vengono utilizzati solo nel caso in cui la proprietà "DataDefaultQuery" non è stata impostata.



Nel caso si utilizzi la proprietà "DataDefaultQuery" per il Data Analysis le proprietà "DateFrom" e "DateTo" vengono ignorate.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Begin Dialog UserDialog 370,154,"TrendCmdTarget" ' %GRID:10,7,1,1
    GroupBox 20,7,340,84,"DataDefaultQuery",.GroupBox1
    TextBox 100,28,250,56,.Query,1
    Text 30,28,60,21,"Query",.Text1
    OKButton 20,105,160,42
    CancelButton 190,105,160,42
    End Dialog
    Dim dlg As UserDialog
    dlg.Query = "DELETE FROM Log1sec"
    If Dialog(dlg) <> -1 Then Exit Sub
    objTrend .DataDefaultQuery = dlg.Query
    objTrend .Requery
End Sub

Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

DateFromCompareCurrent, TrendCmdTarget Property

Syntax dateFromCompareCurrent()

Description Questa proprietà è di sola lettura e consente di recuperare la data iniziale dell'asse dei tempi usata per la comparazione.



Questa proprietà è gestita soltanto dall'oggetto Data Analysis e non dal Trend

Parameter	Description
None	None

Result Date

Example:

```
Option Explicit
Dim objDataAnalysis As TrendCmdTarget
Public Sub SymbolLoading()
    Set objDataAnalysis = GetSynopticObject.GetSubObject
    ("objDataAnalysis").GetObjectInterface
End Sub

Public Sub Click()
    MsgBox "dateFromCompareCurrent = " & CStr
    (objDataAnalysis.dateFromCompareCurrent), vbInformation, GetProjectTitle
End Sub
```

DateFromCurrent, TrendCmdTarget Property

Syntax `dateToCurrent()`

Description	Questa proprietà è di sola lettura e consente di recuperare la data finale dell'asse dei tempi per l'intervallo visualizzato.
--------------------	---



Questa proprietà è gestita soltanto dall'oggetto Data Analysis e non dal Trend

Parameter	Description
None	None

Result	Date
--------	------

Example:

Example:
Option Explicit
Dim objDataAnalysis As TrendCmdTarget

```
Public Sub SymbolLoading()  
    Set objDataAnalysis = GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface  
End Sub
```

```
Public Sub Click()  
    MsgBox "dateToCurrent = " & CStr(objDataAnalysis.dateToCurrent),  
    vbInformation, GetProjectTitle  
End Sub
```

DataFileName, TrendCmdTarget Property

Syntax DataFileName = String

Description	Questa proprietà imposta o restituisce il nome del file di dati associato al Trend. Nel caso in cui si desideri modificare il nome di tale file, è necessario sospendere momentaneamente la registrazione dei dati impostando la proprietà Recording a False. L'estensione del file è obbligatoriamente ".CSV", in caso contrario il sistema provvederà a forzare tale impostazione.
--------------------	--

Parameter	Description
None	None

Result	String
---------------	--------

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.Recording = False
    objTrend.DataFileName = "Prova"
    objTrend.Recording = True
End Sub
```

```

End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

DataFilterBy, TrendCmdTarget Property

Syntax DataFilterBy= _String

Description Questa proprietà imposta o restituisce il campo "Filtro" per l'estrazione dei dati dal Data Logger associato all'oggetto Trend. Il campo "Filtro" rispetta la sintassi SQL e corrisponde alla clausola "WHERE". Tale clausola viene utilizzata solo al momento dell'accesso al database cioè all'avvio del progetto o al passaggio del Trend dallo stato di run a quello di stop. Quando il Trend non è in pausa non viene tenuto in considerazione il filtro per la visualizzazione dei valori.
Nel Data Analysis il valore della proprietà "DataFilterBy" è utilizzato in "AND" sulla clausola WHERE che il Data Analysis ha composto per filtrare i dati. Questo consente di applicare un ulteriore filtro anche quando vengono usati i pulsanti di intervallo data, i pulsanti di scorrimento o viene impostato un filtro per data.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.DataFilterBy = objTrend.GetDateTimeColumnName() & " >= " & (Data
-1)
    objTrend.DataSortBy = objTrend.GetDateTimeColumnName() & " DESC"
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

DataSortBy, TrendCmdTarget Property

Syntax DataSortBy = _String

Description Questa proprietà imposta o restituisce il campo "Ordina" per l'estrazione dei dati dal Data Logger associato all'oggetto Trend. Il campo "Ordina" rispetta la sintassi SQL e corrisponde alla clausola "ORDER BY". Tale clausola viene utilizzata solo al momento dell'accesso al database cioè all'avvio del progetto o al passaggio del Trend dallo stato di run a quello di stop. Quando il Trend non è in pausa non viene tenuto in considerazione l'ordinamento impostato per la visualizzazione dei valori. Occorre tenere presente che il primo record dei valori del Trend è il più recente in ordine di tempo.
Nel Data Analysis il valore della proprietà "DataSortBy" viene sostituito a quello di default "TimeCol DESC".

Parameter	Description
-----------	-------------

None	None
------	------

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.DataFilterBy = objTrend.GetDateTimeColumnName() & " >= " & (Data - 1)
    objTrend.DataSortBy = objTrend.GetDateTimeColumnName() & " DESC"
    objTrend.Requery
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

dateFrom, TrendCmdTarget Property

Syntax dateFrom = _Date

Description

Questa proprietà è utilizzabile soltanto nell'oggetto "Data Analysis" e consente di impostare la data iniziale con cui filtrare i dati. Può anche essere usata in lettura per recuperare l'ultima data impostata. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Requery" per aggiornare e ricaricare i dati dal database. Impostando "dateFrom" e "dateTo" al valore numerico "0", la curva verrà filtrata usando l'attuale selezione dell'intervallo di rappresentazione dei dati.

Parameter	Description
None	None

Result Date

Example:

```
Dim obj As TrendCmdTarget
Public Sub Click()
    If obj Is Nothing Then Set obj = GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
    obj.dateTo = Now
    obj.dateFrom = DateAdd("n", -10, obj.dateTo)
    obj.Requery
End Sub
```

dateFromCompare, TrendCmdTarget Property

Syntax dateFromCompare = _Date

Description

Questa proprietà è utilizzabile soltanto nell'oggetto "Data Analysis" e consente di impostare la data iniziale con cui filtrare i dati per la curva di comparazione. Può anche essere usata in lettura per recuperare l'ultima data impostata. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Requery" per aggiornare e ricaricare i dati dal database. Impostando "dateFromCompare" e "dateToCompare" al valore numerico "0", la

curva verrà filtrata usando l'attuale selezione dell'intervallo di rappresentazione dei dati di comparazione.

Parameter	Description
None	None

Result Date

Example:

```
Dim obj As TrendCmdTarget
Public Sub Click()
    If obj Is Nothing Then Set obj =
        GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
    ' Compare one minute before of the date and time on analysis
    obj.dateToCompare = DateAdd("n", -1, obj.dateTo)
    obj.dateFromCompare = DateAdd("n", -1, obj.dateFrom)
    obj.Requery
End Sub
```

dateTo, TrendCmdTarget Property

Syntax dateTo = _Date

Description Questa proprietà è utilizzabile soltanto nell'oggetto "DataAnalysis" e consente di impostare la data finale con cui filtrare i dati. Può anche essere usata in lettura per recuperare l'ultima data impostata. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Requery" per aggiornare e ricaricare i dati dal database. Impostando "dateFrom" e "dateTo" al valore numerico "0", la curva verrà filtrata usando l'attuale selezione dell'intervallo di rappresentazione dei dati.

Parameter	Description
None	None

Result Date

Example:

```
Dim obj As TrendCmdTarget
Public Sub Click()
    If obj Is Nothing Then Set obj =
        GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
    obj.dateTo = Now
    obj.dateFrom = DateAdd("n", -10, obj.dateTo)
    obj.Requery
End Sub
```

DateToCompareCurrent, TrendCmdTarget Property

Syntax dateToCompareCurrent()

Description Questa proprietà è di sola lettura e consente di recuperare la data finale dell'asse dei tempi usata per la comparazione.



Questa proprietà è gestita soltanto dall'oggetto Data Analysis e non dal Trend

Parameter	Description
None	None

Result Date

Example:

Option Explicit

Dim objDataAnalysis As TrendCmdTarget

Public Sub SymbolLoading()

 Set objDataAnalysis = GetSynopticObject.GetSubObject

 ("objDataAnalysis").GetObjectInterface

End Sub

Public Sub Click()

 MsgBox "dateToCompareCurrent = "

 CStr(objDataAnalysis.dateToCompareCurrent),

 vbInformation, GetProjectTitle

End Sub

DateToCurrent, TrendCmdTarget Property

Syntax dateFromCurrent()

Description Questa proprietà è di sola lettura e consente di recuperare la data iniziale dell'asse dei tempi per l'intervallo visualizzato.



Questa proprietà è gestita soltanto dall'oggetto Data Analysis e non dal Trend

Parameter	Description
None	None

Result Date

Example:

Option Explicit

Dim objDataAnalysis As TrendCmdTarget

Public Sub SymbolLoading()

 Set objDataAnalysis =

 GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface

End Sub

Public Sub Click()

 MsgBox "dateFromCurrent = " & CStr(objDataAnalysis.dateFromCurrent),

 vbInformation, GetProjectTitle

End Sub

DayBtnText, TrendCmdTarget Property

Syntax DayBtnText = _String

Description Questa proprietà imposta o restituisce il testo che dovrà apparire sul "Pulsante Giorno" della barra dei pulsanti per la selezione dell'intervallo di tempo dell'oggetto Data Analysis. Se non viene specificato nulla, Movicon utilizzerà il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objDataAnalysis As TrendCmdTarget
Public Sub Click()
    If Not objDataAnalysis Is Nothing Then
        MsgBox "objDataAnalysis's DayBtnText is " &
            objDataAnalysis.DayBtnText, vbInformation, GetProjectTitle
        objDataAnalysis.DayBtnText = "Day"
        objDataAnalysis.Refresh
    Else
        MsgBox "objDataAnalysis is nothing", vbExclamation, GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objDataAnalysis =
        GetSynopticObject.GetSubObject("DataAnalysis").GetObjectInterface
End Sub
```

dateToCompare, TrendCmdTarget Property

Syntax dateToCompare = _Date

Description Questa proprietà è utilizzabile soltanto nell'oggetto "Data Analysis" e consente di impostare la data finale con cui filtrare i dati per la curva di comparazione. Può anche essere usata in lettura per recuperare l'ultima data impostata. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Requery" per aggiornare e ricaricare i dati dal database. Impostando "dateFromCompare" e "dateToCompare" al valore numerico "0", la curva verrà filtrata usando l'attuale selezione dell'intervallo di rappresentazione dei dati di comparazione.

Parameter	Description
None	None

Result Date

Example:

```
Dim obj As TrendCmdTarget
Public Sub Click()
    If obj Is Nothing Then Set obj =
        GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
    ' Compare one minute before of the date and time on analysis
    obj.dateToCompare = DateAdd("n", -1, obj.dateTo)
    obj.dateFromCompare = DateAdd("n", -1, obj.dateFrom)
```

```
obj.Requery  
End Sub
```

DrawGridAfter, TrendCmdTarget Property

Syntax DrawGridAfter = _Boolean

Description Abilitando questa proprietà la griglia del Trend passa in primo piano rispetto ai valori delle penne.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit  
Dim objTrend As TrendCmdTarget  
Public Sub Click()  
    objTrend.DrawGridAfter = Not objTrend.DrawGridAfter  
    objTrend.Refresh  
End Sub  
Public Sub SymbolLoading()  
    Set objTrend =  
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface  
End Sub
```

ExpandBtnText, TrendCmdTarget Property

Syntax ExpandBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando Espandi dell'oggetto Trend. Se non viene specificato nulla, Movicon utilizzerà il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objTrend As TrendCmdTarget  
Public Sub Click()  
    If Not objTrend Is Nothing Then  
        MsgBox "objTrend 's ExpandBtnText is " & objTrend.ExpandBtnText  
        ,vbInformation,GetProjectTitle  
        objTrend.ExpandBtnText = "Expand Text"  
        objTrend.Refresh  
    Else  
        MsgBox "objTrend is nothing",vbExclamation,GetProjectTitle  
    End If  
End Sub  
Public Sub SymbolLoading()
```

```

        Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub

```

ExtSettingsFile, TrendCmdTarget Property

Syntax ExtSettingsFile = _String

Description Questa proprietà imposta o restituisce il nome del file di configurazione associato al Trend. L'estensione prevista per il file è ".TSXML".

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.ExtSettingsFile = "Test.TSXML"
    objTrend.SaveExtSettings
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

FontHeightLegend, TrendCmdTarget Property

Syntax FontHeightLegend = _Integer

Description Questa proprietà imposta o restituisce l'altezza del font utilizzato nella visualizzazione della Legenda del Trend.

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.FontHeightLegend = 10
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

FontHeightScale, TrendCmdTarget Property

Syntax FontHeightScale = _Integer

Description Questa proprietà imposta o restituisce l'altezza del font utilizzato nella visualizzazione della scala del Trend.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.FontHeightScale = 10
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

FontHeightTime, TrendCmdTarget Property

Syntax FontHeightTime = _Integer

Description Questa proprietà imposta o restituisce l'altezza del font utilizzato nella visualizzazione dell'orario del Trend.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.FontHeightTime = 10
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

FontNameLegend, TrendCmdTarget Property

Syntax FontNameLegend = _String

Description Questa proprietà imposta o restituisce il nome del font utilizzato nella legenda del Trend.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.FontNameLegend = "MS Sans Serif"
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

FontNameScale, TrendCmdTarget Property

Syntax FontNameScale = _String

Description Questa proprietà imposta o restituisce il nome del font utilizzato nella visualizzazione della scala del Trend.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.FontNameScale = "MS Sans Serif"
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

FontNameTime, TrendCmdTarget Property

Syntax FontNameTime = _String

Description Questa proprietà imposta o restituisce il nome del font utilizzato nella visualizzazione dei tempi del Trend.

Parameter	Description
-----------	-------------

None	None
------	------

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.FontNameTime = "MS Sans Serif"
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

FormatTime, TrendCmdTarget Property

Syntax FormatTime = _String

Description Questa proprietà consente di formattare la data e ora con cui visualizzare le date nell'area del tempo del Trend/Data Analysis. La proprietà per visualizzare i millisecondi rimane invariata ed aggiunge in ogni caso i millisecondi. La proprietà per visualizzare la data invece non ha alcun effetto nel caso in cui si usi questa nuova proprietà. Quando la formattazione della data ed ora è personalizzata, questa verrà rappresentata su una riga soltanto. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Refresh" per aggiornare graficamente l'oggetto. Tutti i codici di formattazione utilizzabili in questa proprietà sono elencati nella proprietà "Formato Data Ora".

Parameter	Description
None	None

Result String

Example:

```
Dim obj As TrendCmdTarget
Public Sub Click()
    Dim sStyleFormat() As String
    Dim nItemSelected As Integer

    sStyleFormat = Split("%#x|%W|%a %b %Y", "|")
    nItemSelected = ShowPopupMenu(sStyleFormat)
    If nItemSelected >= 0 Then
        If obj Is Nothing Then Set obj =
        GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
        obj.FormatTime = sStyleFormat(nItemSelected)
        obj.Refresh
    End If
End Sub
```

Freezed, TrendCmdTarget Property

Syntax Freezed = _Boolean

Description Questa proprietà, se impostata al valore booleano True, manda in stop il Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.Freezed = Not objTrend.Freezed
End Sub
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetObjectInterface(
        GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
    )
End Sub
```

GeneralGap, TrendCmdTarget Property

Syntax GeneralGap = _Integer

Description Questa proprietà imposta o restituisce la spaziatura, espressa in pixel, dell'area perimetrale del Trend. Accetta un valore Integer compreso tra 1 e 25.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.GeneralGap = 10
    objTrend.Refresh
End Sub
```

HourBtnText, TrendCmdTarget Property

Syntax HourBtnText = _String

Description Questa proprietà imposta o restituisce il testo che dovrà apparire sul "Pulsante Ora" della barra dei pulsanti per la selezione dell'intervallo di tempo dell'oggetto Data Analysis. Se non viene specificato nulla, Movicon utilizzerà il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objDataAnalysis As TrendCmdTarget
Public Sub Click()
    If Not objDataAnalysis Is Nothing Then
        MsgBox "objDataAnalysis's HourBtnText is " &
            objDataAnalysis.HourBtnText, vbInformation, GetProjectTitle
        objDataAnalysis.HourBtnText = "Hour"
        objDataAnalysis.Refresh
    Else
        MsgBox "objDataAnalysis is nothing", vbExclamation, GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objDataAnalysis =
        GetSynopticObject.GetSubObject("DataAnalysis").GetObjectInterface
End Sub
```

HourRecTime, TrendCmdTarget Property

Syntax HourRecTime

Description Questa proprietà imposta o restituisce il numero di ore che compongono la proprietà "Periodo di campionamento".

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    objTrend.HourRecTime = 1
    objTrend.MinRecTime = 30
    objTrend.SecRecTime = 0
End Sub

Public Sub SymbolLoading()
    Set objTrend =
        GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

HourViewTime, TrendCmdTarget Property

Syntax HourViewTime

Description Questa proprietà imposta o restituisce il numero di ore impostate nella proprietà di visualizzazione del Trend.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.HourViewTime
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

InvertDrawDirection, TrendCmdTarget Property

Syntax InvertDrawDirection = _Boolean

Description Inverte la direzione del disegno del Grafico e della posizione delle Penne. L'applicazione della proprietà è successiva al Refresh del Trend



Questa proprietà non è disponibile per l'oggetto "Data Analysis"

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim oTrend As TrendCmdTarget
Public Sub Click()
    oTrend.InvertDrawDirection = Not(oTrend.InvertDrawDirection)
    oTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set oTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub
```

LegendBrushColor, TrendCmdTarget Property

Syntax LegendBrushColor = _Long

Description Questa proprietà imposta o restituisce il codice del colore di sfondo della legenda del Trend.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.LegendBrushColor = RGB(255,255,0)
End Sub
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

LegendBrushVisible, TrendCmdTarget Property

Syntax LegendBrushColor = _Boolean

Description Questa proprietà permette di abilitare o disabilitare la visualizzazione dello sfondo della legenda del Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.LegendBrushVisible = Not objTrend.LegendBrushVisible
End Sub
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

LinkedDataLogger, TrendCmdTarget Property

Syntax LinkedDataLogger = _String

Description Questa proprietà imposta o restituisce il nome del Data Logger collegato all'oggetto Trend. Attenzione però che una volta impostato un Data Logger differente con questa proprietà è necessario utilizzare la funzione "LinkToDataLogger" per far acquisire la modifica al Trend.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Debug.Print objTrend.LinkedDataLogger
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

MaxFileLength, TrendCmdTarget Property

Syntax MaxFileLength = _Long

Description Questa proprietà imposta o restituisce la lunghezza approssimativa (in Kb) del file di testo (CSV) associato al Trend. Il valore di default è 10Kb. Questa proprietà ha effetto soltanto se si utilizzata in combinazione con la proprietà "StartNewFile" o se comunque è stata abilitata l'opzione "Crea nuovo file" nella configurazione del Trend.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.StartNewFile = True
    objTrend.MaxFileLength = CInt(InputBox("Max file length"))
    objTrend.SaveExtSettings
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

MaxLegendVisiblePen, TrendCmdTarget Property

Syntax MaxLegendVisiblePen = _Long

Description Questa proprietà consente di leggere e impostare il numero massimo di penne visibili in legenda. Nel caso in cui il numero di penne impostate è maggiore del numero massimo di penne visibili, allora si attiveranno dei pulsanti di scroll per spostarsi all'interno della lista. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Refresh" per aggiornare graficamente l'oggetto.

Parameter	Description
None	None

Result Long

Example:

```
Dim obj As TrendCmdTarget
Public Sub Click()
    If obj Is Nothing Then Set obj =
    GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    MsgBox "obj.MaxLegendVisiblePen->" & obj.MaxLegendVisiblePen,
    vbInformation + vbOkOnly, GetProjectTitle
End Sub
```

MaxNumFiles, TrendCmdTarget Property

Syntax MaxNumFiles = _Long

Description Questa proprietà imposta o restituisce il numero di file di testo (CSV), associati al Trend, che verranno creati prima di riciclare. Il valore di default è 10. Questa proprietà ha effetto soltanto se utilizzata in combinazione con la proprietà "StartNewFile" o se comunque è stata abilitata "opzione "Crea nuovo file" nella configurazione del Trend.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.StartNewFile = True
    objTrend.MaxFileLength = CInt(InputBox("Max file length"))
    objTrend.MaxNumFiles = CInt(InputBox("Max num. file"))
    objTrend.SaveExtSettings
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

MeasureBtnText, TrendCmdTarget Property

Syntax MeasureBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando Misura dell'oggetto Data Analysis. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Refresh" per aggiornare graficamente l'oggetto.

Parameter	Description
None	None

Result String

Example:

```
Dim obj As TrendCmdTarget
Public Sub Click()
    If obj Is Nothing Then Set obj =
        GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
    If obj.MeasureBtnText <> "" Then
        obj.MeasureBtnText = ""
    Else
        obj.MeasureBtnText = "MEASURE"
    End If
    obj.Refresh
End Sub
```

MeasureTextColor, TrendCmdTarget Property

Syntax MeasureTextColor = _String

Description Questa proprietà imposta o restituisce il colore assegnato al testo visualizzato dalla misura effettuata tra i punti una penna.
Questa proprietà è gestita soltanto dall'oggetto Data Analysis e non dal Trend.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objDataAnalysis As TrendCmdTarget
Public Sub Click()

    If Not objDataAnalysis Is Nothing Then
        objDataAnalysis.MeasureTextColor = RGB(255,255,255)
    End If
End Sub

Public Sub SymbolLoading()
    Set objDataAnalysis =
        GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
End Sub
```

MinBtnText, TrendCmdTarget Property

Syntax MinBtnText = _String

Description Questa proprietà imposta o restituisce il testo che dovrà apparire sul "Pulsante Minuto" della barra dei pulsanti per la selezione dell'intervallo di tempo dell'oggetto Data Analysis. Se non viene specificato nulla, Movicon utilizzerà il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objDataAnalysis As TrendCmdTarget
Public Sub Click()
    If Not objDataAnalysis Is Nothing Then
        MsgBox "objDataAnalysis's MinBtnText is " &
            objDataAnalysis.MinBtnText, vbInformation, GetProjectTitle
        objDataAnalysis.MinBtnText = "Minutes"
        objDataAnalysis.Refresh
    Else
        MsgBox "objDataAnalysis is nothing", vbExclamation, GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objDataAnalysis =
        GetSynopticObject.GetSubObject("DataAnalysis").GetObjectInterface
End Sub
```

MinRecTime, TrendCmdTarget Property

Syntax MinRecTime = _Integer

Description Questa proprietà imposta o restituisce il numero di minuti che compongono la proprietà "Periodo di campionamento".

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.HourRecTime = 1
    objTrend.MinRecTime = 30
    objTrend.SecRecTime = 0
End Sub

Public Sub SymbolLoading()
    Set objTrend =
        GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

MinViewTime, TrendCmdTarget Property

Syntax MinViewTime

Description Questa proprietà imposta o restituisce il numero di minuti impostati nella proprietà di visualizzazione del Trend.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim strVar As String
    Dim pnX As Integer
    Dim pnY As Integer
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.MinViewTime
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

MonthBtnText, TrendCmdTarget Property

Syntax MonthBtnText = _String

Description Questa proprietà imposta o restituisce il testo che dovrà apparire sul "Pulsante Mese" della barra dei pulsanti per la selezione dell'intervallo di tempo dell'oggetto Data Analysis. Se non viene specificato nulla, Movicon utilizzerà il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objDataAnalysis As TrendCmdTarget
Public Sub Click()
    If Not objDataAnalysis Is Nothing Then
        MsgBox "objDataAnalysis's MonthBtnText is " &
        objDataAnalysis.MonthBtnText, vbInformation, GetProjectTitle
        objDataAnalysis.MonthBtnText = "Month"
        objDataAnalysis.Refresh
    Else
        MsgBox "objDataAnalysis is nothing", vbExclamation, GetProjectTitle
    End If
End Sub
```



```

Public Sub SymbolLoading()
    Set objDataAnalysis =
    GetSynopticObject.GetSubObject("DataAnalysis").GetObjectInterface
End Sub

```

MsecRecTime, TrendCmdTarget Property

Syntax MsecRecTime = _Integer

Description Questa proprietà imposta o restituisce il numero di minuti che compongono la proprietà "Periodo di campionamento".



Come si nota, Movicon permette tempi di campionamento anche pari al millisecondo, che tuttavia richiedono prestazioni impegnative alla CPU del PC e dovranno pertanto essere valutati con competenza dal programmatore.

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.MsecRecTime = 30
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

NetworkBackupServerName, TrendCmdTarget Property

Syntax NetworkBackupServerName = _String

Description Questa proprietà imposta o restituisce il nome dell'eventuale Server di Networking di Backup usato per prelevare i dati da visualizzare nel Trend o data Analysis quando il Server Primario, ovvero quello impostato nella proprietà "NetowrkServerName" è in timeout.



Per visualizzare i dati di un Server è necessario che il Data Logger sia presente anche sul progetto Client, al fine di poter recuperare la struttura del Database. Tuttavia sul progetto Client il Data Logger può essere creato soltanto come struttura, senza quindi dover associare alcuna variabile alle colonne.

Parameter	Description
None	None

Result String

Example:

```
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Debug.Print objTrend.NetworkBackupServerName
End Sub
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub
```

NetworkServerName, TrendCmdTarget Property

Syntax NetworkServerName = _String

Description Questa proprietà restituisce il nome dell'eventuale Server di Networking dal quale dovranno essere prelevati i dati da visualizzare nel Trend o data Analysis.

Parameter	Description
None	None

Result String

Example:

```
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Debug.Print objTrend.NetworkServerName
End Sub
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub
```

NextBtnText, TrendCmdTarget Property

Syntax NextBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando Prossimo dell'oggetto Trend. Se non viene specificato nulla, Movicon utilizzerà il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objTrend As TrendCmdTarget
Public Sub Click()
    If Not objTrend Is Nothing Then
        MsgBox "objTrend        's        NextBtnText        is        " &
            objTrend.NextBtnText,vbInformation,GetProjectTitle
        objTrend.NextBtnText = "Next"
```

```

        objTrend.Refresh
    Else
        MsgBox "objTrend is nothing",vbExclamation,GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub

```

NoneBtnText, TrendCmdTarget Property

Syntax NoneBtnText = _String

Description Questa proprietà imposta o restituisce il testo che dovrà apparire sul "Pulsante nessuno" della barra dei pulsanti per la selezione dell'intervallo di tempo di comparazione dell'oggetto Data Analysis. Se non viene specificato nulla, Movicon utilizzerà il testo di default.

Parameter	Description
None	None

Result String

Example:

```

Dim objDataAnalysis As TrendCmdTarget
Public Sub Click()
    If Not objDataAnalysis Is Nothing Then
        MsgBox "objDataAnalysis's NoneBtnText is " &
            objDataAnalysis.NoneBtnText,vbInformation,GetProjectTitle
        objDataAnalysis.NoneBtnText = "None"
        objDataAnalysis.Refresh
    Else
        MsgBox "objDataAnalysis is nothing",vbExclamation,GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objDataAnalysis =
        GetSynopticObject.GetSubObject("DataAnalysis").GetObjectInterface
End Sub

```

NumCacheRecordFile, TrendCmdTarget Property

Syntax NumCacheRecordFile = _Integer

Description Questa proprietà imposta o restituisce il numero di registrazioni che vengono mantenute nella memoria cache prima di eseguire un accesso al file associato all'oggetto Trend.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.NumCacheRecordFile = 10
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

NumXGridDivision, TrendCmdTarget Property

Syntax NumXGridDivision = _Integer

Description Questa proprietà imposta o restituisce il numero di divisioni maggiori per la griglia dell'asse X. Accetta un valore di tipo Integer.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.NumXGridDivision = 10
    objTrend.Refresh
End Sub
```

NumXMinorGridDivision, TrendCmdTarget Property

Syntax NumXMinorGridDivision = _Integer

Description Questa proprietà imposta o restituisce il numero di divisioni minori per la griglia dell'asse X. Accetta un valore di tipo Integer compreso tra 1 e 10.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
```

```

Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.NumXMinorGridDivision = 10
    objTrend.Refresh
End Sub

```

NumYGridDivision, TrendCmdTarget Property

Syntax NumYMinorGridDivision= _Integer

Description Questa proprietà imposta o restituisce il numero di divisioni maggiori per la griglia dell'asse Y. Accetta un valore di tipo Integer.

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.NumYGridDivision = 10
    objTrend.Refresh
End Sub

```

NumYMinorGridDivision, TrendCmdTarget Property

Syntax NumYMinorGridDivision = _Integer

Description Questa proprietà imposta o restituisce il numero di divisioni minori per la griglia dell'asse Y. Accetta un valore di tipo Integer compreso tra 1 e 10.

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()

```

```

objTrend.NumYMinorGridDivision = 10
objTrend.Refresh
End Sub

```

Page, TrendCmdTarget Property

Syntax Page = _Long

Description Questa proprietà imposta o restituisce il numero di pagina correntemente visualizzata nella finestra di Trend. Lo scorrimento delle pagine è consentito solo quando il Trend è in pausa. Per pagina si intende l'insieme di valori visualizzati nella finestra di Trend in un'unica soluzione. La pagina contenente i dati più recenti ha indice 0 e il limite massimo dipende dal numero di valori campionati e dai campionamenti per pagina.

Parameter	Description
None	None

Result Long

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.Freezed = True
    objTrend.Page = 5
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

PageNextBtnText, TrendCmdTarget Property

Syntax PagePrevBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando Pagina Precedente dell'oggetto Trend. Se non viene specificato nulla, Movicon utilizzerà il testo di default.

Parameter	Description
None	None

Result String

Example:

```

Dim objTrend As TrendCmdTarget
Public Sub Click()
    If Not objTrend Is Nothing Then
        MsgBox "objTrend 's PagePrevBtnText is " &
        objTrend.PagePrevBtnText, vbInformation, GetProjectTitle
        objTrend.PagePrevBtnText = "Prev Page"
        objTrend.Refresh
    Else

```

```

        MsgBox "objTrend is nothing",vbExclamation,GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub

```

PagePrevBtnText, TrendCmdTarget Property

Syntax PageNextBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando Pagina Successiva dell'oggetto Trend. Se non viene specificato nulla, Movicon utilizzerà il testo di default.

Parameter	Description
None	None

Result String

Example:

```

Dim objTrend As TrendCmdTarget
Public Sub Click()
    If Not objTrend Is Nothing Then
        MsgBox "objTrend 's PageNextBtnText is " &
            objTrend.PageNextBtnText,vbInformation,GetProjectTitle
        objTrend.PageNextBtnText = "Next Page"
        objTrend.Refresh
    Else
        MsgBox "objTrend is nothing",vbExclamation,GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub

```

PauseRunBtnText, TrendCmdTarget Property

Syntax PauseRunBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando Run/Pausa dell'oggetto Trend. Se non viene specificato nulla, Movicon utilizzerà il testo di default.

Parameter	Description
None	None

Result String

Example:

```

Dim objTrend As TrendCmdTarget
Public Sub Click()
    If Not objTrend Is Nothing Then

```

```

        MsgBox "objTrend 's PauseRunBtnText is " &
objTrend.PauseRunBtnText,vbInformation,GetProjectTitle
objTrend.PauseRunBtnText = "Run/Pause"
objTrend.Refresh
    Else
        MsgBox "objTrend is nothing",vbExclamation,GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub

```

PenAutoscale, TrendCmdTarget Property

Syntax PenAutoscale(_IpszPenName) = _Boolean

Description Questa proprietà consente di impostare la Scala della penna, specificata nel parametro IpszPenName, in modo automatico.

Parameter	Description
IpszPenName As String	Nome della penna

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.PenAutoscale(objTrend.GetPenNameFromList(0)) = Not
objTrend.PenAutoscale(objTrend.GetPenNameFromList(0))
objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

PenAverageLineColor, TrendCmdTarget Property

Syntax PenAverageLineColor(_IpszPenName) = _Long

Description Questa proprietà imposta o restituisce il colore che assume la traccia di rappresentazione del valore Medio assunto dalla variabile associata alla penna referenziata dal parametro IpszPenName.

Parameter	Description
IpszPenName As String	Nome della penna

Result Long

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.PenAverageLineColor(objTrend.GetPenNameFromList(0)) =
    RGB(255,0,0)
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

PenBackBrushPattern, TrendCmdTarget Property

Syntax PenBackBrushPattern(_lpszPenName) = _Integer

Description Questa proprietà imposta o restituisce lo stile grafico che assume l'area sottesa dalla penna (proprietà gestita solo se la penna è di tipo Area o Rettangolo). I diversi tipi di retinatura sono rappresentati da un valore intero compreso tra 0 e 5. Il valore -1 (default) imposta lo sfondo di tipo solido. Valori al di fuori di questo range generano errore nel richiamo della funzione.



Questa proprietà non è supportata in Windows CE. (Se impostata restituisce sempre -1)

Parameter	Description
lpszPenName As String	Nome della penna

Result Integer

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.PenBackBrushPattern(objTrend.GetPenNameFromList(0)) =
    CInt(InputBox("Insert value"))
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

PenBrushColor, TrendCmdTarget Property

Syntax PenBrushColor = _Long

Description Questa proprietà imposta o restituisce codice del colore di sfondo dell'area delle penne del Trend.

Parameter	Description
-----------	-------------

None	None
------	------

Result Long

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.PenBrushColor = RGB(255,0,0)
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

PenBrushVisible, TrendCmdTarget Property

Syntax PenBrushVisible = _Boolean

Description Questa proprietà imposta o restituisce la proprietà di visualizzazione del colore di sfondo dell'area penne del Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.PenBrushVisible = Not objTrend.PenBrushVisible
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

PenColor, TrendCmdTarget Property

Syntax PenColor(_IpszPenName) = _Long

Description Questa proprietà imposta o restituisce il colore della penna referenziata dal parametro IpszPenName.

Parameter	Description
IpszPenName As String	Nome della penna

Result Long

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.PenColor(objTrend.GetPenNameFromList(0)) = RGB(255,0,0)
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

PenDLColumnName, TrendCmdTarget Property

Syntax PenDLColumnName("_IpszPenName") = _String

Description Questa proprietà in lettura controlla se al trend è associato un datalogger e, se c'è, restituisce il valore della proprietà Data Logger Col per la penna passata come parametro. Se Data Logger Col non è specificata, restituisce il nome della penna. In scrittura imposta il valore della proprietà Data Logger Col per la penna passata come parametro.



La proprietà è disponibile solo per l'oggetto Data Analysis.

Parameter	Description
IpszPenName As String	Nome della penna del trend alla quale associare la colonna del datalogger

Result String

Example:

```
Public Sub Click()
Dim objTrend As TrendCmdTarget
Dim sPenName As String
Dim sPenDLName As String
Dim i As Integer
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
    For i= 0 To objTrend.GetPensNumber()
        sPenName = objTrend.GetPenNameFromList(i)
        If sPenName <> "" Then
'get DataLogger column name
            sPenDLName = objTrend.PenDLColumnName(sPenName)
            MsgBox "PenName = " & sPenName & "; DLColName = " & sPenDLName
'set DataLogger column name
            sPenDLName = "Col" & i
            objTrend.PenDLColumnName(sPenName) = sPenDLName
            MsgBox "PenName = " & sPenName & "; DLColName = " & sPenDLName
        End If
    Next
End sub
```

PenDLRName, TrendCmdTarget Property

Syntax PenDLRName(_IpszPenName) = _String

Description Tramite questa proprietà è possibile leggere o impostare il nome del Data Logger di riferimento della penna. Quando viene modificato il nome del Data Logger di riferimento di una penna occorre usare la funzione "LinkToDataLogger" affinché le modifiche siano applicate al Data Analysis.



Questa proprietà non è gestita dall'oggetto "Trend".

Parameter	Description
IpszPenName As String	Nome della penna del Data Analysis alla quale associare il Data Logger.

Result String

Example:

```
Dim objDataAnalysis As TrendCmdTarget
Public Sub Click()
    Set objDataAnalysis =
    GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
    objDataAnalysis.PenDLRName(sPenName) = sDataLoggerName
    objDataAnalysis.LinkToDataLogger(True)
    objDataAnalysis.Refresh
    Set objDataAnalysis = Nothing
End Sub
```

PenEditable, TrendCmdTarget Property

Syntax PenEditable(_IpszPenName) = _Boolean

Description Questa proprietà abilita o disabilita la possibilità di editare in run time le proprietà della penna referenziata dal parametro IpszPenName.

Parameter	Description
IpszPenName As String	Nome della penna

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.PenEditable(objTrend.GetPenNameFromList(0)) = Not
    objTrend.PenEditable(objTrend.GetPenNameFromList(0))
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

PenFormatScale, TrendCmdTarget Property

Syntax PenFormatScale(_IpszPenName) = _String

Description Questa proprietà imposta il formato di rappresentazione del valore numerico visualizzato nella scala della penna referenziata dal parametro IpszPenName.

Parameter	Description
IpszPenName As String	Nome della penna

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.PenFormatScale(objTrend.GetPenNameFromList(0)) = "xx.xx" 'i.e.
    for xx.xx -> value 3.7 is represented like 03.70
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

PenLogarithmicScale, TrendCmdTarget Property

Syntax PenLogarithmicScale(_IpszPenName) = _Boolean

Description Questa proprietà abilita o disabilita la scala logaritmica per la penna referenziata dal parametro IpszPenName.

Parameter	Description
IpszPenName As String	Nome della penna

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.PenLogarithmicScale(objTrend.GetPenNameFromList(0)) = True
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

PenMaxLineColor, TrendCmdTarget Property

Syntax PenMaxLineColor(_IpszPenName) = _Long

Description Questa proprietà imposta o restituisce il colore che assume la traccia di rappresentazione del valore Minimo assunto dalla variabile associata alla penna referenziata dal parametro IpszPenName.

Parameter	Description
IpszPenName As String	Nome della penna

Result Long

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.PenMaxLineColor(objTrend.GetPenNameFromList(0)) = RGB(255,0,0)
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

PenMaxValue, TrendCmdTarget Property

Syntax PenMaxValue(_IpszPenName) = _Double

Description Questa proprietà imposta o restituisce il valore massimo per la scala della penna specificata. Dopo l'impostazione di tale valore è necessario eseguire un comando di Refresh affinché tale modifica venga riportata a livello grafico.

Parameter	Description
IpszPenName As String	Nome della penna

Result Double

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.PenMaxValue(objTrend.GetPenNameFromList(0)) = 200
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

PenMinLineColor, TrendCmdTarget Property

Syntax PenMinLineColor(_IpszPenName) = _Long

Description Questa proprietà imposta o restituisce il colore che assume la traccia di rappresentazione del valore Minimo assunto dalla variabile associata alla penna referenziata dal parametro IpszPenName.

Parameter	Description
IpszPenName As String	Nome della penna

Result Long

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.PenMinLineColor(objTrend.GetPenNameFromList(0)) = RGB(255,0,0)
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

PenMinValue, TrendCmdTarget Property

Syntax PenMinValue(_IpszPenName) = _Double

Description Questa proprietà imposta o restituisce il valore minimo per la scala della penna specificata. Dopo l'impostazione di tale valore è necessario eseguire un comando di Refresh affinché tale modifica venga riportata a livello grafico.

Parameter	Description
IpszPenName As String	Nome della penna

Result Double

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.PenMinValue(objTrend.GetPenNameFromList(0)) = 0
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

PenPlotType, TrendCmdTarget Property

Syntax PenPlotType(_IpszPenName) = _Integer

Description Questa proprietà imposta o restituisce l'indice relativo al tipo di linea (o tipo di rappresentazione) utilizzato per la penna specificata. Per tali valori è possibile utilizzare l'enumeratore ePlottingType:

Valori ammessi:

enum_PT_POLYLINE = linea semplice (valore 0)

enum_PT_POLYRECTANGLE = barre verticali (valore 1)

enum_PT_POLYAREA = area piena (valore 2)

enum_PT_POLYLINEANDLINE = area a linee verticali (valore 3)

enum_PT_POLYLINESTEP = linea a gradino (valore 4)

enum_PT_POLYAREASTEP = area a gradino (valore 5)

Parameter	Description
IpszPenName As String	Nome della penna

Result Double

Example:

Option Explicit

Dim objTrend As TrendCmdTarget

Public Sub Click()

objTrend.**PenPlotType**(objTrend.GetPenNameFromList(0))

=

enum_PT_POLYAREA

objTrend.Refresh

End Sub

Public Sub SymbolLoading()

Set

objTrend

=

GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface

End Sub

PenScaleRightBottom, TrendCmdTarget Property

Syntax PenScaleRightBottom(_IpszPenName) = _Boolean

Description Questa proprietà imposta o restituisce la posizione della scala relativa alla penna referenziata.

I valori possibili sono:

False=Sinistra/Alto

True=Destra/Basso

Parameter	Description
IpszPenName As String	Nome della penna

Result Boolean

Example:

Option Explicit

Dim objTrend As TrendCmdTarget

Public Sub Click()


```

Dim ScalePos As Boolean

ScalePos = objTrend.PenScaleRightBottom(objTrend.GetPenNameFromList(0))
objTrend.PenScaleRightBottom(objTrend.GetPenNameFromList(0)) = Not
ScalePos
objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set                                objTrend                                =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

PenShowAverageLine, TrendCmdTarget Property

Syntax PenShowAverageLine(_IpszPenName) = _Boolean

Description Questa proprietà abilita o disabilita la visualizzazione della linea del valore medio per la penna specificata.

Parameter	Description
IpszPenName As String	Nome della penna

Result Boolean

Example:
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
 Dim tmp As Boolean

 tmp = objTrend.**PenShowAverageLine**(objTrend.GetPenNameFromList(0))
 objTrend.**PenShowAverageLine**(objTrend.GetPenNameFromList(0)) = Not tmp
 objTrend.Refresh
End Sub
Public Sub SymbolLoading()
 Set objTrend =
 GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

PenShowMaxLine, TrendCmdTarget Property

Syntax PenShowMaxLine(_IpszPenName) = _Boolean

Description Questa proprietà abilita o disabilita la visualizzazione della linea del valore massimo per la penna specificata.

Parameter	Description
IpszPenName As String	Nome della penna

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim tmp As Boolean

    tmp = objTrend.PenShowMaxLine(objTrend.GetPenNameFromList(0))
    objTrend.PenShowMaxLine(objTrend.GetPenNameFromList(0)) = Not tmp
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend                                     =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

PenShowMinLine, TrendCmdTarget Property

Syntax PenShowMinLine(_IpszPenName) = _Boolean

Description Questa proprietà abilita o disabilita la visualizzazione della linea del valore minimo per la penna specificata.

Parameter	Description
IpszPenName As String	Nome della penna

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim tmp As Boolean

    tmp = objTrend.PenShowMinLine(objTrend.GetPenNameFromList(0))
    objTrend.PenShowMinLine(objTrend.GetPenNameFromList(0)) = Not tmp
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend                                     =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

PenShowScale, TrendCmdTarget Property

Syntax PenShowScale(_IpszPenName) = _Boolean

Description Questa proprietà abilita o disabilita la visualizzazione della scala per la penna specificata.

Parameter	Description
IpszPenName As String	Nome della penna

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim tmp As Boolean

    tmp = objTrend.PenShowScale(objTrend.GetPenNameFromList(0))
    objTrend.PenShowScale(objTrend.GetPenNameFromList(0)) = Not tmp
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

PenSize, TrendCmdTarget Property

Syntax PenSize = _Integer

Description Questa proprietà imposta o restituisce la dimensione, espressa in pixel, dell'area destinata alla visualizzazione delle penne. Accetta un valore Integer compreso tra 5 e 25.

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.PenSize = 10
    objTrend.Refresh
End Sub

```

PenStyle, TrendCmdTarget Property

Syntax PenStyle("_lpszPenName") = _Short

Description Questa proprietà imposta o restituisce lo stile associato alla penna specificata dal parametro lpszPenName.
I Valori validi sono:

0 SOLID
 1 DASH
 2 DOT
 3 DASHDOT
 4 DASHDOTDOT
 5 NULL

Nota:

I valori DOT, DASHDOT e DASHDOTDOT non sono possono essere usati su Windows CE perchè non supportati.

Parameter	Description
lpszPenName As String	Nome della penna del trend alla quale assegnare lo stile grafico

Result Short

Example:

```
Public Sub Click()
Dim objTrend As TrendCmdTarget
Dim sPenName As String
Dim nPenStyle as Integer

Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
For i= 0 To objTrend.GetPensNumber()
    sPenName = objTrend.GetPenNameFromList(i)
    If sPenName <> "" Then
'get pen style
        nPenStyle = objTrend.PenStyle(sPenName)
'set pen style
        If nPenStyle < 0 Or nPenStyle > 5 Then
            nPenStyle = 0
        Else
            nPenStyle = nPenStyle + 1
        End If
        objTrend.PenStyle(sPenName) = nPenStyle
    End If
Next
End sub
```

PenVariable, TrendCmdTarget Property

Syntax PenVariable(_lpszPenName) = _String

Description Questa proprietà imposta o restituisce il nome della variabile associata alla penna specificata. Nel caso in cui si desideri modificare tale proprietà è necessario sospendere momentaneamente la registrazione impostando la proprietà Recording a False. Una eventuale modifica della variabile associata alla penna non influisce sulle variabili del Data Logger eventualmente collegato.

Parameter	Description
lpszPenName As String	Nome della penna

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim tmp As Boolean
    Debug.Print objTrend.PenVariable(objTrend.GetPenNameFromList(0))
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

PenWidth, TrendCmdTarget Property

Syntax PenWidth(_IpszPenName) = _Integer

Description Questa proprietà imposta o restituisce la larghezza della penna specificata. I valori ammessi sono compresi fra 1 e 5, valori al di fuori di questo range generano un errore.

Parameter	Description
IpszPenName As String	Nome della penna

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim tmp As Boolean
    objTrend.PenWidth(objTrend.GetPenNameFromList(0)) = 2
End Sub
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

PrevBtnText, TrendCmdTarget Property

Syntax PrevBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando Precedente dell'oggetto Trend. Se non viene specificato nulla, Movicon utilizzerà il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objTrend As TrendCmdTarget
Public Sub Click()
    If Not objTrend Is Nothing Then
        MsgBox "objTrend 's PrevBtnText is " & objTrend.PrevBtnText, vbInformation, GetProjectTitle
        objTrend.PrevBtnText = "Prev"
        objTrend.Refresh
    Else
        MsgBox "objTrend is nothing", vbExclamation, GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub
```

PrintBtnText, TrendCmdTarget Property

Syntax PrintBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando Stampa dell'oggetto Trend. Se non viene specificato nulla, Movicon utilizzerà il testo di default.



Questa proprietà non è supportata in Windows CE. (Se impostata restituisce sempre una stringa vuota)

Parameter	Description
None	None

Result String

Example:

```
Dim objTrend As TrendCmdTarget
Public Sub Click()
    If Not objTrend Is Nothing Then
        MsgBox "objTrend 's PrintBtnText is " &
            objTrend.PrintBtnText, vbInformation, GetProjectTitle
        objTrend.PrintBtnText = "Print"
        objTrend.Refresh
    Else
        MsgBox "objTrend is nothing", vbExclamation, GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub
```

Recording, TrendCmdTarget Property

Syntax Recording = _Boolean

Description Questa proprietà abilita o disabilita la registrazione nel Trend dei valori associati alle penne.



E' possibile modificare la proprietà Recording solo se l'opzione "Leggi dati in Background" (nelle proprietà "Esecuzione" del Trend) è Disabilitata.

Parameter	Description
IpszPenName As String	Nome della penna

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim tmp As Boolean

    tmp = objTrend.Recording
```

```

        objTrend.Recording = Not tmp
        objTrend.Refresh
    End Sub
    Public Sub SymbolLoading()
        Set                                objTrend
        GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
    End Sub

```

RecordOnFile, TrendCmdTarget Property

Syntax RecordOnFile = _Boolean

Description Questa proprietà abilita o disabilita alla registrazione su file, in formato "CSV", dei dati campionati dall'oggetto Trend. Nel caso in cui si intenda modificare tale impostazione è necessario sospendere momentaneamente la registrazione impostando la proprietà Recording a False.

Parameter	Description
IpszPenName As String	Nome della penna

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim tmp As Boolean

    objTrend.Recording = False
    tmp = objTrend.RecordOnFile
    objTrend.RecordOnFile = Not tmp
    objTrend.Recording = True
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set                                objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

SampleDateTime, TrendCmdTarget Property

Syntax SampleDateTime(_IPosition) = _Date

Description Questa proprietà imposta o restituisce la data e l'ora, in formato Date, del campionamento indicato come parametro. L'eventuale modifica della data influisce solo sul valore nel buffer del Trend e non nel file o nel Data Logger associato.



Questa funzione non è gestita dall'oggetto "Data Analysis".

Parameter	Description
IPosition As Long	Numero del campionamento

Result Date

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.SampleDateTime(0)
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

SampleDateTimeMs, TrendCmdTarget Property

Syntax SampleDateTimeMs(_IPosition) = _Integer

Description Questa proprietà imposta o restituisce il numero di millisecondi, abbinati alla data e all'ora, del campionamento indicato come parametro. L'eventuale modifica della data influisce solo sul valore nel buffer del Trend e non nel file o nel Data Logger associato.

Parameter	Description
IPosition As Long	Numero del campionamento

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    If objTrend Is Nothing Then Exit Sub
    Debug.Print objTrend.SampleDateTimeMs(0)
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

SamplePerUpdate, TrendCmdTarget Property

Syntax SamplePerUpdate = _Integer

Description Questa proprietà imposta o restituisce il numero di campionamenti per ogni aggiornamento grafico, cioè il numero di dati campionati da visualizzare per ogni scorrimento della pagina di Trend. Nel caso in cui si intenda modificare tale impostazione è necessario sospendere momentaneamente la registrazione impostando la proprietà Recording a False.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    If objTrend Is Nothing Then Exit Sub
    objTrend.Recording = False
    objTrend.SamplePerUpdate = 56
    objTrend.Recording = True
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

Samples, TrendCmdTarget Property

Syntax Samples = _Long

Description Questa proprietà rappresenta il numero totale di campionamenti che il Trend o il Data Analysis possono gestire. Il Trend ha un suo buffer di lavoro che è limitato e può al massimo essere impostato al valore 10.000. Per il Data Analysis invece questo limite non c'è.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim s As String
    If objTrend Is Nothing Then
        Set objTrend =
        GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
    End If
    s$ = InputBox("VAL.:", "Samples", CStr(objTrend.Samples))
    If s$ <> "" Then
        objTrend.Recording = False
        objTrend.Samples = (CLng(s$))
        objTrend.Recording = True
        objTrend.Refresh
    End If
End Sub
```

SampleValue, TrendCmdTarget Property

Syntax SampleValue(IpszPenName, IPosition) = Double

Description Questa proprietà imposta o restituisce il valore del campionamento della penna indicata alla posizione i-esima nel buffer del Trend.



Questa funzione non è gestita dall'oggetto "Data Analysis".

Parameter	Description
IpszPenName As String IPosition As Long	Nome della penna Numero del campionamento

Result Double

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim s As String
    If objTrend Is Nothing Then
        Set objTrend = GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
    End If
    Dim i As Long
    For i = 1 To 100
        Debug.Print objTrend.SampleValue(objTrend.GetPenNameFromList(0),i)
        objTrend.SampleValue(objTrend.GetPenNameFromList(0),i) = 0
    Next i
End Sub
```

ScrollPosition, TrendCmdTarget Property

Syntax ScrollPosition = _Long

Description Questa proprietà di sola lettura restituisce, quando il Trend è in pausa, il numero del campionamento più recente mostrato nella pagina del Trend attualmente visualizzata.

Parameter	Description
IpszPenName As String	Nome della penna
IPosition As Long	Numero del campionamento

Result Long

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    If objTrend Is Nothing Then
        Set objTrend = GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
    End If
    Debug.Print objTrend.ScrollPosition
End Sub
```

SecBtnText, TrendCmdTarget Property

Syntax SecBtnText = _String

Description Questa proprietà imposta o restituisce il testo che dovrà apparire sui pulsanti di scorrimento quando si seleziona come intervallo di tempo da visualizzare il Minuto dell'oggetto Data Analysis. Se non viene specificato nulla, Movicon utilizzerà il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objDataAnalysis As TrendCmdTarget
Public Sub Click()
    If Not objDataAnalysis Is Nothing Then
        MsgBox "objDataAnalysis's SecBtnText is " &
            objDataAnalysis.SecBtnText, vbInformation, GetProjectTitle
        objDataAnalysis.SecBtnText = "Seconds"
        objDataAnalysis.Refresh
    Else
        MsgBox "objDataAnalysis is nothing", vbExclamation, GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objDataAnalysis =
        GetSynopticObject.GetSubObject("DataAnalysis").GetObjectInterface
End Sub
```

SecRecTime, TrendCmdTarget Property

Syntax SecRecTime = _Integer

Description Questa proprietà imposta o restituisce il numero di secondi che compongono la proprietà "Periodo di campionamento".

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim objTrend As TrendCmdTarget
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
    objTrend.HourRecTime = 1
    objTrend.MinRecTime = 30
    objTrend.SecRecTime = 0
End Sub

Public Sub SymbolLoading()
    Set objTrend =
        GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

End Sub

SecViewTime, TrendCmdTarget Property

Syntax SecViewTime = Integer

Description	Questa proprietà imposta o restituisce il numero di secondi impostato nell'intervallo di visualizzazione.
--------------------	---

Parameter	Description
None	None

Result	Integer
---------------	---------

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim objTrend As TrendCmdTarget
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
    objTrend.SecViewTime = 20
End Sub
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

ShiftGrid, TrendCmdTarget Property

Syntax ShiftGrid = _Boolean

Description	Questa proprietà imposta o restituisce l'abilitazione allo scorrimento della griglia del Trend. Accetta un valore Boolean.
--------------------	--

Parameter	Description
None	None

Result	Boolean
---------------	---------

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.ShiftGrid = Not objTrend.ShiftGrid
    objTrend.Refresh
End Sub
```

ShowBreakLines, TrendCmdTarget Property

Syntax ShowBreakLines = _Boolean

Description Questa proprietà imposta o restituisce l'abilitazione alla visualizzazione delle linee di interruzione. Le linee di interruzione sono delle linee perpendicolari alla direzione del Trend che indicano, quando il Trend e' in pausa, i punti in cui la registrazione dei valori si e' interrotta per un tempo superiore al periodo di campionamento impostato. Accetta un valore Boolean.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.ShowBreakLines = Not objTrend.ShowBreakLines
    objTrend.Refresh
End Sub
```

ShowCompareTimeFrameBtn, TrendCmdTarget Property

Syntax ShowCompareTimeFrameBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere i pulsanti per la selezione dell'intervallo di rappresentazione dei dati di comparazione per l'oggetto Data Analysis. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Refresh" per aggiornare graficamente l'oggetto.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim obj As TrendCmdTarget
Public Sub Click()
    If obj Is Nothing Then Set obj =
        GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
    obj.ShowCompareTimeFrameBtn = Not obj.ShowCompareTimeFrameBtn
    obj.Refresh
End Sub
```

ShowDate, TrendCmdTarget Property

Syntax ShowDate = _Boolean

Description Questa proprietà imposta o restituisce l'abilitazione alla visualizzazione della data dei campionamenti nel Trend. Accetta un valore Boolean.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.ShowDate = Not objTrend.ShowDate
    objTrend.Refresh
End Sub
```

ShowExpandBtn, TrendCmdTarget Property

Syntax ShowExpandBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante di comando Espandi dell'oggetto Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objTrend As TrendCmdTarget
Public Sub Click()
    If Not objTrend Is Nothing Then
        MsgBox "objTrend 's ShowExpandBtn is " &
            objTrend.ShowExpandBtn, vbInformation, GetProjectTitle
        objTrend.ShowExpandBtn = Not objTrend.ShowExpandBtn
        objTrend.Refresh
    Else
        MsgBox "objTrend is nothing", vbExclamation, GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub
```

ShowFirstPointBtn, TrendCmdTarget Property

Syntax ShowFirstPointBtn = _Boolean

Description Questa proprietà rende visibile o meno il pulsante di scorrimento del cursore sul primo punto della curva visualizzata sul Data Analysis.
Questa proprietà è gestita soltanto dall'oggetto Data Analysis e non dal Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objDataAnalysis As TrendCmdTarget
Public Sub Click()

    If Not objDataAnalysis Is Nothing Then
        objDataAnalysis.ShowFirstPointBtn = Not objDataAnalysis.ShowFirstPointBtn
    End If
End Sub

Public Sub SymbolLoading()
    Set objDataAnalysis = GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
End Sub
```

ShowLegend, TrendCmdTarget Property

Syntax ShowLegend = _Boolean

Description Questa proprietà imposta o restituisce l'abilitazione alla visualizzazione della legenda del Trend. Accetta un valore Boolean.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.ShowLegend = Not objTrend.ShowLegend
    objTrend.Refresh
End Sub
```

ShowLastPointBtn, TrendCmdTarget Property

Syntax ShowLastPointBtn = _Boolean

Description Questa proprietà rende visibile o meno il pulsante di scorrimento del cursore sull'ultimo punto della curva visualizzata sul Data Analysis.
Questa proprietà è gestita soltanto dall'oggetto Data Analysis e non dal Trend.

Parameter	Description
None	None

Result Boolean

Example:

Option Explicit

Dim objDataAnalysis As TrendCmdTarget

Public Sub Click()

```
    If Not objDataAnalysis Is Nothing Then
        objDataAnalysis.ShowNextPointBtn = Not
        objDataAnalysis.ShowNextPointBtn
```

```
    End If
```

End Sub

Public Sub SymbolLoading()

```
    Set objDataAnalysis =
    GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
```

End Sub

ShowMeasureBtn, TrendCmdTarget Property

Syntax ShowMeasureBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante di Misura per l'oggetto Data Analysis. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Refresh" per aggiornare graficamente l'oggetto.

Parameter	Description
None	None

Result Boolean

Example:

Dim obj As TrendCmdTarget

Public Sub Click()

```
    If obj Is Nothing Then Set obj =
    GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
    obj.ShowMeasureBtn = Not obj.ShowMeasureBtn
    obj.Refresh
```

End Sub

ShowMinorXGrid, TrendCmdTarget Property

Syntax ShowMinorXGrid = _Boolean

Description Questa proprietà imposta o restituisce l'abilitazione alla visualizzazione della griglia minore dell'asse X del Trend. Accetta un valore Boolean.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.ShowMinorXGrid = Not objTrend.ShowMinorXGrid
    objTrend.Refresh
End Sub
```

ShowMinorYGrid, TrendCmdTarget Property

Syntax ShowMinorYGrid = _Boolean

Description Questa proprietà imposta o restituisce l'abilitazione alla visualizzazione della griglia minore dell'asse Y del Trend. Accetta un valore Boolean.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.ShowMinorYGrid = Not objTrend.ShowMinorYGrid
    objTrend.Refresh
End Sub
```

ShowMsec, TrendCmdTarget Property

Syntax ShowMsec = _Boolean

Description Questa proprietà imposta o restituisce l'abilitazione alla visualizzazione dei millisecondi nell'orario del Trend. Accetta un valore Boolean.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.ShowMsec = Not objTrend.ShowMsec
    objTrend.Refresh
End Sub
```

ShowNextBtn, TrendCmdTarget Property

Syntax ShowNextBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante di comando Prossimo dell'oggetto Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objTrend As TrendCmdTarget
Public Sub Click()
    If Not objTrend Is Nothing Then
        MsgBox "objTrend 's ShowNextBtn is " &
            objTrend.ShowNextBtn, vbInformation, GetProjectTitle
        objTrend.ShowNextBtn = Not objTrend.ShowNextBtn
        objTrend.Refresh
    Else
        MsgBox "objTrend is nothing", vbExclamation, GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub
```

ShowNextPointBtn, TrendCmdTarget Property

Syntax ShowNextPointBtn = _Boolean

Description Questa proprietà rende visibile o meno il pulsante di scorrimento del cursore sul punto successivo della curva visualizzata sul Data Analysis.
Questa proprietà è gestita soltanto dall'oggetto Data Analysis e non dal Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objDataAnalysis As TrendCmdTarget
Public Sub Click()
    If Not objDataAnalysis Is Nothing Then
        objDataAnalysis.ShowNextPointBtn=
        objDataAnalysis.ShowNextPointBtn
    End If
End Sub

Public Sub SymbolLoading()
    Set
        objDataAnalysis
        GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
    =
End Sub
```

ShowPageNextBtn, TrendCmdTarget Property

Syntax ShowPageNextBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante di comando Pagina Successiva dell'oggetto Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objTrend As TrendCmdTarget
Public Sub Click()
    If Not objTrend Is Nothing Then
        MsgBox "objTrend 's ShowPageNextBtn is " &
        objTrend.ShowPageNextBtn,vbInformation,GetProjectTitle
        objTrend.ShowPageNextBtn = Not objTrend.ShowPageNextBtn
        objTrend.Refresh
    Else
        MsgBox "objTrend is nothing",vbExclamation,GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub
```

ShowPagePrevBtn, TrendCmdTarget Property

Syntax ShowPagePrevBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante di comando Pagina Precedente dell'oggetto Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objTrend As TrendCmdTarget
Public Sub Click()
    If Not objTrend Is Nothing Then
        MsgBox "objTrend 's ShowPagePrevBtn is " &
            objTrend.ShowPagePrevBtn,vbInformation,GetProjectTitle
        objTrend.ShowPagePrevBtn = Not objTrend.ShowPagePrevBtn
        objTrend.Refresh
    Else
        MsgBox "objTrend is nothing",vbExclamation,GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub
```

ShowPauseRunBtn, TrendCmdTarget Property

Syntax ShowPauseRunBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante di comando Run/Pausa dell'oggetto Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objTrend As TrendCmdTarget
Public Sub Click()
    If Not objTrend Is Nothing Then
        MsgBox "objTrend 's ShowPauseRunBtn is " &
            objTrend.ShowPauseRunBtn,vbInformation,GetProjectTitle
        objTrend.ShowPauseRunBtn = Not objTrend.ShowPauseRunBtn
        objTrend.Refresh
    Else
        MsgBox "objTrend is nothing",vbExclamation,GetProjectTitle
    End If
End Sub
```

```

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub

```

ShowPen, TrendCmdTarget Property

Syntax ShowPen = _Boolean

Description Questa proprietà imposta o restituisce l'abilitazione alla visualizzazione delle penne del Trend. Accetta un valore Boolean.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub

Public Sub Click()
    objTrend.ShowPen = Not objTrend.ShowPen
    objTrend.Refresh
End Sub

```

ShowPenLabels, TrendCmdTarget Property

Syntax ShowPenLabels(_IpszPenName) = Boolean

Description Questa proprietà consente di leggere ed impostare lo stato di visualizzazione dell'etichette dei valori in corrispondenza di ogni punto del grafico. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Refresh" per aggiornare graficamente l'oggetto.

Parameter	Description
IpszPenName as string	Nome della penna

Result Boolean

Example:

```

Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Dim i As Long
    Dim sPenList() As String
    Dim nPenSelected As Long

    If obj Is Nothing Then Set obj =
    GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    Erase sPenList
    For i = 0 To obj.GetPensNumber - 1

```

```

        ReDim Preserve sPenList(i)
        sPenList(i) = obj.GetPenNameFromList(i)
    Next
    nPenSelected = ShowPopupMenu(sPenList)
    If nPenSelected >= 0 Then
        obj.ShowPenLabels(sPenList(nPenSelected)) = Not
        obj.ShowPenLabels(sPenList(nPenSelected))
        obj.Refresh
    End If
End Sub

```

ShowPenPoints, TrendCmdTarget Property

Syntax ShowPenPoints(_IpszPenName) = Boolean

Description Questa proprietà, se abilitata, permette di visualizzare tutti i punti per la linea selezionata.

Parameter	Description
IpszPenName As String	Nome della penna

Result Integer

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.ShowPenPoints(objTrend.GetPenNameFromList(0)) = Not
    objTrend.ShowPenPoints(objTrend.GetPenNameFromList(0))
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

ShowPrevBtn, TrendCmdTarget Property

Syntax ShowPrevBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante di comando Precedente dell'oggetto Trend.

Parameter	Description
None	None

Result Boolean

Example:

```

Dim objTrend As TrendCmdTarget
Public Sub Click()
    If Not objTrend Is Nothing Then
        MsgBox "objTrend 's ShowPrevBtn is " &
        objTrend.ShowPrevBtn, vbInformation, GetProjectTitle
    End If
End Sub

```

```

objTrend.ShowPrevBtn = Not objTrend.ShowPrevBtn
objTrend.Refresh
Else
    MsgBox "objTrend is nothing",vbExclamation,GetProjectTitle
End If
End Sub

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub

```

ShowPrevPointBtn, TrendCmdTarget Property

Syntax ShowPrevPointBtn = _Boolean

Description Questa proprietà rende visibile o meno il pulsante di scorrimento del cursore sul punto precedente della curva visualizzata sul Data Analysis.
Questa proprietà è gestita soltanto dall'oggetto Data Analysis e non dal Trend.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Dim objDataAnalysis As TrendCmdTarget
Public Sub Click()

    If Not objDataAnalysis Is Nothing Then
        objDataAnalysis.ShowPrevPointBtn= Not
        objDataAnalysis.ShowPrevPointBtn
    End If
End Sub

Public Sub SymbolLoading()
    Set objDataAnalysis =
    GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
End Sub

```

ShowPrintBtn, TrendCmdTarget Property

Syntax ShowPrintBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante di comando Stampa dell'oggetto Trend.



Questa proprietà non è supportata in Windows CE.(Se impostata restituisce sempre ' false')

Parameter	Description
None	None

Result Boolean

Example:

```
Dim objTrend As TrendCmdTarget
Public Sub Click()
    If Not objTrend Is Nothing Then
        MsgBox "objTrend 's ShowPrintBtn is " &
            objTrend.ShowPrintBtn,vbInformation,GetProjectTitle
        objTrend.ShowPrintBtn = Not objTrend.ShowPrintBtn
        objTrend.Refresh
    Else
        MsgBox "objTrend is nothing",vbExclamation,GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("Trend").GetObjectInterface
End Sub
```

ShowSavedValues, TrendCmdTarget Property

Syntax ShowSavedValues(_IpszPenName) = Boolean

Description Questa proprietà, se abilitata, permette di visualizzare i valori salvati con le funzioni SaveAllCurrentValue o SaveCurrentValue, per la penna specificata con il parametro IpszPenName. I valori salvati delle penne sono visualizzati attraverso una linea tratteggiata.

Parameter	Description
IpszPenName As String	Nome della penna

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.ShowSavedValues(objTrend.GetPenNameFromList(0)) = Not
    objTrend.ShowSavedValues(objTrend.GetPenNameFromList(0))
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

ShowTime, TrendCmdTarget Property

Syntax ShowTime = _Boolean

Description Questa proprietà imposta o restituisce l'abilitazione alla visualizzazione dell'orario del Trend. Accetta un valore Boolean.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.ShowTime = Not objTrend.ShowTime
    objTrend.Refresh
End Sub
```

ShowTimeFrameBtn, TrendCmdTarget Property

Syntax ShowTimeFrameBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere i pulsanti per la selezione dell'intervallo di rappresentazione dei dati per l'oggetto Data Analysis. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Refresh" per aggiornare graficamente l'oggetto.

Parameter	Description
None	None

Result Boolean

Example:

```
Dim obj As TrendCmdTarget
Public Sub Click()
    If obj Is Nothing Then Set obj =
        GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
    obj.ShowTimeFrameBtn = Not obj.ShowTimeFrameBtn
    obj.Refresh
End Sub
```

ShowTitle, TrendCmdTarget Property

Syntax ShowTitle = _Boolean

Description Questa proprietà imposta o restituisce l'abilitazione alla visualizzazione del titolo del Trend. Accetta un valore Boolean.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.ShowTitle = Not objTrend.ShowTitle
    objTrend.Refresh
End Sub

```

ShowXGrid, TrendCmdTarget Property

Syntax ShowXGrid = _Boolean

Description Questa proprietà imposta o restituisce l'abilitazione alla visualizzazione della griglia dell'asse X del Trend. Accetta un valore Boolean.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.ShowXGrid = Not objTrend.ShowXGrid
    objTrend.Refresh
End Sub

```

ShowYGrid, TrendCmdTarget Property

Syntax ShowYGrid = _Boolean

Description Questa proprietà imposta o restituisce l'abilitazione alla visualizzazione della griglia dell'asse Y del Trend. Accetta un valore Boolean.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.ShowYGrid = Not objTrend.ShowYGrid

```

```

objTrend.Refresh
End Sub

```

ShowZoomBtn, TrendCmdTarget Property

Syntax ShowZoomBtn = _Boolean

Description Questa proprietà permette di visualizzare o nascondere il pulsante di Zoom. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Refresh" per aggiornare graficamente l'oggetto.

Parameter	Description
None	None

Result Boolean

Example:

```

Dim obj As TrendCmdTarget
Public Sub Click()
    If obj Is Nothing Then Set obj =
        GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
    obj.ShowZoomBtn = Not obj.ShowZoomBtn
    obj.Refresh
End Sub

```

StartNewFile , TrendCmdTarget Property

Syntax StartNewFile = _Boolean

Description Questa proprietà permette di abilitare la creazione di un nuovo file, di formato CSV, ogni volta che il progetto viene eseguito in run. Affinchè tale modifica abbia effetto operativo, occorre salvare le nuove impostazioni nel file di configurazione del Trend. Se impostata al valore booleano True, al primo riavvio del progetto il sistema carica la nuova configurazione e provvede a inizializzare il nuovo file.

Parameter	Description
None	None

Result Long

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.StartNewFile = True
    objTrend.MaxFileLength = CInt(InputBox("Max file length"))
    objTrend.SaveExtSettings
End Sub
Public Sub SymbolLoading()
    Set objTrend =
        GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

StatAverageValue, TrendCmdTarget Property

Syntax StatAverageValue(IpszPenName) = Double

Description Questa proprietà imposta o restituisce il valore medio dei campionamenti di una penna calcolato sui valori visualizzati nella finestra di Trend. Una eventuale modifica del valore rimane valida solo fino al successivo aggiornamento del Trend, quando cioè il sistema calcola nuovamente il valore medio.

Parameter	Description
None	None

Result Double

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Debug.Print objTrend.StatAverageValue(objTrend.GetPenNameFromList(0))
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

StatMaxValue, TrendCmdTarget Property

Syntax StatMaxValue(IpszPenName) = Double

Description Questa proprietà imposta o restituisce il valore massimo dei campionamenti di una penna calcolato sui valori visualizzati nella finestra di Trend. Una eventuale modifica del valore rimane valida solo fino al successivo aggiornamento del Trend, quando cioè il sistema calcola nuovamente il valore massimo.

Parameter	Description
None	None

Result Double

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Debug.Print objTrend.StatMaxValue(objTrend.GetPenNameFromList(0))
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

StatMinValue, TrendCmdTarget Property

Syntax StatMinValue(IpszPenName) = Double

Description Questa proprietà imposta o restituisce il valore minimo dei campionamenti di una penna calcolato sui valori visualizzati nella finestra di Trend. Una eventuale modifica del valore rimane valida solo fino al successivo aggiornamento del Trend, quando cioè il sistema calcola nuovamente il valore minimo.

Parameter	Description
None	None

Result Double

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Debug.Print objTrend.StatMinValue(objTrend.GetPenNameFromList(0))
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

TimeBrushColor, TrendCmdTarget Property

Syntax TimeBrushColor = _Long

Description Questa proprietà imposta o restituisce codice del colore di sfondo dell'area tempi del Trend.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.TimeBrushColor = RGB(255,0,0)
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

TimeBrushVisible, TrendCmdTarget Property

Syntax TimeBrushVisible = _Boolean

Description Questa proprietà imposta o restituisce la proprietà di visualizzazione del colore di sfondo dell'area tempi del Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.TimeBrushVisible = Not objTrend.TimeBrushVisible
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

TimeFrameBtnColor, TrendCmdTarget Property

Syntax TimeFrameBtnColor = _Long

Description Questa proprietà consente di leggere ed impostare il colore usato per visualizzare i pulsanti per la selezione dell'intervallo di rappresentazione dei dati per l'oggetto Data Analysis. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Refresh" per aggiornare graficamente l'oggetto.

Parameter	Description
None	None

Result Long

Example:

```
Dim obj As TrendCmdTarget
Public Sub Click()
    Dim IColor As Long

    If obj Is Nothing Then Set obj =
    GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
    If ChooseColor(IColor) Then
        obj.TimeFrameBtnColor = IColor
        obj.Refresh
    End If
End Sub
```

TimeTextColor, TrendCmdTarget Property

Syntax TimeTextColor = _Long

Description Questa proprietà imposta o restituisce la proprietà di visualizzazione del colore del testo dell'area tempi del Trend.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.TimeTextColor = RGB(255,0,0)
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

TimeScale, TrendCmdTarget Property

Syntax TimeScale= _Byte

Description Questa proprietà imposta o restituisce il valore corrispondente alla selezione della 'Scala dei Tempi' in modo da poter adattare l'asse dei tempi secondo la seguente lista:

0 = Adjust to Values
1 = Absolute Range
2 = Adjust to Range

Questa proprietà è gestita soltanto dall'oggetto Data Analysis e non dal Trend.

Parameter	Description
None	None

Result Byte

Example:

```
Option Explicit

Enum eTimeScale
    eAdjustToValues = 0
    eAbsoluteRange = 1
    eAdjustToRange = 2
End Enum
Dim objDataAnalysis As TrendCmdTarget
Public Sub Click()

    If Not objDataAnalysis Is Nothing Then
        objDataAnalysis.TimeScale = eAdjustToValues
    End If
End Sub
```

```

        objDataAnalysis.Requery
    End If
End Sub

Public Sub SymbolLoading()
    Set                                objDataAnalysis                                =
    GetSynopticObject.GetSubObject("objDataAnalysis").GetObjectInterface
End Sub

```

TrendQualityFreezeMode, TrendCmdTarget Property

Syntax _Byte

Description Imposta la modalità di gestione della qualità per fermare o meno il Trend nel caso la qualità delle penne diventi 'Non Buona'. E' possibile impostare i valori enumerali di tipo "enum_TrendQualityFreezeMode":
 enum_TRA_ANYPENBAD : Il Trend si ferma appena una penna assume la Qualità Non Buona
 enum_TRA_ALLPENBAD : Il Trend si ferma quando tutte le penne assumono la Qualità Non Buona
 enum_TRA_DONTFREEZE : Il Trend non si ferma se una o tutte le penne assumono la Qualità Non Buona

Parameter	Description
None	None

Result None

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim sCommand(0 To 2) As String
    Dim nCommand As Integer
    Set objTrend = GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    sCommand(0) = "ANY PEN BAD"
    sCommand(1) = "ALL PEN BAD"
    sCommand(2) = "DONT FREEZE"
    nCommand = ShowPopupMenu(sCommand)
    Select Case nCommand
    Case 0
        objTrend.TrendQualityFreezeMode = enum_TRA_ANYPENBAD
    Case 1
        objTrend.TrendQualityFreezeMode = enum_TRA_ALLPENBAD
    Case 2
        objTrend.TrendQualityFreezeMode = enum_TRA_DONTFREEZE
    End Select
    Set objTrend = Nothing
End Sub

```

TrendBrushColor, TrendCmdTarget Property

Syntax TrendBrushColor = _Long

Description Questa proprietà imposta o restituisce codice del colore di sfondo dell'area del Trend.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.TrendBrushColor = RGB(255,0,0)
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

TrendBrushVisible, TrendCmdTarget Property

Syntax TrendBrushVisible = _Boolean

Description Questa proprietà imposta o restituisce la proprietà di visualizzazione del colore di sfondo dell'area Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.TrendBrushVisible = Not objTrend.TrendBrushVisible
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

TrendRunningType, TrendCmdTarget Property

Syntax TrendRunningType = _Integer

Description Questa proprietà di sola lettura restituisce il tipo di esecuzione per l'oggetto Trend inteso come Run/Stop, Solo Run e Solo Stop.

Ritorna un valore Integer con il seguente significato:
0=Run/Stop

1=Solo Run
2=Solo Stop.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Debug.Print objTrend.TrendRunningType
    objTrend.Refresh
End Sub
Public Sub SymbolLoading()
    Set                                objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

VariableAddValue, TrendCmdTarget Property

Syntax VariableAddValue = _String

Description Questa proprietà imposta o restituisce la variabile di registrazione a comando del Trend. Se la variabile "Agg. Val." viene usata il Trend non registra più su base temporale, ma esegue una registrazione ogni volta che tale variabile viene impostata al valore 1. Eseguita la registrazione Movicon riporta la variabile al valore 0. Impostando di nuovo la variabile con una stringa nulla il Trend riprende la registrazione a tempo.
Prima di utilizzare la proprietà per cambiare la variabile associata è necessario sospendere momentaneamente la registrazione impostando la proprietà Recording a False.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim sName As String
    Debug.Print "VariableAddValue = " & objTrend.VariableAddValue
    GetVariableNameFromList(sName)
    objTrend.Recording = False
    objTrend.VariableAddValue = sName
    objTrend.Recording = True
    Debug.Print "VariableAddValue = " & objTrend.VariableAddValue
End Sub
Public Sub SymbolLoading()
    Set                                objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

VariableCursorPosIn, TrendCmdTarget Property

Syntax VariableCursorPosIn = _String

Description Questa proprietà imposta o restituisce il nome della variabile associata alla funzione di "Cursore In" dell'oggetto Trend. Per impostare una nuova variabile è necessario sospendere momentaneamente la registrazione impostando la proprietà Recording a False.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim sName As String
    Debug.Print "VariableCursorPosIn = " & objTrend.VariableCursorPosIn
    GetVariableNameFromList(sName)
    objTrend.Recording = False
    objTrend.VariableCursorPosIn = sName
    objTrend.Recording = True
    Debug.Print "VariableCursorPosIn = " & objTrend.VariableCursorPosIn
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

VariableCursorPosOut, TrendCmdTarget Property

Syntax VariableCursorPosOut = _String

Description Questa proprietà imposta o restituisce il nome della variabile associata alla funzione di "Cursore Out" dell'oggetto Trend. Per impostare una nuova variabile è necessario sospendere momentaneamente la registrazione impostando la proprietà Recording a False.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim sName As String
    Debug.Print "VariableCursorPosOut = " & objTrend.VariableCursorPosOut
    GetVariableNameFromList(sName)
    objTrend.Recording = False
```

```

objTrend.VariableCursorPosOut = sName
objTrend.Recording = True
Debug.Print "VariableCursorPosOut = " & objTrend.VariableCursorPosOut
End Sub
Public Sub SymbolLoading()
    Set                                objTrend                                =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

VariableEnabling, TrendCmdTarget Property

Syntax VariableEnabling = _String

Description Questa proprietà imposta o restituisce il nome della variabile associata alla funzione di "Abilita" dell'oggetto Trend. Per impostare una nuova variabile e' necessario sospendere momentaneamente la registrazione impostando la proprietà Recording a False.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim sName As String
    Debug.Print "VariableEnabling = " & objTrend.VariableEnabling
    GetVariableNameFromList(sName)
    objTrend.Recording = False
    objTrend.VariableEnabling = sName
    objTrend.Recording = True
    Debug.Print "VariableEnabling = " & objTrend.VariableEnabling
End Sub
Public Sub SymbolLoading()
    Set                                objTrend                                =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

VariableFreezedMode, TrendCmdTarget Property

Syntax VariableFreezedMode = _String

Description Questa proprietà imposta o restituisce il nome della variabile associata alla funzione di "Start / Stop " dell'oggetto Trend. Per impostare una nuova variabile e' necessario sospendere momentaneamente la registrazione impostando la proprietà Recording a False.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim sName As String
    Debug.Print "VariableFreezedMode = " & objTrend.VariableFreezedMode
    GetVariableNameFromList(sName)
    objTrend.Recording = False
    objTrend.VariableFreezedMode = sName
    objTrend.Recording = True
    Debug.Print "VariableFreezedMode = " & objTrend.VariableFreezedMode
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

=

VariableResetAllValues, TrendCmdTarget Property

Syntax VariableResetAllValues = _String

Description Questa proprietà imposta o restituisce il nome della variabile associata alla funzione di "Reset" dell'oggetto Trend. Per impostare una nuova variabile e' necessario sospendere momentaneamente la registrazione impostando la proprietà Recording a False.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim sName As String
    Debug.Print "VariableResetAllValues = " & objTrend.VariableResetAllValues
    GetVariableNameFromList(sName)
    objTrend.Recording = False
    objTrend.VariableResetAllValues = sName
    objTrend.Recording = True
    Debug.Print "VariableResetAllValues = " & objTrend.VariableResetAllValues
End Sub
Public Sub SymbolLoading()
    Set                                     objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

=

VariableScrollEnd, TrendCmdTarget Property

Syntax VariableScrollEnd = _String

Description Questa proprietà imposta o restituisce il nome della variabile associata alla funzione di "Fine" dell'oggetto Trend. Per impostare una nuova variabile e' necessario sospendere momentaneamente la registrazione impostando la proprietà Recording a False.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim sName As String
    Debug.Print "VariableScrollEnd = " & objTrend.VariableScrollEnd
    GetVariableNameFromList(sName)
    objTrend.Recording = False
    objTrend.VariableScrollEnd = sName
    objTrend.Recording = True
    Debug.Print "VariableScrollEnd = " & objTrend.VariableScrollEnd
End Sub
Public Sub SymbolLoading()
    Set                                objTrend                                =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

VariableScrollNext, TrendCmdTarget Property

Syntax VariableScrollNext = _String

Description Questa proprietà imposta o restituisce il nome della variabile associata alla funzione di "Prossimo" dell'oggetto Trend. Per impostare una nuova variabile e' necessario sospendere momentaneamente la registrazione impostando la proprietà Recording a False.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim sName As String
    Debug.Print "VariableScrollNext = " & objTrend.VariableScrollNext
    GetVariableNameFromList(sName)
    objTrend.Recording = False
    objTrend.VariableScrollNext = sName
    objTrend.Recording = True
    Debug.Print "VariableScrollNext = " & objTrend.VariableScrollNext
End Sub
Public Sub SymbolLoading()
    Set                                objTrend                                =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

VariableScrollNextPage, TrendCmdTarget Property

Syntax VariableScrollNextPage = _String

Description Questa proprietà imposta o restituisce il nome della variabile associata alla funzione di "Prossima pagina" dell'oggetto Trend. Per impostare una nuova variabile e' necessario sospendere momentaneamente la registrazione impostando la proprietà Recording a False.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim sName As String
    Debug.Print "VariableScrollNextPage = " & objTrend.VariableScrollNextPage
    GetVariableNameFromList(sName)
    objTrend.Recording = False
    objTrend.VariableScrollNextPage = sName
    objTrend.Recording = True
    Debug.Print "VariableScrollNextPage = " & objTrend.VariableScrollNextPage
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

VariableScrollPrev, TrendCmdTarget Property

Syntax VariableScrollPrev = _String

Description Questa proprietà imposta o restituisce il nome della variabile associata alla funzione di "Precedente" dell'oggetto Trend. Per impostare una nuova variabile e' necessario sospendere momentaneamente la registrazione impostando la proprietà Recording a False.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim sName As String
    Debug.Print "VariableScrollPrev = " & objTrend.VariableScrollPrev
    GetVariableNameFromList(sName)
```

```

objTrend.Recording = False
objTrend.VariableScrollPrev = sName
objTrend.Recording = True
Debug.Print "VariableScrollPrev = " & objTrend.VariableScrollPrev
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

VariableScrollPrevPage, TrendCmdTarget Property

Syntax VariableScrollPrevPage = _String

Description Questa proprietà imposta o restituisce il nome della variabile associata alla funzione di "Pagina precedente" dell'oggetto Trend. Per impostare una nuova variabile e' necessario sospendere momentaneamente la registrazione impostando la proprietà Recording a False.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim sName As String
    Debug.Print "VariableScrollPrevPage = " & objTrend.VariableScrollPrevPage
    GetVariableNameFromList(sName)
    objTrend.Recording = False
    objTrend.VariableScrollPrevPage = sName
    objTrend.Recording = True
    Debug.Print "VariableScrollPrevPage = " & objTrend.VariableScrollPrevPage
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

VariableScrollStart, TrendCmdTarget Property

Syntax VariableScrollStart = _String

Description Questa proprietà imposta o restituisce il nome della variabile associata alla funzione di "Inizio" dell'oggetto Trend. Per impostare una nuova variabile e' necessario sospendere momentaneamente la registrazione impostando la proprietà Recording a False.

Parameter	Description

None	None
------	------

Result String

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Dim sName As String
    Debug.Print "VariableScrollStart = " & objTrend.VariableScrollStart
    GetVariableNameFromList(sName)
    objTrend.Recording = False
    objTrend.VariableScrollStart = sName
    objTrend.Recording = True
    Debug.Print "VariableScrollStart = " & objTrend.VariableScrollStart
End Sub
Public Sub SymbolLoading()
    Set                                objTrend                                =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

Vertical, TrendCmdTarget Property

Syntax Vertical = _Boolean

Description Questa proprietà imposta o restituisce la configurazione dello stile verticale o orizzontale dell'oggetto Trend. Accetta un valore Boolean.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetSubObject("TrendObj").GetObjectInterface
End Sub
Public Sub Click()
    objTrend.Vertical = Not objTrend.Vertical
    objTrend.Refresh
End Sub
```

ViewSamples, TrendCmdTarget Property

Syntax ViewSamples = _Long

Description Questa proprietà di sola lettura restituisce il numero di campionamenti visualizzati nella finestra di Trend.

Parameter	Description
-----------	-------------

None	None
------	------

Result Long

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    Debug.Print "ViewSamples = " & objTrend.ViewSamples
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

Visible, TrendCmdTarget Property

Syntax Visible(_IpszPenName) = _Boolean

Description Questa proprietà abilita o disabilita la visualizzazione della penna indicata come parametro.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.Visible(objTrend.GetPenNameFromList(0)) = Not
    objTrend.Visible(objTrend.GetPenNameFromList(0))
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

WeekBtnText, TrendCmdTarget Property

Syntax WeekBtnText = _String

Description Questa proprietà imposta o restituisce il testo che dovrà apparire sul "Pulsante Settimana" della barra dei pulsanti per la selezione dell'intervallo di tempo dell'oggetto Data Analysis. Se non viene specificato nulla, Movicon utilizzerà il testo di default.

Parameter	Description
None	None

Result String

Example:

```
Dim objDataAnalysis As TrendCmdTarget
Public Sub Click()
    If Not objDataAnalysis Is Nothing Then
        MsgBox "objDataAnalysis's WeekBtnText is " &
            objDataAnalysis.WeekBtnText, vbInformation, GetProjectTitle
        objDataAnalysis.WeekBtnText = "Week"
        objDataAnalysis.Refresh
    Else
        MsgBox "objDataAnalysis is nothing", vbExclamation, GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set objDataAnalysis =
        GetSynopticObject.GetSubObject("DataAnalysis").GetObjectInterface
End Sub
```

XGridColor, TrendCmdTarget Property

Syntax XGridColor = _Long

Description Questa proprietà imposta il codice del colore utilizzato per la griglia dell'asse X.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.XGridColor = RGB(245,0,0)
End Sub
Public Sub SymbolLoading()
    Set objTrend =
        GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

XGridLogarithmic, TrendCmdTarget Property

Syntax XGridLogarithmic = _Boolean

Description Questa proprietà abilita la scala logaritmica per la griglia X dell'area Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.XGridColor = Not objTrend.XGridColor
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

XGridUseNormalLine, TrendCmdTarget Property

Syntax XGridUseNormalLine = _Boolean

Description Questa proprietà imposta o restituisce il tipo di linea, normale o retinato, per la griglia X dell'area Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.XGridUseNormalLine = Not objTrend.XGridUseNormalLine
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

XY, TrendCmdTarget Property

Syntax XY = _Boolean

Description Questa proprietà imposta o restituisce lo stile del grafico sul Trend, se di tipo XY oppure di tipo Orizzontale. La modalità verticale del grafico non può essere selezionata tramite questa proprietà, ma è necessario utilizzare la proprietà "Vertical".

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.XY = Not objTrend.XY
End Sub
Public Sub SymbolLoading()
    Set                                objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

YearBtnText, TrendCmdTarget Property

Syntax YearBtnText = _String

Description Questa proprietà imposta o restituisce il testo che dovrà apparire sul "Pulsante Anno" della barra dei pulsanti per la selezione dell'intervallo di tempo dell'oggetto Data Analysis. Se non viene specificato nulla, Movicon utilizzerà il testo di default.

Parameter	Description
None	None

Result String

Example:

```

Dim objDataAnalysis As TrendCmdTarget
Public Sub Click()
    If Not objDataAnalysis Is Nothing Then
        MsgBox "objDataAnalysis's YearBtnText is " &
            objDataAnalysis.YearBtnText, vbInformation, GetProjectTitle
        objDataAnalysis.YearBtnText = "Year"
        objDataAnalysis.Refresh
    Else
        MsgBox "objDataAnalysis is nothing", vbExclamation, GetProjectTitle
    End If
End Sub

Public Sub SymbolLoading()
    Set                                objDataAnalysis
    GetSynopticObject.GetSubObject("DataAnalysis").GetObjectInterface
End Sub

```

YGridColor, TrendCmdTarget Property

Syntax YGridColor = _Long

Description Questa proprietà imposta il codice del colore utilizzato per la griglia dell'asse Y.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.YGridColor = RGB(245,0,0)
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

YGridLogarithmic, TrendCmdTarget Property

Syntax YGridLogarithmic = _Boolean

Description Questa proprietà abilita la scala logaritmica per la griglia Y dell'area Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.YGridColor = Not objTrend.YGridColor
End Sub
Public Sub SymbolLoading()
    Set objTrend =
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub
```

YGridUseNormalLine, TrendCmdTarget Property

Syntax YGridUseNormalLine = _Boolean

Description Questa proprietà imposta o restituisce il tipo di linea, normale o retinato, per la griglia Y dell'area Trend.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
```

```

        objTrend.YGridUseNormalLine = Not objTrend.YGridUseNormalLine
End Sub
Public Sub SymbolLoading()
    Set                                objTrend
    GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

ZoomBtnText, TrendCmdTarget Property

Syntax ZoomBtnText = _String

Description Questa proprietà imposta o restituisce un testo per il pulsante di comando Zoom dell'oggetto Trend/Data Analysis. Nel caso in cui la proprietà sia stata impostata con un nuovo valore, occorre usare il metodo "Refresh" per aggiornare graficamente l'oggetto.

Parameter	Description
None	None

Result String

Example:

```

Dim obj As TrendCmdTarget
Public Sub Click()
    If obj Is Nothing Then Set obj =
    GetSynopticObject.GetSubObject("objTrend").GetObjectInterface
    If obj.MeasureBtnText <> "" Then
        obj.ZoomBtnText = ""
    Else
        obj.ZoomBtnText = "ZOOM"
    End If
    obj.Refresh
End Sub

```

ZoomMode, TrendCmdTarget Property

Syntax ZoomMode = _Boolean

Description Questa proprietà imposta o restituisce la modalità zoom del grafico. Attivando la modalità zoom è possibile ingrandire una porzione di grafico mediante selezione dell'area col mouse. Disattivando la modalità zoom il grafico rimane ingrandito e lo si può scorrere col cursore. Usare la funzione ResetZoom() per tornare alle condizioni iniziali.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Dim objTrend As TrendCmdTarget
Public Sub Click()
    objTrend.ZoomMode = Not objTrend.ZoomMode
End Sub

```

```

Public Sub SymbolLoading()
    Set objTrend = GetSynopticObject.GetAbsoluteSubObject("Trend").GetObjectInterface
End Sub

```

1.17.9. UIInterface

1.18. Utilizzo della UIInterface

L'utilizzo dell'interfaccia di programmazione "UIInterface" è diretto, cioè non è necessario istanziare un oggetto di tipo "UIInterface" ma è possibile utilizzare le sue proprietà e metodi direttamente nel codice VB Script. In pratica tutte le proprietà e funzioni della "UIInterface" sono disponibili direttamente dall'Intellisense indipendentemente dal contesto VB Script in cui ci si trova.

Functions

AlphaNumericEntry, UIInterface Function

Syntax AlphaNumericEntry(_IpszVariableName, _nMaxChars)

Description Visualizza un display alfanumerico per l'impostazione di valori alfanumerici con mouse o touchscreen da associare alla variabile di Movicon. Indicato per PC senza tastiera.

Parameter	Description
IpszVariableName As String	Nome della variabile di Movicon.
nMaxChars As Integer	Opzionale.Valore indicante il numero massimo di caratteri digitabili.

Result Boolean

Example:

```

Sub Main
    ...
    AlphaNumericEntry("Setpoint_23", 3)
    ...
End Sub

```

ChooseColor, UIInterface Function

Syntax ChooseColor(_pColor)

Description Questa funzione visualizza una finestra con all'interno una serie di colori. La selezione di un colore chiude la finestra e assegna al parametro pColor la codifica RGB del colore selezionato.
L'uscita dalla tavola dei colori con il tasto Esc o con un clic del mouse all'esterno della stessa restituisce un valore False e il parametro pColor uguale a zero.
Il valore di ritorno indica l'avvenuta selezione del colore con True mentre indica un'errata selezione del colore con False.



Questa funzione non è supportata in Windows CE. (Se impostata restituisce sempre 'false')

Parameter	Description
pColor As Long	Variabile di appoggio per il valore del colore selezionato.

Result Boolean

Example:

```
Public Sub Click()  
    Dim pColor As Long  
    ' View select window (Visualizza finestra)  
    ChooseColor(pColor)  
    MsgBox("Color=" & pColor, ,"ChooseColor")  
    'GlobalContainerName enabled (Avendo abilitato la proprietà  
    GlobalContainerName a true)  
    ContainerDoc.BackColor = pColor  
End Sub
```

DoSomeEvents, UIInterface Function

Syntax DoSomeEvents(_nMaxEvents)

Description Questa funzione, a differenza della funzione DoEvents, consente di rilasciare parte delle risorse basic per eseguire altre funzioni all'interno dello stesso codice. Infatti, una volta entrati in un loop di programma in cui e' stata utilizzata la DoEvents, non sara' possibile eseguire altre funzioni fino a che non terminera' il loop stesso. Invece utilizzando la funzione DoSomeEvents e' possibile specificare, mediante il parametro nMaxEvents, il numero di loop da eseguire per poi verificare la necessita' di eseguire altre operazioni. Tale funzione risulta utile in quei codici basic associati ai disegni che eseguiti in un loop rilascino parte del tempo di esecuzione alla Interfaccia utente (User Interface).

Parameter	Description
nMaxEvents As Integer	Numero di loop da eseguire.

Result Boolean

Example:

```
Sub Main  
    While Not CBool(GetVariableValue("Bit"))  
        DoSomeEvents(100)  
    Wend  
    Debug.Print "End loop"  
End Sub
```

EditRuntimeUsers, UIInterface Function

Syntax EditRuntimeUsers()

Description Consente di editare gli utenti runtime utilizzando il tool apposito messo a disposizione da Movicon.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
...
    EditRuntimeUsers
...
End Sub
```

ExecuteCommand, UIInterface Function

Syntax ExecuteCommand(_IpszCommand)

Description Questa funzione permette di eseguire i comandi a disposizione nella lista comandi di Movicon all'interno di uno Script. Il parametro IpszCommand deve riportare una stringa contenente il tipo di comando e i parametri per l'esecuzione dello stesso. La sintassi dovrà essere del tipo:
"<CommandType ...parametri di comando ...">Tipo Comando</CommandType>"

Quando il carattere apice (') viene inserito direttamente nel parametro di un comando, accade che Movicon interpreti questo carattere come la chiusura del valore del parametro. Ad esempio la seguente riga non è corretta:

```
Public Sub Click()
..Show tooltip
    ExecuteCommand("<CommandType          action='1'          topic='Hello
'World'">Help</CommandType>")
End Sub
```

Una soluzione è quella di usare "'" al posto di "'":

```
Public Sub Click()
    Show tooltip
    ExecuteCommand("<CommandType          action='1'          topic='Hello
&apos;World&apos;'>Help</CommandType>")
End Sub
```

I parametri di comando possono essere:

Comandi Variabili

```
variable='NomeVariabile'          SecondVariable='DestinationVariable'
action='ActionType' strobe='StrobeValue' value='VariableValue' max='MaxVal'
min='MinVal' chars='MaxChars' alis="" pwd=""
```

dove:

```
ActionType
0 = Set Variable
1 = Reset Variable
2 = Toggle Variable
3 = Impulso Temporizzato
4 = Incrementa Variable
5 = Decrementa Variable
6 = Pad Alfanumerico
7 = Pad Numerico
8 = Appendi Valore
9 = Rimuovi Valore
10 = Cambia Segno +/-
11 = Appendi Decimale ON-OFF
```

12 = Trasferisci Valore
13 = Reset Statistiche
14 = Trasferisci Valore Minimo
15 = Trasferisci Valore Massimo
16 = Trasferisci Valore Medio
17 = Set Valore Tabella Stringhe
18 = Imposta Alias Sinottico

Comandi Sinottici

synoptic='SynopticName' action='ActionType' monitor='0'
parameter='ParameterList' x='Xposition' y='Yposition' width='Width'
height='Height' Caption='1' Border='1' Resizeable='0' SysMenu='0'
MinimizeBox='0' MaximizeBox='0' KeepPrintProportions='0' PageW='-1' PageH='-1'
LMargin='-1' RMargin='-1' TMargin='-1' BMargin='-1'

dove:

ActionType

0 = Apri Normale (Cambio pagina)

1 = Apri Modale (pop-up)

2 = Apri Frame (multimonitor)

3 = Apri in processo separato (Open Safe)

4 = Stampa

5 = Chiudi e ritorna a precedente

6 = Esegui Sinapsi

7 = Apri successivo (cambio pagina ID successivo)

8 = Apri precedente (cambio pagina ID precedente)

9 = Cattura e stampa

10 = Cattura e Salva

Parameterlist

valoreparametro1, valoreparametro2, ..., valoreparametroN

Comandi Script

script='ScriptName' action='ActionType' parameters='ParameterList'
newInstAllowed='0' timeout='TimeoutValue'

dove:

ActionType

0 = Avvia Normale

1 = Avviasincronizzato

2 = Avvia in processo separato (Safe Mode)

3 = Stop

4 = Scarica

Parameterlist

valoreparametro1, valoreparametro2, ..., valoreparametroN

Comandi Utente

action='ActionType' level='UserLevel'

dove:

ActionType

0 = Log On

1 = Log Off

2 = Edita Lista Utenti

Comandi Report

dlr='Data LoggerName' action='ActionType' x='XPosition' y='YPosition'
width='Width' height='Height' Toolbar='0' GroupTree='true/false' TemplateFile=''
DestinationFile=' ' Query=' ' ReferencePeriod='0' ExportFormat='0' SelectDate='0'
Sep='59' MaxPages='0' LeftMargin='-1' RightMargin='-1' TopMargin='-1'
BottomMargin='-1' PrintDlg='0' Landscape='0' Printer=' ' Recipient=' ' EmbRep=''

dove:

ActionType

0 = Visualizza Sincrono

1 = Stampa Sincrono

2 = Visualizza Report

3 = Stampa Report

4 = Muovi Primo

5 = Muovi Ultimo

6 = Muovi Precedente
7 = Muovi Prossimo
8 = Attiva
9 = Salva
10 = Elimina
11 = Query di Filtro
12 = Esegui Query
13 = Esporta Report
14 = Data Analysis
15 = Visualizza Report Testuale
16 = Stampa Report Testuale
17 = Salva Report Testuale
18 = Appendi Report Testuale
19 = Esporta Ricetta
20 = Importa Ricetta
21 = Esporta e Invia Email
22 = Visualizza Report Interno
23 = Stampa Report Interno
24 = Salva Report Interno
25 = Invia Report Interno
26 = Leggi Ricetta

ReferencePeriod

0 = Nessuno
1 = Oggi
2 = Ieri e Oggi
3 = Settimana Corrente
4 = Mese Corrente
5 = Anno Corrente
6 = Ultimi 7 Giorni
7 = Ultimi 30 Giorni
8 = Ultimi 60 Giorni
9 = Ultimi 90 Giorni
10 = Ultimo Anno
11 = Ultimi 2 Anni
12 = Ultimi 5 Anni
13 = Ultimi 10 Anni

ExportFormat

0 = Pdf
1 = Html
2 = Txt
3 = Csv
4 = Xls
5 = Mht
6 = Rtf
7 = Jpeg

SelectDate

0 = la finestra di selezione data non verrà mostrata
1 = verrà aperta la finestra per la selezione della data da applicare al filtro dei dati

Comandi Menù

menu='MenuName' x='-1' y='-1'

Comandi Sistema

action='ActionType'
parameters='Parameter' workingpath="" timeout='TimeoutValue'

dove:

ActionType

0 = Arresta Sistema Operativo
1 = Arresta Applicazione
2 = Esegui Applicazione
3 = Esegui Applicazione e Attendi
4 = Esegui File Audio
5 = Beep
6 = Speack
7 = Riavvio Sistema
8 = Mostra o Nasconde la Finestra di Output
9 = Tempo di Attesa

Parameter
Nome Applicazione

Comandi Help

action='ActionType' topic='Topic'

dove:

ActionType
0 = Topic
1 = Tooltip popup

Cambio Lingua

Language='Language'

Comandi Allarmi

*action='ActionType' AreaFilter="" Report='Report' Toolbar='true/false'
GroupTree='true/false' Period='Period' Duration='Duration' Date='Date'
TemplateFile="" DestinationFile="" Query="" MaxPages='10' PageHeight='-1'
PageWidth='-1' LeftMargin='-1' RightMargin='-1' TopMargin='-1' BottomMargin='-1'
PrintDlg='0' Landscape='0' Printer="" Recipient=""*

dove:

ActionType
0 = Riconosci Tutti
1 = Reset Tutti
2 = Abilita Suono
3 = Visualizza Report
4 = Stampa Report
5 = Esporta Report
6 = Visualizza Report Testuale
7 = Stampa Report Testuale
8 = Salva Report Testuale
9 = Appendi Report Testuale
10 = Esporta e Invia Email
11 = Visualizza Report Interno
12 = Stampa Report Interno
13 = Salva Report Interno
14 = Invia Report Interno
15 = Reset Statistiche

Report
OrderByDate
OrderByDuration
GroupByFrequency
GroupByThreshold

ReferencePeriod
0 = Period
1 = Today
2 = Yestarday or Today
3 = Current week
4 = Current month
5 = Current year
6 = Last 7 days
7 = Last 30 days
8 = Last 60 days
9 = Last 90 days
10 = Last 1 years
11 = Last 2 years
12 = Last 5 years
13 = Last 10 years

Comandi Evento

eventname='Event1'

Parameter	Description
IpszCommand As String	Riga di comando.

Result Boolean

Example1:

```
Public Sub Click()
    'Set VAR00001 = 1
    ExecuteCommand("<CommandType variable='VAR00001' action='0'
        strobe='0' value='1' max='100' min='0' _ chars=' '>Variable</CommandType>")
End Sub
```

Example2:

```
Public Sub Click()
    'Open Synopric MDI
    ExecuteCommand("<CommandType synoptic='LayOut' action='0' monitor='0'
        parameter='' x='-1' y='-1' width='0' height='0' Caption='true' Border='true'
        Resizeable='false' SysMenu='false' MinimizeBox='false' MaximizeBox='false'
        KeepPrintProportions='0' PageW='-1' PageH='-1' LMargin='-1' RMargin='-1'
        TMargin='-1' BMargin='-1'>Synoptic</CommandType>")
End Sub
```

Example3:

```
Public Sub Click()
    'Execute Script Normal
    ExecuteCommand("<CommandType script='Script1' action='0'
        parameters='1,2,3' newInstAllowed='0' timeout='3000'
        >Script</CommandType>")
End Sub
```

Example4:

```
Public Sub Click()
    'LogOn User
    ExecuteCommand("<CommandType action='0'
        level='5'>Users</CommandType>")
End Sub
```

Example5:

```
Public Sub Click()
    'View Embedded Report
    ExecuteCommand("<CommandType dlr='' action='22' Query='' ReferencePeriod='0'
        MaxPages='0' PageHeight='-1' PageWidth='-1' LeftMargin='-1' RightMargin='-1'
        TopMargin='-1' BottomMargin='-1' PrintDlg='0' Landscape='0' Printer=''
        EmbRep='Report1'>Report</CommandType>")
End Sub
```

Example6:

```
Public Sub Click()
    'Data Logger View Textual Report
    ExecuteCommand("<CommandType dlr='Data Logger' action='15'
        TemplateFile='Template.txt' DestinationFile='' Query='' MaxPages='0'
        >Report</CommandType>")
End Sub
```

Example7:

```
Public Sub Click()
    'Launch calc.exe
    ExecuteCommand("<CommandType action='2' parameters='calc.exe'
        timeout='5000'>System</CommandType>")
    'Close Supervisor
    ExecuteCommand("<CommandType action='1' parameters=''
        timeout='5000'>System</CommandType>")
End Sub
```

Example8:

```
Public Sub Click()
    'Show tooltip
```

```

        ExecuteCommand("<CommandType    action='1'    topic='Welcome    to
Movicon'>Help</CommandType>")
End Sub

```

Example9:

```

Public Sub Click()
    'Change language to English, that has been previously defined in the String Table
    ExecuteCommand("<CommandType
action='English'>Language</CommandType>")
End Sub

```

Example10:

```

Public Sub Click()
    'AckAll Alarms
    ExecuteCommand("<CommandType    action='0'    Report=""    Toolbar='true'
GroupTree='false' Period="" Duration="" Date="">Alarm</CommandType>")
End Sub

```

Example11:

```

Public Sub Click()
    'Alarms View Textual Report
    ExecuteCommand("<CommandType    action='6'    TemplateFile='Template.txt'
DestinationFile="" Query="" MaxPages='0' >Alarm</CommandType>")
End Sub

```

GetLastActiveSynoptic, UIInterface Function

Syntax GetLastActiveSynoptic

Description Restituisce una stringa contenente il nome dell'ultimo sinottico aperto in visualizzazione.

Parameter	Description
None	None

Result String

Example:

```

Public Sub Click()
    Dim sResult As String
    sResult = GetLastActiveSynoptic
    Debug.Print sResult
End Sub

```

GetMonitorCoordinates, UIInterface Function

Syntax GetMonitorCoordinates(_nMonitor, _pLeft, _pTop, _pRight, _pBottom)

Description Questa funzione, studiata per i sistemi multimonitor, puo' essere utilizzata anche nei sistemi a un solo monitor. Restituisce le coordinate in pixel dell'area visualizzata dal monitor il cui numero viene passato come parametro. L'indice per i monitor inizia dal valore 0.



Questa funzione è supportata solo in parte su Windows CE. (AmMESSo solo il parametro nMonitor=1, altrimenti ritorna sempre 'false')

Parameter	Description
nMonitor	Numero del monitor.
pLeft	Coordinata iniziale x.
pTop	Coordinata iniziale y.
pRight	Coordinata finale x.
pBottom	Coordinata finale y.

Result Boolean

Example:

```
Public Sub Click()

Dim nMonitor As Integer

Dim pLeft As Variant
Dim pTop As Variant
Dim pRight As Variant
Dim pBottom As Variant

nMonitor = 0
GetMonitorCoordinates(nMonitor, pLeft, pTop, pRight, pBottom)

Debug.Print pLeft
Debug.Print pTop
Debug.Print pRight
Debug.Print pBottom
End Sub
```

GetNumMonitors, UIInterface Function

Syntax GetNumMonitors()

Description Questa funzione restituisce un valore Integer indicante il numero di monitor installati nel sistema.



Questa funzione non è supportata in Windows CE. (Se impostata restituisce sempre 1)

Parameter	Description
None	None

Result Integer

Example:

```
Public Sub Click()
    Debug.Print GetNumMonitors
End Sub
```


GetPasswordFromLevel, UIInterface Function

Syntax GetPasswordFromLevel(_nLevel)

Description Funzione di richiesta Password su Livello di Utenza. La funzione attiverà la richiesta del livello di utenza password di Movicon specificato come parametro. Movicon, se abilitata la gestione delle password, visualizzerà la finestra per la richiesta dell'utente e della password.

Parameter	Description
nLevel As Long	Livello utente.

Result Boolean

Example:

```
Sub Main
    '...
    'questa procedura richiede il livello di password da abilitare mediante una dialog
    (vedere manuale Win Wrap Basic per ulteriori informazioni sulle finestre di
    Dialogo), 'quindi abilita l'utente che si dichiara previa richiesta di password.
    Begin Dialog UserDialog 200,150
        Text 10,10,180,15,"Livello password"
        OptionGroup .options
        OptionButton 10,30,180,15,"Option &1"
        OptionButton 10,45,180,15,"Option &2"
        OptionButton 10,60,180,15,"Option &3"
        OptionButton 10,75,180,15,"Option &4"
        OptionButton 10,90,180,15,"Option &5"
        OKButton 80,120,40,20
    End Dialog
    Dim level As Integer
    Dim dlg As UserDialog
    dlg.options = 0
    Dialog dlg
    level = dlg.options + 1
    GetPasswordFromLevel(level)
    '...
End Sub
```

GetPasswordFromUser, UIInterface Function

Syntax GetPasswordFromUser(_IpszUser,[bLockUserName])

Description Tramite questa funzione è possibile richiedere il LogOn di un utente specifico di Movicon. Il nome dell'utente che dovrà eseguire il LogOn viene specificato nel parametro "IpszUser" della funzione. La funzione aprirà quindi la finestra di LogOn di Movicon dove sarà necessario inserire il nome utente e password. La differenza rispetto al LogOn classico è che in questo caso soltanto l'utente specificato nel parametro della funzione potrà eseguire l'autenticazione. Il tentativo di autenticare un utente differente fallirà. Se nel sistema è già loggato un utente l'esecuzione della funzione non produrrà nessun effetto e non verrà aperta la finestra di LogOn.



dalla versione 11.4.1151 è stato aggiunto un nuovo parametro opzionale, di tipo boolean, che permette di forzare il lock del nome utente nella finestra di richiesta password.

Parameter	Description
-----------	-------------

IpszUser As String	Nome Utente per il quale richiedere l'autenticazione.
--------------------	---

Result Boolean

Example:

```
Public Sub Click()
    Debug.Print GetPasswordFromUser("User1")
End Sub
```

GetVariableNameFromList, UIInterface Function

Syntax GetVariableNameFromList(_IpszVarName)

Description Consente di selezionare in runtime una variabile dal Database variabili di Movicon aprendo l'apposita finestra del Database variabili. La funzione restituisce nel parametro stringa che indica il nome della variabile selezionata dall'operatore, ed inoltre restituisce un parametro boolean che sarà True in caso di successo, False in caso di fallimento.



Questa funzione non è supportata in Windows CE (Se impostata restituisce sempre "false").



Questa funzione non può essere eseguita all'interno di una risorsa Basic Script. Può soltanto essere eseguita all'interno del codice del sinottico o degli oggetti, a meno che il sinottico non sia impostato come "Esegui in Altro Thread". La funzione può anche essere usata nel dropping code dei template.

Parameter	Description
IpszVarName As String	Nome variabile.

Result Boolean

Example:

```
Sub Main
    Dim sName As String
    GetVariableNameFromList (sName)
    Debug.Print sName
End Sub
```

GetWindowPos, UIInterface Function

Syntax GetWindowPos(_nX, _nY, _nWidth, _nHeight, _nShow)

Description Permette di reperire informazioni in merito alla posizione e alla dimensione della finestra contenente il sinottico.



Permette di reperire informazioni in merito alla posizione e alla dimensione della finestra di Movicon rispetto allo schermo.

Parameter	Description
nX As Integer	Variabile di appoggio per la coordinata X angolo alto sinistra.
nY As Integer	Variabile di appoggio per la coordinata Y angolo alto sinistra.
nWidth As Integer	Variabile di appoggio per la dimensione larghezza.
nHeight As Integer	Variabile di appoggio per la dimensione altezza.
nShow As Integer	Variabile di appoggio per la modalità di visualizzazione del sinottico.

Result Boolean

Example:

```
Public Sub Click()
    Dim nX As Variant
    Dim nY As Variant
    Dim nWidth As Variant
    Dim nHeight As Variant
    Dim nShow As Variant
    GetWindowPos(nX , nY , nWidth , nHeight , nShow )
    MsgBox "nX = " & nX & " nY = " & nY & "nWidth = " & nWidth & "nHeight = " &
    nHeight & "nShow = " & nShow , vbOkOnly, GetProjectTitle
End Sub
```

HideLongOperationDialog, UIInterface Function

Syntax HideLongOperationDialog

Description Questa Funzione chiude la finestra di dialogo aperta con la funzione "ShowLongOperationDialog".

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
    ...
    ShowLongOperationDialog(GetProjectTitle, True)
    ...
    HideLongOperationDialog
    ...
End Sub
```

IsLongOperationAborted, UIInterface Function

Syntax IsLongOperationAborted

Description Questa funzione verifica se una LongOperation è stata annullata (tramite il tasto Annulla delle Dialog Box ShowLongOperation) oppure no: ritorna il valore booleano True se l'operazione termina senza interruzione. Se l'operazione è interrotta manualmente il valore di ritorno è False.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
...
Dim Abort As boolean
Abort = IsLongOperationAborted
If Not Abort = 1 then
    HideLongOperationDialog
'....
End If
'...
End Sub
```

LoadPicture, UIInterface Function

Syntax LoadPicture(_lpszFileName)

Description Restituisce un oggetto contenente l'immagine specificata dal parametro lpszFileName.



Questa funzione non è supportata in Windows CE. (Se impostata restituisce sempre ' null')

Parameter	Description
lpszFileName As String	Nome del file immagine.

Result Object

Example:

```
Public Sub Click()
...
Dim imgX As ListImage
' Add images to ListImages collection.
Set imgX = ImageList1.ListImages.Add(, "rocket",
LoadPicture("D:\Documenti\Test Progetti\x_animata.gif"))
...
End Sub
```

LogoffActiveUser, UIInterface Function

Syntax LogoffActiveUser()

Description Questa funzione forza il logoff dell'utente attualmente loggato nell'applicazione. La funzione non è disponibile per gli utenti Web Client.

Parameter	Description
None	None

Result Boolean

Example:

```
Sub Main
...
LogoffActiveUser
...
End Sub
```

NumericEntry, UIInterface Function

Syntax NumericEntry (_IpszVariableName, _nLowLimit, _nHighLimit)

Description Visualizza sullo schermo un pad numerico per l'introduzione di numeri come valori per le variabili di Movicon. Questa funzione è appositamente concepita per sistemi privi di tastiera (touchscreen).

Parameter	Description
IpszVariableName As Integer	Nome della variabile desiderata.
nLowLimit As Variant	Opzionale. Valore di controllo come limite inferiore per la variabile.
nHighLimit As Variant	Opzionale. Valore di controllo come limite superiore per la variabile.

Result Boolean

Example:

```
Sub Main
...
Dim Low As Integer, High As Integer
Low = 10
High = 100
NumericEntry("VAR0001", Low, High)
...
End Sub
```

OpenModalSynoptic, UIInterface Function

Syntax OpenModalSynoptic(IpszSynopticName, IpszParameterFile, nX, nY, nWidth, nHeight)

Description Apre una finestra sinottico del progetto in modo modale. Una finestra modale e' una finestra secondaria che cattura tutti gli input dall'utente fino a che non viene terminata.

Parameter	Description
IpszSynopticName As String	Nome finestra sinottico
IpszParameterFile As String	File per parametrizzazione sinottico (vedi Synoptic Interface)
nX As Long	Coordinata X finestra sinottico
nY As Long	Coordinata Y finestra sinottico
nWidth As Long	Larghezza della finestra sinottico
nHeight As Long	Altezza della finestra sinottico

Result Boolean

Example:

```
Public Sub Click()  
...  
    OpenModalSynoptic("Sinottico1", "Param.txt", 100, 100, 600, 300)  
...  
End Sub
```

OpenSynoptic, UIInterface Function

Syntax OpenSynoptic(_IpszSynopticName, _nShow)

Description Funzione di apertura di un sinottico del progetto.

Il parametro nShow non viene in realtà più considerato. Il sinottico viene sempre aperto in modo normale. Il parametro è rimasto per compatibilità verso la versione precedente di Movicon in modo che eventualmente si possa copiare il codice script.



Su Windows CE il valore di ritorno "True" indica che il comando è stato preso in consegna e verrà eseguito. Questo significa ad esempio che il valore di ritorno potrà essere "True" anche se poi il sinottico richiesto non esiste. Questo funzionamento è dovuto al fatto che il comando di apertura del sinottico viene di fatto eseguito con un ritardo per permettere al sinottico attivo di essere scaricato dalla memoria (se Delay = 0) prima dell'apertura di quello richiesto. Questo al fine di ottimizzare al meglio la gestione della memoria.

Parameter	Description
IpszSynopticName As String	Nome del sinottico da aprire.
nShow As Integer	Parametro non utilizzato.

Result Boolean

Example:

```

Sub Main
...
OpenSynoptic("Synoptic1", 1)
...
End Sub

```

OpenSynopticEx, UIInterface Function

Syntax OpenSynopticEx(lpszSynopticName, nShow, lpszParameterFile, nMonitor, bNotMDIFrame, bModal)

Description Apre una finestra sinottico del progetto in differenti modi a seconda dei parametri specificati. Il parametro nMonitor indica, nei sistemi multimonitor, in quale monitor deve apparire la finestra; il parametro bNotMDIFrame abilita l'apertura della finestra al di fuori del frame principale; il parametro bModal abilita l'apertura della finestra in modo modale. Il parametro nShow NON è utilizzato ma è stato lasciato per compatibilità con versioni precedenti.

Quando un sinottico viene aperto sul monitor secondario (parametro nMonitor > 1) la finestra verrà sempre aperta massimizzata. Quando un sinottico viene aperto nello stesso frame MDI di Movicon (parametro bNotMDIFrame = False) e non è modale (parametro bModal = False) la finestra verrà sempre aperta massimizzata. Quando un sinottico viene aperto nel monitor principale (parametro nMonitor = 0) ed non è nello stesso frame MDI di Movicon (parametro bNotMDIFrame = True) la finestra verrà aperta con le dimensioni di default impostate nel sinottico e alle coordinate x=0 e Y=0.

Parameter	Description
lpszSynopticName As String	Nome finestra sinottico
nShow As Integer	Parametro non utilizzato
lpszParameterFile As String	File per parametrizzazione sinottico
nMonitor As Integer	Indice del monitor in cui aprire la finestra
bNotMDIFrame As Boolean	Apre la finestra al di fuori del frame di Movicon
bModal As Boolean	Apre la finestra in modo modale (richiede bNotMDIFrame = True)

Result Boolean

Example:

```

Public Sub Click()
...
OpenSynopticEx("Sinottico1", 1, "Param.txt", 0, True, True)
...
End Sub

```

OpenSynopticParameter, UIInterface Function

Syntax OpenSynopticParameter(_lpszSynopticName, _nShow, _lpszParameterFile)

Description Apre un sinottico del progetto specificando il file di parametrizzazione nel parametro lpszParameterFile. Il parametro nShow NON è utilizzato ma è stato lasciato per compatibilità con versioni precedenti. Utilizzando questo comando la finestra verrà sempre aperta massimizzata.

Parameter	Description
lpszSynopticName As String	Nome del sinottico da aprire.
nShow As Integer	Parametro non utilizzato.
lpszParameterFile As String	File di parametrizzazione.

Result Boolean

Example:

```
Sub Main
...
OpenSynopticParameter("Sinottico1", 0, "Param.txt")
...
End Sub
```

SayThis, UIInterface Function

Syntax SayThis(_lpszSpeechText)

Description Realizza la sintesi vocale della stringa di testo passata come parametro. Questa funzione utilizza le SAPI di Windows e necessita di una libreria di fonemi relativa alla lingua che si intende utilizzare.



Questa funzione non è supportata in Windows CE. (Se impostata restituisce sempre 'false')

Parameter	Description
lpszSpeechText As String	Testo passato al sintetizzatore vocale.

Result Boolean

Example:

```
Public Sub Click()
Dim lpszText As String

' Select the text to speech
lpszText = InputBox("Text to Speech:", "AgentSpeak", "I'm Robby", 100, 100)
' Send text to Agent
SayThis(lpszText)
End Sub
```

SelectResourceFromList, UIInterface Function

Syntax SelectResourceFromList(_lpszTitle,_lpszResType)

Description Visualizza una finestra con l'elenco delle risorse del progetto, appartenenti al tipo individuato dal parametro nResType. Il valore di ritorno indica il nome selezionato

dalla lista dopo la chiusura della finestra con OK. Se la finestra viene chiusa con Annulla il valore di ritorno è una stringa nulla.
Questa funzione viene normalmente utilizzata nel codice dei template per personalizzare l'inserimento di simboli dalla libreria. Per maggiori informazioni a riguardo, consultare l'help dell'evento OnCustomizeSymbol.

Tipo di risorsa:
enum_SYNOPTIC_RESOURCES (valore 1) = Sinottici
enum_SCRIPT_RESOURCES (valore 2) = Basic Script
enum_ACCELERATOR_RESOURCES (valore 4) = Acceleratori
enum_MENU_RESOURCES (valore 8) = Menu
enum_PARAMETER_RESOURCES (valore 16) = File di parametrizzazione
enum_REPORT_RESOURCES (valore 32) = Report Interno



Questa funzione non è supportata in Windows CE. (Se impostata restituisce sempre una stringa vuota)



Questa funzione non può essere eseguita all'interno di una risorsa Basic Script. Può soltanto essere eseguita all'interno del codice del sinottico o degli oggetti e nel dropping code dei template.

Parameter	Description
IpszTitle As String	Titolo finestra.
IpszResType As Integer	Tipo di risorsa. E' possibile concatenare più tipi di risorsa, ad esempio scrivendo: enum_SYNOPTIC_RESOURCES Or enum_SCRIPT_RESOURCES in questo modo la finestra di selezione della risorsa verrà aperta con due tab, quella di selezione sinottici e quella di selezione basic script.

Result String

Example:
Sub OnCustomizeSymbol (...)
 Dim sSyn As String
 sSyn = **SelectResourceFromList** ("", enum_SYNOPTIC_RESOURCES Or
 enum_SCRIPT_RESOURCES)
 If sSyn = "" Then
 bRet = False
 Else
 Prop("Syn") = sSyn
 bShowPropInsp = False
 End If
End Sub

SetDefPrinterOrient, UIInterface Function

Syntax SetDefPrinterOrient(_bLandScape)

Description Questa funzione consente di impostare l'orientamento della stampante di sistema predefinita. Se impostato a True l'orientamento sarà orizzontale, se impostato a False l'orientamento sarà verticale.



Questa proprietà non è supportata in Windows CE.

Parameter	Description
bLandscape As Boolean	Orientamento orizzontale della stampa.

Result None

Example:

```
Public Sub Click()
    ...
    SetDefPrinterOrient(True)
    ...
End Sub
```

SetRedraw, UIInterface Function

Syntax SetRedraw(_bSet)

Description Questa funzione permette di abilitare o disabilitare la gestione grafica dei sinottici aperti. Il parametro bSet identifica uno dei due modi. L'utilizzo è molto delicato in quanto la disabilitazione comporta anche la perdita da parte dell'utente della possibilità di interagire con i vari oggetti o simboli presenti nella pagina. La funzione è utile per disabilitare la visualizzazione di lunghe operazioni grafiche sul progetto e renderle attive solo al momento che sono state ultimate.

Parameter	Description
bSet As Boolean	Valore di abilitazione.

Result None

Example:

```
Sub Main
    If MsgBox("Do you suspend redraw on the synoptic ?", vbYesNo, "SetRedraw") =
        vbYes Then
        SetRedraw(False)
        Wait 5
    End If
    SetRedraw(True)
End Sub
```

SetWindowPos, UIInterface Function

Syntax SetWindowPos(_nX, _nY, _nWidth, _nHeight, _nShow)

Description Questa funzione permette di spostare e ridimensionare la finestra di Movicon.

Il parametro nShow può assumere i seguenti valori:

0= nasconde
1= ripristina
2= riduce a icona
3= massimizza

Parameter	Description
-----------	-------------

nX As Integer	Coordinata X angolo in alto a sinistra.
nY As Integer	Coordinata Y angolo in alto a sinistra.
nWidth As Integer	Larghezza finestra.
nHeight As Integer	Altezza finestra.
nShow As Integer	Definizione della modalità di visualizzazione: 0= nasconde 1= ripristina 2= riduce a icona 3= massimizza

Result Boolean

Example:

```
Sub Main
...
SetWindowPos(100,100,450,450,1)
...
End Sub
```

ShowHTMLDialog, UIInterface Function

Syntax ShowHTMLDialog(_lpszURL, _varArgIn, _varArgOut)

Description Visualizza una finestra di dialogo contenente la pagina HTML specificata nel parametro lpszURL. Tramite i parametri varArgIn e varArgOut è possibile passare e ricevere dei valori di qualsiasi tipo. Riferirsi a DHTML per maggiori informazioni sull'utilizzo di questi parametri all'interno della pagina.
La finestra di dialogo aperta è modale, quindi non è possibile accedere ad altri comandi al di fuori, fino alla sua chiusura.
Il valore di ritorno indica se la finestra di dialogo è stata aperta con successo (True) oppure no (False).



Questa funzione non è supportata su Windows CE. In questo contesto la funzione non eseguirà nessun comando e ritornerà il valore "False".



I due parametri varArgIn e varArgOut devono essere gestiti dentro la pagina html tramite le funzioni window.dialogArguments e window.returnValue.
Per maggiori informazioni può anche consultare la documentazione online sull'uso del DHTML, Java Script, ecc.
([http://msdn.microsoft.com/en-us/library/ms533723\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/ms533723(VS.85).aspx)
<http://msdn.microsoft.com/en-us/library/ms534371.aspx>)

Sul sito Progea è anche disponibile l'esempio "ES_ShowHTMLDialog".

Parameter	Description
lpszURL As String	Percorso e nome (compresa l'estensione) della pagina HTML da aprire
varArgIn As Variant	Opzionale: Valori di input passati alla pagina HTML
varArgOut As Variant	Opzionale: Valori di output ricevuti dalla pagina HTML

Result Boolean

Example:

```
Sub Main
    Dim IpszURL As String
    Dim varArgIn As Variant
    Dim varArgOut As Variant

    IpszURL = "C:\Documenti\Index.html"
    varArgIn = 5
    ShowHTMLDialog(IpszURL, varArgIn, varArgOut)
    MsgBox ("varArgIn=" & varArgIn & vbCrLf & "varArgOut=" & varArgOut,
    "ShowHTMLDialog")
End Sub
```

ShowLongOperationDialog, UIInterface Function

Syntax ShowLongOperationDialog(_IpszTitle, _bCancelOperation)

Description Questa Funzione mostra una finestra di dialogo modale tramite la quale si può visualizzare un messaggio per avvisare l'utente che una determinata operazione è in corso. Il parametro "bCancelOperation", se impostato a True, consente di visualizzare il tasto Cancel sulla finestra, per poter eseguire la chiusura della finestra stessa anche prima che questo venga fatto da codice tramite la funzione "HideLongOperationDialog". Occorre tener presente che alla pressione del tasto Cancel sulla ShowLongOperationDialog si attiva una memoria interna che consente di testare con la funzione "IsLongOperationAborted" se la ShowLongOperationDialog è stata annullata. Fino a quando la memoria non si azzerà la ShowLongOperationDialog non sarà più visualizzata per quel contesto (risorsa basic script, disegno in un sinottico, ecc.). La memoria si resetta quando viene utilizzato il metodo "HideLongOperationDialog".

Parameter	Description
IpszTitle As String	Titolo della finestra.
bCancelOperation As Boolean	Presenza tasto 'Cancel' per la chiusura della finestra.

Result Boolean

Example:

```
Sub Main
    ...
    ShowLongOperationDialog("Movicon", True)
    ...
    HideLongOperationDialog
End Sub
```

ShowMenu, UIInterface Function

Syntax ShowMenu(_IpszMenu)

Description Consente di visualizzare una risorsa Menù presente nelle risorse del progetto. La finestra menù verrà visualizzata in corrispondenza del puntatore del mouse.

Parameter	Description
IpszMenu As String	Menù da visualizzare.

Result Boolean

Example:

```
Sub Main
    ...
    ShowMenu("Menu1")
    ...
End Sub
```

Properties

ActiveLanguage, UIInterface Property

Syntax ActiveLanguage = _String

Description Permette di impostare o leggere la lingua attiva.

Parameter	Description
None	None

Result String

Example:

```
Public Sub Click()
    Dim sLanguage As String
    sLanguage = ActiveLanguage
    Debug.Print sLanguage
End Sub
```

MainVisible, UIInterface Property

Syntax MainVisible = _Boolean

Description Se impostata al valore booleano 0, rende la finestra dell'applicazione Movicon invisibile. Impostata in seguito al valore 1, la verrà ripristinata la finestra di Movicon (visibile).

Parameter	Description
None	None

Result Boolean

Example:

```

Sub Main
    Debug.Print "MainVisible: " & MainVisible
    MainVisible = False
    Wait 5
    MainVisible = True
End Sub

```

1.18.1. UserAndGroupCmdTarget

Functions

GetActiveUserObject, UserAndGroupCmdTarget Function

Syntax GetActiveUserObject()

Description Questa funzione permette di reperire l'oggetto utente relativo all'utente attualmente loggato. Una volta reperito l'oggetto saranno disponibili tutti i metodi e le proprietà descritte nell'interfaccia UserCmdTarget. Se nessun utente ha effettuato il login, la funzione restituisce un oggetto Nothing.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo UserCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```

Option Explicit
Public Sub Click()
    Dim objUserAndGroup As UserAndGroupCmdTarget
    Dim objUser As UserCmdTarget

    Set objUserAndGroup = GetUserAndGroup
    If Not objUserAndGroup Is Nothing Then
        Set objUser = objUserAndGroup.GetActiveUserObject
        If Not objUser Is Nothing Then
            MsgBox(objUser.Name & " is Logged In", vbOkOnly,
                GetProjectTitle)
            Set objUser = Nothing
        End If
        Set objUserAndGroup = Nothing
    End If
End Sub

```

GetDesignGroupAtPos, UserAndGroupCmdTarget Function

Syntax GetDesignGroupAtPos(nPosition)

Description Questa funzione permette di reperire, nella lista dei gruppi utenti definiti in sviluppo, l'oggetto "gruppo utenti" identificato dalla posizione nPosition del parametro. Una volta reperito l'oggetto saranno disponibili tutti i metodi e le proprietà dell'interfaccia UserGroupCmdTarget.

L'oggetto restituito è diverso da Nothing solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore. Questa funzione infatti, come la "GetObject()", ha una protezione che impedisce l'accesso ai livelli inferiori.

Parameter	Description
nPosition As Integer	Indice di riferimento del gruppo i-esimo nella lista dei gruppi utenti definiti in sviluppo

Result Object
Viene restituito un oggetto di tipo UserGroupCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

Option Explicit
Public Sub Click()

```
Dim objUserAndGroup As UserAndGroupCmdTarget
Dim objGroup As UserGroupCmdTarget
Dim nNumDesignGroups As Integer

Set objUserAndGroup = GetUserAndGroup
If Not objUserAndGroup Is Nothing Then
    objUserAndGroup.LogonUser("Administrator", "Administrator")
End If

nNumDesignGroups = objUserAndGroup.GetNumDesignGroups()
MsgBox "Number of design groups = " & nNumDesignGroups

For i = 0 To nNumDesignGroups - 1
    Set objGroup = objUserAndGroup.GetDesignGroupAtPos(i)
    If Not objGroup Is Nothing Then
        MsgBox "Design group at pos " & i & " is: " & objGroup.Name
    End If
Next
End Sub
```

GetGroupObject, UserAndGroupCmdTarget Function

Syntax GetGroupObject(_IpszGroupName)

Description Questa funzione permette di reperire l'oggetto gruppo utenti referenziato dal parametro IpszGroupName. Una volta reperito l'oggetto saranno disponibili tutti i metodi e le proprietà dell'interfaccia UserGroupCmdTarget.

L'oggetto restituito è diverso da Nothing solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore. Questa funzione infatti, come la "GetObject()", ha una protezione che impedisce l'accesso ai livelli inferiori.



La funzione GetGroupObject consente di recuperare il riferimento anche di un gruppo creato in runtime. Il gruppo viene ricercato prima fra la lista di quelli configurati nel progetto e poi fra quelli creati in runtime. Di conseguenza se c'è un gruppo con lo stesso nome configurato in sviluppo e in runtime, viene recuperato il riferimento di quello configurato in sviluppo.

Parameter	Description
IpszGroupName As String	Nome del Gruppo da reperire

Result Object
Viene restituito un oggetto di tipo UserGroupCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserGroupCmdTarget
    Dim obj3 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetGroupObject("Group1")
        Set obj3 = obj1.GetUserObject("Group1","Ut1")
        Debug.Print obj1.EnableAutoLogoff
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.Description
            Set obj2 = Nothing
        End If
        If Not obj3 Is Nothing Then
            Debug.Print obj3.Description
            Set obj3 = Nothing
        End If
    End If
End Sub
```

GetNumActiveUsers, UserAndGroupCmdTarget Function

Syntax GetNumActiveUsers()

Description Questa funzione restituisce il numero di utenti attualmente loggati.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Dim numUsers As Long
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        numUsers = myObject.GetNumActiveUsers
        MsgBox(numUsers & " Users are Logged In", vbOkOnly, GetProjectTitle)
    End If
End Sub
```


GetNumDesignGroups, UserAndGroupCmdTarget Function

Syntax GetNumDesignGroups()

Description Questa funzione restituisce il numero di gruppi di utenti definiti in sviluppo

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit  
Public Sub Click()
```

```
Dim objUserAndGroup As UserAndGroupCmdTarget  
Dim objGroup As UserGroupCmdTarget  
Dim nNumDesignGroups As Integer
```

```
    Set objUserAndGroup = GetUserAndGroup  
    If Not objUserAndGroup Is Nothing Then  
        objUserAndGroup.LogonUser("Administrator", "Administrator")  
    End If  
  
    nNumDesignGroups= objUserAndGroup.GetNumDesignGroups()  
    MsgBox "Number of design groups = " & nNumDesignGroups  
    For i = 0 To nNumDesignGroups - 1  
        Set objGroup = objUserAndGroup.GetDesignGroupAtPos(i)  
        If Not objGroup Is Nothing Then  
            MsgBox "Design group at pos " & i & " is: " & objGroup.Name  
        End if  
    Next  
End Sub
```

GetNumRuntimeGroups, UserAndGroupCmdTarget Function

Syntax GetNumRuntimeGroups()

Description Questa funzione restituisce il numero di gruppi di utenti definiti in runtime tramite il tool "EditUser".

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit  
Public Sub Click()  
Dim objUserAndGroup As UserAndGroupCmdTarget  
Dim objGroup As UserGroupCmdTarget
```

```

Dim nNumRuntimeGroups As Integer

Set objUserAndGroup = GetUserAndGroup

    If Not objUserAndGroup Is Nothing Then
        objUserAndGroup.LogonUser("Administrator", "Administrator")
    End If
    nNumRuntimeGroups= objUserAndGroup.GetNumRuntimeGroups()
    MsgBox "Number of Runtime groups = " & nNumRuntimeGroups
    For i = 0 To nNumRuntimeGroups - 1
        Set objGroup = objUserAndGroup.GetRuntimeGroupAtPos(i)
        If Not objGroup Is Nothing Then
            MsgBox "Runtime group at pos " & i & " is: " & objGroup.Name
        End if
    Next
End Sub

```

GetRuntimeGroupAtPos, UserAndGroupCmdTarget Function

Syntax GetRuntimeGroupAtPos(nPosition)

Description Questa funzione permette di reperire, nella lista dei gruppi utenti definiti in runtime, l'oggetto "gruppo utenti" identificato dalla posizione nPosition del parametro. Una volta reperito l'oggetto saranno disponibili tutti i metodi e le proprietà dell'interfaccia UserGroupCmdTarget. L'oggetto restituito è diverso da Nothing solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore. Questa funzione infatti, come la "GetGroupObject()", ha una protezione che impedisce l'accesso ai livelli inferiori.

Parameter	Description
nPosition As Integer	Indice di riferimento del gruppo i-esimo nella lista dei gruppi utenti definiti in runtime

Result Object
Viene restituito un oggetto di tipo UserGroupCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```

Option Explicit
Public Sub Click()
Dim objUserAndGroup As UserAndGroupCmdTarget
Dim objGroup As UserGroupCmdTarget
Dim nNumRuntimeGroups As Integer

    Set objUserAndGroup = GetUserAndGroup
    If Not objUserAndGroup Is Nothing Then
        objUserAndGroup.LogonUser("Administrator", "Administrator")
    End If

    nNumRuntimeGroups= objUserAndGroup.GetNumRuntimeGroups()
    MsgBox "Number of Runtime groups = " & nNumRuntimeGroups

    For i = 0 To nNumRuntimeGroups - 1
        Set objGroup = objUserAndGroup.GetRuntimeGroupAtPos(i)
        If Not objGroup Is Nothing Then
            MsgBox "Runtime group at pos " & i & " is: " & objGroup.Name
        End if
    Next
End Sub

```

GetUserObject, UserAndGroupCmdTarget Function

Syntax GetUserObject(_lpszGroupName, _lpszUserName)

Description Questa funzione permette di reperire l'oggetto gruppo utenti referenziato dal parametro lpszUserName. Una volta reperito l'oggetto saranno disponibili tutti i metodi e le proprietà dell'interfaccia UserCmdTarget. L'oggetto restituito è diverso da Nothing solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore. Questa funzione infatti, come la "GetGroupObject()", ha una protezione che impedisce l'accesso ai livelli inferiori.

Parameter	Description
lpszGroupName As String	Nome del Gruppo di appartenenza
lpszUserName As String	Nome dell'Utente da reperire

Result Object
Viene restituito un oggetto di tipo UserCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserGroupCmdTarget
    Dim obj3 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetGroupObject("Group1")
        Set obj3 = obj1.GetUserObject("Group1","Ut1")
        Debug.Print obj1.EnableAutoLogoff
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.Description
            Set obj2 = Nothing
        End If
        If Not obj3 Is Nothing Then
            Debug.Print obj3.Description
            Set obj3 = Nothing
        End If
    End If
End Sub
```

GetXMLSettings, UserAndGroupCmdTarget Function

Syntax GetXMLSettings

Description Questa funzione restituisce una stringa con il contenuto XML del file di progetto relativo alla lista dei Gruppi utente e degli utenti.

Nota: Le funzioni e le proprietà per la gestione dei Gruppi e degli Utenti sono accessibili solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore.

Parameter	Description
-----------	-------------

None	None
------	------

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim MyUsersAndGroups As UserAndGroupCmdTarget
    GetPasswordFromLevel(0)
    Set MyUsersAndGroups = GetUserAndGroup
    If Not MyUsersAndGroups Is Nothing Then
        MsgBox MyUsersAndGroups.GetXMLSettings,vbOkOnly,""
    End If
    Set MyUsersAndGroups = Nothing
End Sub
```

LogonUser, UserAndGroupCmdTarget Function

Syntax LogonUser(_IpszUserName,_IpszPassword)

Description Questa funzione permette di eseguire il login per l'utente riferito dal nome utente e dalla password passati come parametri.



Utilizzata da una istanza Web Client (sui Simboli e Oggetti) ha lo stesso effetto di eseguire il Login di un utente locale al Server e non di uno lato Web Client.

Parameter	Description
IpszUserName As String	Nome Utente
IpszPassword As String	Password

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        If myObject.LogonUser("Utente1","pwdUtente1") then
            MsgBox("Utente1 has logged in!", vbOkOnly, GetProjectTitle)
        End If
    End If
End Sub
```

ReadRuntimeUsersXML, UserAndGroupCmdTarget Function

Syntax ReadRuntimeUsersXML(vUserName, vPassword)

Description Questa funzione restituisce una stringa con il contenuto XML del file di progetto relativo alla lista degli utenti runtime.
Nota: I parametri relativi al nome utente e password devono corrispondere ad un utente di livello pari o superiore a quello di amministratore. Nel caso in cui questi parametri non vengano impostati, occorre che sia loggato un utente di livello pari o superiore a quello di amministratore (tramite ad esempio la funzione "LogonUser" o il login di un utente da console).

Parameter	Description
vUserName As String	Nome dell'utente di livello pari o superiore a quello di Amministratore (default = "").
vPassword As Strings	Password dell'utente specificato nel parametro "vUserName" (default = "").

Result String

```

Example:
Option Explicit
Sub Main
    Dim sXML As String
    Dim sUserName As String
    Dim sPassword As String
    sUserName = This.GetParameter(0)
    sPassword = This.GetParameter(1)
    If sUserName <> "" Then
        sXML = GetUserAndGroup.ReadRuntimeUsersXML(sUserName,
            sPassword)
    Else
        GetUserAndGroup.LogonUser("Progea", "Progea")
        sXML = GetUserAndGroup.ReadRuntimeUsersXML()
        LogoffActiveUser
    End If
End Sub

```

SaveRuntimeUsersXML, UserAndGroupCmdTarget Function

Syntax SaveRuntimeUsersXML(IpszNewXMLSettings, bCrypt, vUserName, vPassword)

Description Questa funzione salva il contenuto della stringa XML che viene passata come parametro nel file di progetto corrispondente alla lista degli utenti runtime.
Nota: I parametri relativi al nome utente e password devono corrispondere ad un utente di livello pari o superiore a quello di amministratore. Nel caso in cui questi parametri non vengano impostati, occorre che sia loggato un utente di livello pari o superiore a quello di amministratore (tramite ad esempio la funzione "LogonUser" o il login di un utente da console).

Parameter	Description
IpszNewXMLSettings As String	Stringa XML contenente gli utenti runtime: <UserAndGroupSettings> </UserAndGroupSettings>
bCrypt As Boolean	Consente di decidere se salvare il file degli utenti runtime in modo criptato o non criptato. bCrypt = True: Il file degli utenti runtime viene salvato in modo criptato e quindi non sar� leggibile aprendolo con un editor testuale. bCrypt = False: Il file degli utenti runtime viene salvato in modo non criptato e quindi sar� leggibile aprendolo con un editor testuale.

vUserName As String	Nome dell'utente di livello pari o superiore a quello di Amministratore (default = "").
vPassword As Strings	Password dell'utente specificato nel parametro "vUserName" (default = "").

Result Boolean

Example:

Option Explicit

Sub Main

```

Dim sXML As String
Dim sUserName As String
Dim sPassword As String
Dim bCrypt As Boolean
Dim bResult As Boolean

sUserName = This.GetParameter(0)
sPassword = This.GetParameter(1)
bCrypt = CBool(Val(This.GetParameter(2)))

If sUserName <> "" Then
    sXML = GetUserAndGroup.ReadRuntimeUsersXML(sUserName,
sPassword)
    bResult = GetUserAndGroup.SaveRuntimeUsersXML(sXML, bCrypt,
sUserName, sPassword)
Else
    GetUserAndGroup.LogonUser("Progea", "Progea")
    sXML = GetUserAndGroup.ReadRuntimeUsersXML
    bResult = GetUserAndGroup.SaveRuntimeUsersXML(sXML, bCrypt)
    LogoffActiveUser
End If

If bResult = False Then
    MsgBox "Error on writing the runtime xml users file!", vbExclamation +
vbOkOnly, GetProjectTitle
End If
End Sub

```

Properties

AllowResizingForUsersBelowThisLevel, UserAndGroupCmdTarget Property

Syntax AllowResizingForUsersBelowThisLevel = _Integer

Description Tramite questa proprietà è possibile leggere o impostare il livello utente richiesto per le operazioni di sistema di ridimensionamento o riduzione ad icona della finestra di Movicon. Impostando il livello "0" non verrà chiesta nessuna autenticazione per le operazioni di riduzione ad icona o ridimensionamento della finestra di Movicon. Il tentativo di chiudere la finestra invece richiederà sempre un livello di Amministratore (1023), indipendentemente dal valore impostato in questa proprietà. La proprietà potrà essere letta o scritta solo se si è effettuato il login di un utente (tramite ad esempio la funzione "LogonUser" o il login di un utente da console) con un livello pari o superiore a quello di amministratore.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim oUserGroup As UserAndGroupCmdTarget

    Set oUserGroup = GetUserAndGroup()
    oUserGroup.LogonUser("Administrator", "Administrator")
    oUserGroup.AllowResizingForUsersBelowThisLevel = 0
    Set oUserGroup = Nothing
End Sub
```

AllowRuntimeChangesForUsersBelowThisLevel, UserAndGroupCmdTarget Property

Syntax AllowRuntimeChangesForUsersBelowThisLevel = _Integer

Description Tramite questa proprietà è possibile modificare il Livello Utenti oltre il quale non sarà possibile eseguire modifiche in Runtime. Se ad esempio si imposta "Livello Utente 5 (5)", durante il Runtime sarà possibile modificare soltanto gli utenti fino al quinto livello.

I valori possibili per i vari livelli sono:

```
enum_UserLevel0 = livello 0
enum_UserLevel1 = livello 1
enum_UserLevel2 = livello 2
enum_UserLevel3 = livello 3
enum_UserLevel4 = livello 4
enum_UserLevel5 = livello 5
enum_UserLevel6 = livello 6
enum_UserLevel7 = livello 7
enum_UserLevel8 = livello 8
enum_UserLevelAdministrator = livello amministratore
enum_UserLevelDeveloper = livello sviluppatore
```

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        myObject.AllowRuntimeChangesForUsersBelowThisLevel = enum_UserLevel6
    End If
End Sub
```

DefaultPrivAdminAccessLevel, UserAndGroupCmdTarget Property

Syntax DefaultPrivAdminAccessLevel = _Long

Description Questa proprietà imposta o restituisce la maschera di Livello Predefinito di Accesso da assegnare agli Utenti del Sistema Operativo appartenenti al gruppo Administrators.



Questa proprietà non è supportata in Windows CE. (Se impostata restituisce sempre zero)

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        Debug.Print myObject.DefaultPrivAdminAccessLevel
    End If
End Sub
```

DefaultPrivAdminLevel, UserAndGroupCmdTarget Property

Syntax DefaultPrivAdminLevel = _Integer

Description Questa proprietà imposta o restituisce il Livello Predefinito di Password da assegnare agli Utenti del Sistema Operativo appartenenti al gruppo Administrators.

I valori possibili per i livelli sono:

```
enum_UserLevel0 = livello 0
enum_UserLevel1 = livello 1
enum_UserLevel2 = livello 2
enum_UserLevel3 = livello 3
enum_UserLevel4 = livello 4
enum_UserLevel5 = livello 5
enum_UserLevel6 = livello 6
enum_UserLevel7 = livello 7
enum_UserLevel8 = livello 8
enum_UserLevelAdministrator= livello amministratore
enum_UserLevelDeveloper = livello sviluppatore
```



Questa proprietà non è supportata in Windows CE. (Se impostata restituisce sempre zero)

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        myObject.DefaultPrivAdminLevel = enum_UserLevelAdministrator
    End If
End Sub
```

DefaultPrivGuestAccessLevel, UserAndGroupCmdTarget Property

Syntax DefaultPrivGuestAccessLevel = _Long

Description Questa proprietà imposta o restituisce la maschera di Livello Predefinito di Accesso da assegnare agli Utenti del Sistema Operativo appartenenti al gruppo Guests.



Questa proprietà non è supportata in Windows CE. (Se impostata restituisce sempre zero)

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        Debug.Print myObject.DefaultPrivGuestAccessLevel
    End If
End Sub
```

DefaultPrivGuestLevel, UserAndGroupCmdTarget Property

Syntax DefaultPrivGuestLevel = _Integer

Description Questa proprietà imposta o restituisce il Livello Predefinito di Password da assegnare agli Utenti del Sistema Operativo appartenenti al gruppo Guests.

I valori possibili per i livelli sono:

```
enum_UserLevel0 = livello 0
enum_UserLevel1 = livello 1
enum_UserLevel2 = livello 2
enum_UserLevel3 = livello 3
enum_UserLevel4 = livello 4
```

enum_UserLevel5 = livello 5
enum_UserLevel6 = livello 6
enum_UserLevel7 = livello 7
enum_UserLevel8 = livello 8
enum_UserLevelAdministrator = livello amministratore
enum_UserLevelDeveloper = livello sviluppatore



Questa proprietà non è supportata in Windows CE. (Se impostata restituisce sempre zero)

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        myObject.DefaultPrivGuestLevel = enum_enum_UserLevel0
    End If
End Sub
```

DefaultPrivUserAccessLevel, UserAndGroupCmdTarget Property

Syntax DefaultPrivUserAccessLevel = _Long

Description Questa proprietà imposta o restituisce la maschera di Livello Predefinito di Accesso da assegnare agli Utenti del Sistema Operativo appartenenti al gruppo Users.



Questa proprietà non è supportata in Windows CE. (Se impostata restituisce sempre zero)

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        Debug.Print myObject.DefaultPrivUserAccessLevel
    End If
End Sub
```

DefaultPrivUserLevel, UserAndGroupCmdTarget Property

Syntax DefaultPrivUserLevel = _Integer

Description Questa proprietà imposta o restituisce il Livello Predefinito di Password da assegnare agli Utenti del Sistema Operativo appartenenti al gruppo Users.

I valori possibili per i livelli sono:

enum_UserLevel0 = livello 0
enum_UserLevel1 = livello 1
enum_UserLevel2 = livello 2
enum_UserLevel3 = livello 3
enum_UserLevel4 = livello 4
enum_UserLevel5 = livello 5
enum_UserLevel6 = livello 6
enum_UserLevel7 = livello 7
enum_UserLevel8 = livello 8
enum_UserLevelAdministrator = livello amministratore
enum_UserLevelDeveloper = livello sviluppatore



Questa proprietà non è supportata in Windows CE. (Se impostata restituisce sempre 'false')

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        myObject.DefaultPrivUserLevel = enum_enum_UserLevel5
    End If
End Sub
```

EnableAutoLogoff, UserAndGroupCmdTarget Property

Syntax EnableAutoLogoff = _Boolean

Description Questa proprietà imposta o restituisce la configurazione di AutoLogoff per Gli Utenti e i Gruppi Utenti. Se posta al valore booleano True, verrà eseguito il Logoff automaticamente da Movicon dopo il tempo impostato nella proprietà "Logoff Automatico dopo (sec)". Il conteggio del tempo inizierà dal momento in cui l'utente non eseguirà più alcuna operazione sulla tastiera o con il mouse.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        MsgBox("EnableAutoLogoff is: " &
            CBool(myObject.EnableAutoLogoff), vbOkOnly, GetProjectTitle)
    End If
End Sub
```

EnableNTUserLogin, UserAndGroupCmdTarget Property

Syntax EnableNTUserLogin = _Boolean

Description Questa proprietà imposta o restituisce la configurazione di "Abilitazione Utenti NT"



Questa proprietà non è supportata in Windows CE. (Se impostata restituisce sempre 'false')

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        MsgBox("EnableNTUserLogin is: " &
            CBool(myObject.EnableNTUserLogin), vbOkOnly, GetProjectTitle)
    End If
End Sub
```

EnableRuntimeUsers, UserAndGroupCmdTarget Property

Syntax EnableRuntimeUsers = _Boolean

Description Questa proprietà, se impostata al valore booleano True, permette di attivare la gestione degli Utenti di Runtime, ovvero permette di poter aggiungere, modificare e cancellare nuovi utenti durante il Runtime del progetto.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        MsgBox("EnableRuntimeUsers is: " &
            CBool(myObject.EnableRuntimeUsers),vbOkOnly,GetProjectTitle)
    End If
End Sub
```

EnableRuntimeUsersSecurity, UserAndGroupCmdTarget Property

Syntax EnableRuntimeUsersSecurity = _Boolean

Description Questa proprietà permette leggere o di scrivere un valore booleano per verificare o impostare lo stato della gestione della sicurezza degli Utenti di Runtime del progetto attraverso l'inserimento automatico del GUID o del Codice d'Accesso. Se il valore è impostato a True la gestione della sicurezza è attiva.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        MsgBox("EnableRuntimeUsersSecurity is: " &
            CStr(myObject.EnableRuntimeUsersSecurity),vbOkOnly,GetProjectTitle)
    End If
End Sub
```

MinimumPasswordLength, UserAndGroupCmdTarget Property

Syntax MinimumPasswordLength = _Long

Description Questa proprietà imposta o restituisce il numero minimo di caratteri di cui dovrà essere composta la Password degli Utenti definiti nel progetto.



La lunghezza minima per la Password Utenti definita per default da Movicon è di sei caratteri. Per questioni di normative e di sicurezza si consiglia di non diminuire questo valore.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        myObject.MinimumPasswordLength = 60
    End If
End Sub
```

MinimumUserLength, UserAndGroupCmdTarget Property

Syntax MinimumUserLength = _Long

Description Questa proprietà imposta o restituisce il numero minimo di caratteri di cui dovrà essere composto il Nome degli Utenti definiti nel progetto.



La lunghezza minima per il Nome Utente definita per default da Movicon è di quattro caratteri. Per questioni di normative e di sicurezza si consiglia di non diminuire questo valore.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        myObject.MinimumUserLength = 6
    End If
End Sub
```

RuntimeUserAccessCode, UserAndGroupCmdTarget Property

Syntax RuntimeUserAccessCode = _String

Description Questa proprietà permette di impostare il valore alfanumerico del Codice di Accesso utilizzato per bloccare l'accesso al file RTUsers degli Utenti di Runtime da parte di progetti che non hanno lo stesso codice accesso. La proprietà è modificabile solo dopo il login al progetto (anche tramite la LogonUser) di un utente Amministratore del progetto stesso.

Se impostato, il Codice di Accesso viene utilizzato al posto del GUID del progetto ed inserito nel file .rtusers.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
Dim myObject As UserAndGroupCmdTarget
Set myObject = GetUserAndGroup
If Not myObject Is Nothing Then
MsgBox("RuntimeUserAccessCode is: " & myObject.RuntimeUserAccessCode,
vbOkOnly,GetProjectTitle)
End If
End Sub
```

SecsAutoLogoffTimeout, UserAndGroupCmdTarget Property

Syntax SecsAutoLogoffTimeout = _Long

Description Questa proprietà imposta o restituisce il tempo in secondi di autologoff dopo il quale verrà eseguito il Logoff automatico da Movicon se la proprietà EnableAutoLogoff è posta al valore booleano True.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Public Sub Click()
Dim myObject As UserAndGroupCmdTarget
Set myObject = GetUserAndGroup
If Not myObject Is Nothing Then
myObject.SecsAutoLogoffTimeout= 35
End If
End Sub
```

Functions

GetDaysBeforePasswExpires, UserCmdTarget Function

Syntax GetDaysBeforePasswExpires()

Description Questa funzione restituisce il numero di giorni che sono trascorsi da quando l'utente è stato creato o da quando è stata rinnovata la password scaduta. Una volta scaduta, alla successiva autenticazione dell'Utente, verrà chiesto di inserire una nuova password. Il calcolo viene effettuato in base al valore impostato nella proprietà Durata Password (Giorni) in fase di progetto o tramite codice basic con la proprietà ExpiringDaysPassword. Se il valore di questa proprietà viene lasciato a zero, verrà ereditato il valore impostato nella proprietà "Durata Password (Giorni)" del Gruppo di appartenenza.

La funzione calcola i giorni in base all'attributo "TimePassword" nel file .movprj per gli utenti di Sviluppo e nel file .rtusers per gli utenti di Runtime. Esso viene popolato la prima volta con la data e ora di creazione dell'utente, es:

TimePassword="2009-03-13T10:19:42Z"

La funzione è accessibile solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel, UserAndGroupCmdTarget Property. In caso contrario la funzione restituisce un valore pari a 0.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetUserObject("Group1","Ut1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.GetDaysBeforePasswExpires
            Set obj2 = Nothing
        End If
    End If
End Sub
```

GetGroupObject, UserCmdTarget Function

Syntax GetGroupObject()

Description Questa funzione restituisce un oggetto di tipo UserGroupCmdTarget relativo al gruppo di appartenenza dell'utente. Tramite questa funzione sarà possibile accedere alle proprietà e ai metodi relativi al gruppo di appartenenza.

L'oggetto restituito è diverso da Nothing solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore o uguale al valore definito dalla proprietà 'Max. Livello Utente Editabile' (modificabile anche dalla proprietà 'AllowResizingForUsersBelowThisLevel' dell'interfaccia 'UserAndGroupCmdTarget'). Questa funzione infatti ha una protezione che impedisce l'accesso ai livelli inferiori.

Parameter	Description
None	None

Result Object
Viene restituito un oggetto di tipo UserGroupCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

Option Explicit

```
Public Sub Click()  
Dim oUserAndGroup As UserAndGroupCmdTarget  
Dim oUser As UserCmdTarget  
Dim oUserGroup As UserGroupCmdTarget  
    GetPasswordFromLevel(enum_UserLevelAdministrator)  
    Set oUserAndGroup = GetUserAndGroup  
    oUserAndGroup.AllowRuntimeChangesForUsersBelowThisLevel = enum_UserLevelAdministrator ' Level 1023  
  
    Set oUser = oUserAndGroup.GetActiveUserObject  
    Set oUserGroup = oUser.GetGroupObject  
    If Not oUserGroup Is Nothing Then  
        Debug.Print "Group Name:" & oUserGroup.Name & " - Group Description: " & oUserGroup.Description  
    Else  
        Debug.Print "Group Object is Nothing!"  
    End If  
End Sub
```

GetLastTimeUserAction, UserCmdTarget Function

Syntax GetLastTimeUserAction()

Description Questa funzione restituisce il tempo trascorso dall'ultima interazione dell'utente col sistema.
La funzione è accessibile solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel.
In caso contrario la funzione restituisce un valore pari a 0.00.00.

Parameter	Description
None	None

Result Date

Example:

```
Option Explicit
Sub Main()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetUserObject("Group1","Ut1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.GetLastTimeUserAction
            Set obj2 = Nothing
        End If
    End If
End Sub
```

GetListAccessVariable, UserCmdTarget Function

Syntax GetListAccessVariable()

Description Questa funzione restituisce una stringa contenente la lista delle variabili accessibili dall'utente in base al proprio Livello di Accesso. Per rinfrescare la lista delle variabili accessibili, nel caso si preveda una variazione della configurazione delle stesse, fare riferimento alla funzione ResetListAccessVariables.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetActiveUserObject
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            obj2.ResetListAccessVariables
            Debug.Print obj2.GetListAccessVariable
            Set obj2 = Nothing
        End If
    End If
End Sub
```

GetProp, UserCmdTarget Function

Syntax GetProp(_lpszPropID)

Description Questa funzione permette di recuperare il valore di una proprietà precedentemente impostata per l'utente selezionato. In questo modo si possono gestire delle

proprietà, ad esempio da un web client, che vengono mantenute per tutta la durata della sessione in cui opera quell'utente. Queste proprietà non sono retentive quindi verranno perse quando l'utente esegue il logoff.

Parameter	Description
lpszPropID	Nome identificativo della proprietà da recuperare.

Result String

Example 1:

'Code executable also on WebClient

Option Explicit

Public Sub Click()

 Dim objUser As UserCmdTarget

 Set objUser = GetSynopticObject.GetActiveUserObject

 If Not objUser Is Nothing Then

 objUser.**SetProp**("Prop1","Name")

 objUser.**SetProp**("Prop2","Description")

 Debug.Print "Prop1 = " & objUser.**GetProp**("Prop1") & " - " & "Prop2 = "

 & objUser.**GetProp**("Prop2")

 Set objUser = Nothing

 End If

End Sub

Example 2:

'Code not executable on WebClient

Option Explicit

Public Sub Click()

 Dim objUserAndGroup As UserAndGroupCmdTarget

 Dim objUser As UserCmdTarget

 Set objUserAndGroup = GetUserAndGroup

 If Not objUserAndGroup Is Nothing Then

 Set objUser = objUserAndGroup.GetActiveUserObject

 If Not objUser Is Nothing Then

 objUser.**SetProp**("Prop1","Name")

 objUser.**SetProp**("Prop2","Description")

 Debug.Print "Prop1 = " & objUser.**GetProp**("Prop1") & " - " &

"Prop2 = " & objUser.**GetProp**("Prop2")

 Set objUser = Nothing

 End If

 Set objUserAndGroup = Nothing

 End If

End Sub

GetXMLSettings, UserCmdTarget Function

Syntax GetXMLSettings()

Description Questa funzione restituisce una stringa con il contenuto XML del file di progetto relativo all'utente.

La funzione è accessibile solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel.

In caso contrario la funzione restituisce una stringa vuota.

Parameter	Description
-----------	-------------

None	None
------	------

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetUserObject("Group1","Ut1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            MsgBox "GetXMLSettings Is -> " & obj2.GetXMLSettings,
                vbOkOnly, GetProjectTitle
            Set obj2 = Nothing
        End If
    End If
End Sub
```

IsRemote, UserCmdTarget Function

Syntax IsRemote()

Description Questa funzione restituisce il valore booleano True se l'utente ha eseguito il login da remoto.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetUserObject("Group1","Ut1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.IsRemote
            Set obj2 = Nothing
        End If
    End If
End Sub
```

ResetListAccessVariables, UserCmdTarget Function

Syntax ResetListAccessVariables()

Description Questa funzione permette di rinfrescare la lista delle variabili accessibili in base al livello di accesso dell'utente che ha effettuato il login. In caso di errore la funzione restituisce il valore booleano True.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetActiveUserObject
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            obj2.ResetListAccessVariables
            Debug.Print obj2.GetListAccessVariable
            Set obj2 = Nothing
        End If
    End If
End Sub
```

SetProp, UserCmdTarget Function

Syntax SetProp(_IpszPropID, _IpszValue)

Description Questa funzione permette di impostare il valore di una proprietà per l'utente selezionato. In questo modo si possono gestire delle proprietà, ad esempio da un web client, che vengono mantenute per tutta la durata della sessione in cui opera quell'utente. Queste proprietà non sono retentive quindi verranno perse quando l'utente esegue il logoff.

Parameter	Description
IpszPropID as string	Nome identificativo della proprietà da impostare.
IpszValue as string	Valore da impostare nella proprietà.

Result Boolean

Example 1:

```
'Code executable also on WebClient
Option Explicit
Public Sub Click()
    Dim objUser As UserCmdTarget

    Set objUser = GetSynopticObject.GetActiveUserObject
    If Not objUser Is Nothing Then
        objUser.SetProp("Prop1","Name")
        objUser.SetProp("Prop2","Description")
        Debug.Print "Prop1 = " & objUser.GetProp("Prop1") & " - " & "Prop2 = "
        & objUser.GetProp("Prop2")
        Set objUser = Nothing
    End If
End Sub
```

Example 2:

```
'Code not executable on WebClient
Option Explicit
Public Sub Click()
    Dim objUserAndGroup As UserAndGroupCmdTarget
    Dim objUser As UserCmdTarget

    Set objUserAndGroup = GetUserAndGroup
    If Not objUserAndGroup Is Nothing Then
        Set objUser = objUserAndGroup.GetActiveUserObject
        If Not objUser Is Nothing Then
            objUser.SetProp("Prop1","Name")
            objUser.SetProp("Prop2","Description")
            Debug.Print "Prop1 = " & objUser.GetProp("Prop1") & " - " &
"Prop2 = " & objUser.GetProp("Prop2")
            Set objUser = Nothing
        End If
        Set objUserAndGroup = Nothing
    End If
End Sub
```

Properties

AccessLevel, UserCmdTarget Property

Syntax AccessLevel = _Long

Description Questa proprietà permette di definire la maschera di Livello di Accesso da associare all'utente.

La proprietà è accessibile solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel. In caso contrario il valore restituito è pari a 1.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetUserObject("Group1","Ut1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.AccessLevel
            Set obj2 = Nothing
        End If
    End If
End Sub
```

AccountDisabled, UserCmdTarget Property

Syntax AccountDisabled = _Boolean

Description Questa proprietà abilita o disabilita l'Utente. La proprietà è accessibile in scrittura solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetActiveUserObject
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.AccountDisabled
            Set obj2 = Nothing
        End If
    End If
End Sub
```

CannotChangePassword, UserCmdTarget Property

Syntax CannotChangePassword= _Boolean

Description Questa proprietà consente di leggere o impostare l'opzione a livello di utente che blocca la possibilità di modificare la password a runtime durante la fase di login. Per impostare la proprietà da Basic Script occorre che a runtime sia loggato un utente di livello pari o superiore a quello di amministratore tramite, ad esempio, la funzione 'LogonUser' o il login di un utente eseguito da console.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim objUserAndGroup As UserAndGroupCmdTarget
    Dim objUserGroup As UserGroupCmdTarget
    Dim objUser As UserCmdTarget
```

```

Set objUserAndGroup = GetUserAndGroup
If Not objUserAndGroup Is Nothing Then
    objUserAndGroup.LogonUser("Admin","Admin")
    Set objUserGroup = objUserAndGroup.GetGroupObject("Users")
    If Not objUserGroup Is Nothing Then
        Set objUser = objUserGroup.GetUserObject("User01")
        objUser.CannotChangePassword = Not
        objUser.CannotChangePassword
    End If
    LogoffActiveUser
End If
End Sub

```

CommandListLogoff, UserCmdTarget Property

Syntax CommandListLogoff

Description Questa proprietà restituisce una stringa contenente il codice XML della Lista Comandi di LogOff associata all'utente referenziato.

Parameter	Description
None	None

Result String

Example:

```

Sub Main()
    Dim objUserAndGroup As UserAndGroupCmdTarget
    Dim objUser As UserCmdTarget
    Set objUserAndGroup = GetUserAndGroup
    If Not objUserAndGroup Is Nothing Then
        objUserAndGroup.LogonUser("Administrator", "Administrator")
        Set objUser = objUserAndGroup.GetUserObject("Users","User01")
        If Not objUser Is Nothing Then
            MsgBox "CommandListLogoff" & objUser.CommandListLogoff,
                vbInformation, GetProjectTitle
            Set objUser = Nothing
        End If
        Set objUserAndGroup = Nothing
    End If
End Sub

```

CommandListLogon, UserCmdTarget Property

Syntax CommandListLogon

Description Questa proprietà restituisce una stringa contenente il codice XML della Lista Comandi di LogOn associata all'utente referenziato.

Parameter	Description

None	None
------	------

Result String

Example:

```
Sub Main()
    Dim objUserAndGroup As UserAndGroupCmdTarget
    Dim objUser As UserCmdTarget
    Set objUserAndGroup = GetUserAndGroup
    If Not objUserAndGroup Is Nothing Then
        objUserAndGroup.LogonUser("Administrator", "Administrator")
        Set objUser = objUserAndGroup.GetUserObject("Users","User01")
        If Not objUser Is Nothing Then
            MsgBox "CommandListLogon = " & objUser.CommandListLogon,
                vbInformation, GetProjectTitle
            Set objUser = Nothing
        End If
        Set objUserAndGroup = Nothing
    End If
End Sub
```

Description, UserCmdTarget Property

Syntax Description = _String

Description Questa proprietà imposta o restituisce un testo descrittivo relativo al profilo dell'utente. La descrizione dell'utente verrà utilizzata dal sistema per identificare l'Utente attivo, e quindi registrarlo ove previsto, secondo le disposizioni richieste dalla "Firma Elettronica".
La proprietà è accessibile in scrittura solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetActiveUserObject
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.Description
            Set obj2 = Nothing
        End If
    End If
End Sub
```

Email, UserCmdTarget Property

Syntax Description = _String

Description Questa proprietà imposta o restituisce l'indirizzo E-mail dell'Utente per l'invio di e-mail.
La proprietà è accessibile (lettura/scrittura) solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel.
In caso contrario la proprietà restituisce una stringa vuota.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetUserObject("Group1","Ut1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.Email
            Set obj2 = Nothing
        End If
    End If
End Sub
```

EnableAutoLoggoff, UserCmdTarget Property

Syntax EnableAutoLoggoff = _Boolean

Description Questa proprietà imposta o restituisce la configurazione di AutoLoggoff per l'Utente referenziato. Se posta al valore booleano True, verrà eseguito il Logoff automaticamente da Movicon dopo il tempo impostato nella proprietà "Logoff Automatico dopo (sec)". Il conteggio del tempo inizierà dal momento in cui l'utente non eseguirà più alcuna operazione sulla tastiera o con il mouse.



Se la proprietà "Abilita Auto Loggoff" viene disabilitata, un utente che esegue il Logon rimarrà attivo fino a che non verrà eseguito un comando di disattivazione (stabilito dal programmatore nel progetto) o l'utente non sarà sostituito da un utente diverso.

se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetActiveUserObject
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.EnableAutoLoggoff
            Set obj2 = Nothing
        End If
    End If
End Sub
```

ExpiringDaysPassword, UserCmdTarget Property

Syntax ExpiringDaysPassword = _Long

Description

Questa proprietà permette di definire il numero di giorni dopo i quali la Password dell'Utente scade e non sarà più utilizzabile. Una volta scaduta, alla successiva autenticazione dell'Utente, verrà chiesto di inserire una nuova password. L'inserimento di un valore in questo campo ha significato soltanto se la proprietà dell'Utente "Modifica Password Obbligatoria" è stata abilitata. Se il valore di questa proprietà viene lasciato a zero, verrà ereditato il valore impostato nella proprietà "Durata Password (Giorni)" del Gruppo di appartenenza. La proprietà è accessibile (lettura/scrittura) solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel. In caso contrario il valore restituito è pari a 0.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetUserObject("Group1","Ut1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.ExpiringDaysPassword
            Set obj2 = Nothing
        End If
    End If
End Sub
```

FaxAreaCode, UserCmdTarget Property

Syntax FaxAreaCode = _String

Description Questa proprietà imposta o restituisce il prefisso nazionale relativo al numero per i Fax dell'Utente.
La proprietà è accessibile (lettura/scrittura) solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel.
In caso contrario la proprietà restituisce una stringa vuota.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetUserObject("Group1", "Ut1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.FaxAreaCode
            Set obj2 = Nothing
        End If
    End If
End Sub
```

FaxCountryCode, UserCmdTarget Property

Syntax FaxCountryCode = _String

Description Questa proprietà imposta o restituisce il prefisso internazionale relativo al numero per i Fax dell'Utente.
La proprietà è accessibile (lettura/scrittura) solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel.
In caso contrario la proprietà restituisce una stringa vuota.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
```

```

Dim obj2 As UserCmdTarget
Set obj1 = GetUserAndGroup
If Not obj1 Is Nothing Then
    Set obj2 = obj1.GetUserObject("Group1","Ut1")
    Set obj1 = Nothing
    If Not obj2 Is Nothing Then
        Debug.Print obj2.FaxCountryCode
        Set obj2 = Nothing
    End If
End If
End Sub

```

FaxPhoneNumber, UserCmdTarget Property

Syntax FaxPhoneNumber = _String

Description Questa proprietà imposta o restituisce il numero per i Fax dell'Utente. La proprietà è accessibile (lettura/scrittura) solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel. In caso contrario la proprietà restituisce una stringa vuota.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetUserObject("Group1","Ut1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.FaxPhoneNumber
            Set obj2 = Nothing
        End If
    End If
End Sub

```

Language, UserCmdTarget Property

Syntax Language = _String

Description Questa proprietà permette di leggere o impostare la lingua di default per l'Utente. Inserendo una lingua in questa proprietà quando l'Utente esegue il Logon verrà automaticamente attivata quella lingua. Questo consente quindi di avere un sistema multilingua e di poter associare ad ogni utente la lingua di default desiderata.

Parameter	Description
-----------	-------------

None	None
------	------

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Dim myUser As UserCmdTarget
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        Set myUser = myObject.GetUserObject("Group1","User1")
        If Not myUser Is Nothing Then
            MsgBox("User's Language = " & myUser.Language, vbOkOnly, GetProjectTitle)
        End If
    End If
End Sub
```

Level, UserCmdTarget Property

Syntax Level = _Integer

Description Questa proprietà imposta o restituisce il Livello Utente assegnato all'utente.

I valori possibili per i livelli sono:

```
enum_UserLevel0 = livello 0
enum_UserLevel1 = livello 1
enum_UserLevel2 = livello 2
enum_UserLevel3 = livello 3
enum_UserLevel4 = livello 4
enum_UserLevel5 = livello 5
enum_UserLevel6 = livello 6
enum_UserLevel7 = livello 7
enum_UserLevel8 = livello 8
enum_UserLevelAdministrator = livello amministratore
enum_UserLevelDeveloper = livello sviluppatore
```

La proprietà è accessibile solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel. In caso contrario il valore restituito è pari a -1.

Parameter	Description
None	None

Result Integer

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetUserObject("Group1","Ut1")
        Set obj1 = Nothing
    End If
End Sub
```

```

        If Not obj2 Is Nothing Then
            Debug.Print obj2.Level
            Set obj2 = Nothing
        End If
    End If
End Sub

```

Locked, UserCmdTarget Property

Syntax Locked = _Boolean

Description Questa proprietà permette di bloccare l'accesso ad un utente specifico. La proprietà è accessibile solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel, UserAndGroupCmdTarget Property. In caso contrario il valore restituito è pari al valore booleano False.

Parameter	Description
None	None

Result Boolean

Example:

```

Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetUserObject("Group1","Ut1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.Locked
            Set obj2 = Nothing
        End If
    End If
End Sub

```

LogoffScript, UserCmdTarget Property

Syntax LogoffScript = _String

Description Questa proprietà imposta o restituisce il nome dello script da eseguire al LogOff dell'utente. La proprietà è accessibile (lettura/scrittura) solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel. In caso contrario la proprietà restituisce una stringa vuota.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
Dim myObject As UserAndGroupCmdTarget
Dim myUser As UserCmdTarget
Set myObject = GetUserAndGroup
If Not myObject Is Nothing Then
Set myUser = myObject.GetUserObject("Group1","Ut1")
If Not myUser Is Nothing Then
MsgBox("LogoffScript is: " & myUser.LogoffScript, vbOkOnly, GetProjectTitle)
End If
End If
End Sub
```

LogonScript, UserCmdTarget Property

Syntax LogonScript = _String

Description Questa proprietà imposta o restituisce il nome dello script da eseguire al LogOn dell'utente.
La proprietà è accessibile (lettura/scrittura) solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel.
In caso contrario la proprietà restituisce una stringa vuota.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
Dim myObject As UserAndGroupCmdTarget
Dim myUser As UserCmdTarget
Set myObject = GetUserAndGroup
If Not myObject Is Nothing Then
Set myUser = myObject.GetUserObject("Group1","Ut1")
If Not myUser Is Nothing Then
MsgBox("LogonScript is: " & myUser.LogonScript, vbOkOnly, GetProjectTitle)
End If
End If
End Sub
```

MobileAreaCode, UserCmdTarget Property

Syntax MobileAreaCode = _String

Description Questa proprietà imposta o restituisce il prefisso nazionale relativo al numero di telefono mobile dell'Utente.
 La proprietà è accessibile (lettura/scrittura) solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel.
 In caso contrario la proprietà restituisce una stringa vuota.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
Dim myObject As UserAndGroupCmdTarget
Dim myUser As UserCmdTarget
Set myObject = GetUserAndGroup
If Not myObject Is Nothing Then
Set myUser = myObject.GetUserObject("Group1","Ut1")
If Not myUser Is Nothing Then
MsgBox("MobileAreaCode is: " & myUser.MobileAreaCode, vbOkOnly, GetProjectTitle)
End If
End If
End Sub
```

MobileCountryCode, UserCmdTarget Property

Syntax MobileCountryCode = _String

Description Questa proprietà imposta o restituisce il prefisso internazionale relativo al numero di telefono mobile dell'Utente.
 La proprietà è accessibile (lettura/scrittura) solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel.
 In caso contrario la proprietà restituisce una stringa vuota.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
Dim myObject As UserAndGroupCmdTarget
Dim myUser As UserCmdTarget
Set myObject = GetUserAndGroup
If Not myObject Is Nothing Then
Set myUser = myObject.GetUserObject("Group1","Ut1")
If Not myUser Is Nothing Then
MsgBox("MobileCountryCode is: " & myUser.MobileCountryCode, vbOkOnly, GetProjectTitle)
End If
End If
End Sub
```

End If
End If
End Sub

MobilePhoneNumber, UserCmdTarget Property

Syntax MobilePhoneNumber = _String

Description Questa proprietà imposta o restituisce il numero di telefono mobile dell'Utente. La proprietà è accessibile (lettura/scrittura) solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel. In caso contrario la proprietà restituisce una stringa vuota.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
Dim myObject As UserAndGroupCmdTarget
Dim myUser As UserCmdTarget
Set myObject = GetUserAndGroup
If Not myObject Is Nothing Then
Set myUser = myObject.GetUserObject("Group1","Ut1")
If Not myUser Is Nothing Then
MsgBox("MobilePhoneNumber is: " & myUser.MobilePhoneNumber, vbOkOnly,
GetProjectTitle)
End If
End If
End Sub
```

MustChangedPasswordLogon, UserCmdTarget Property

Syntax MustChangedPasswordLogon = _Boolean

Description Questa proprietà permette di bloccare l'accesso ad un utente specifico. La proprietà è accessibile solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel, UserAndGroupCmdTarget Property. In caso contrario il valore restituito è pari al valore booleano False.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetUserObject("Group1","Ut1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.MustChangedPasswordLogon
            Set obj2 = Nothing
        End If
    End If
End Sub
```

Name, UserCmdTarget Property

Syntax Name = _String

Description Questa proprietà imposta o restituisce il nome dell'utente.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim myObject As UserAndGroupCmdTarget
    Dim myUser As UserCmdTarget
    Set myObject = GetUserAndGroup
    If Not myObject Is Nothing Then
        Set myUser = myObject.GetUserObject("Group1","Ut1")
        If Not myUser Is Nothing Then
            MsgBox("Name is: " & myUser.Name, vbOkOnly, GetProjectTitle)
        End If
    End If
End Sub
```

OnLine, UserCmdTarget Property

Syntax OnLine = _Boolean

Description Questa proprietà restituisce il valore booleano True se l'utente referenziato è OnLine (ha effettuato il Login).
La proprietà è accessibile in scrittura solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà

AllowRuntimeChangesForUsersBelowThisLevel, UserAndGroupCmdTarget Property.
Impostare la proprietà a True significa che verrà effettuato automaticamente il Login per l'utente referenziato.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetUserObject("Group1","Ut1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.OnLine
            Set obj2 = Nothing
        End If
    End If
End Sub
```

Password, UserCmdTarget Property

Syntax Password = _String

Description Questa proprietà consente di impostare la password dell'Utente.
La proprietà è accessibile (in scrittura) solo se si è effettuato il login di un utente (tramite ad esempio la funzione "LogonUser" o il login di un utente da console) con un livello pari o superiore a quello di amministratore (1023) e se l'utente referenziato ha un Livello Utente uguale o inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel.
In lettura la proprietà restituisce sempre una stringa vuota. Per poter visualizzare le proprietà dell'utente e quindi anche la password ad esso associata si può utilizzare la funzione "GetXMLSettings".

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim objUserAndGroup As UserAndGroupCmdTarget
    Dim objUser As UserCmdTarget
    Dim sNewPassword As String

    sNewPassword = InputBox("Insert New Password:",GetProjectTitle,"")
    Set objUserAndGroup = GetUserAndGroup
    objUserAndGroup.LogonUser("Administrator","Administrator")
    objUserAndGroup.AllowRuntimeChangesForUsersBelowThisLevel = 8
    If Not objUserAndGroup Is Nothing Then
        Set objUser = objUserAndGroup.GetUserObject("Group1","User01")
        If Not objUser Is Nothing Then
```

```

        objUser.Password = sNewPassword
    End If
    Set objUserAndGroup = Nothing
End If
LogoffActiveUser
End Sub

```

SecsAutoLoggoffTimeout, UserCmdTarget Property

Syntax SecsAutoLoggoffTimeout = _Long

Description Questa proprietà imposta o restituisce il tempo dopo il quale Movicon eseguirà l'Auto Logoff dell'eventuale Utente attivo. Naturalmente questa impostazione ha significato soltanto se la proprietà "Abilita Auto Logoff" è stata attivata. Il conteggio del tempo inizierà dal momento in cui l'utente non eseguirà più alcuna operazione sulla tastiera o con il mouse.

La proprietà è accessibile solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel, UserAndGroupCmdTarget Property. In caso contrario il valore restituito è pari a 0.

Parameter	Description
None	None

Result Long

Example:

```

Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetUserObject("Group1","Ut1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.SecsAutoLoggoffTimeout
            Set obj2 = Nothing
        End If
    End If
End Sub

```

VoiceAreaCode, UserCmdTarget Property

Syntax VoiceAreaCode = _String

Description Questa proprietà imposta o restituisce il prefisso internazionale relativo al numero per i messaggi vocali dell'Utente. La proprietà è accessibile (lettura/scrittura) solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel.

In caso contrario la proprietà restituisce una stringa vuota.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
Dim myObject As UserAndGroupCmdTarget
Dim myUser As UserCmdTarget
Set myObject = GetUserAndGroup
If Not myObject Is Nothing Then
Set myUser = myObject.GetUserObject("Group1","Ut1")
If Not myUser Is Nothing Then
MsgBox("VoiceAreaCode is: " & myUser.VoiceAreaCode, vbOkOnly, GetProjectTitle)
End If
End If
End Sub
```

VoiceCountryCode, UserCmdTarget Property

Syntax VoiceCountryCode = _String

Description Questa proprietà imposta o restituisce il prefisso internazionale relativo al numero per i messaggi vocali dell'Utente.
La proprietà è accessibile (lettura/scrittura) solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel.
In caso contrario la proprietà restituisce una stringa vuota.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
Dim myObject As UserAndGroupCmdTarget
Dim myUser As UserCmdTarget
Set myObject = GetUserAndGroup
If Not myObject Is Nothing Then
Set myUser = myObject.GetUserObject("Group1","Ut1")
If Not myUser Is Nothing Then
MsgBox("VoiceCountryCode is: " & myUser.VoiceCountryCode , vbOkOnly,
GetProjectTitle)
End If
End If
End Sub
```

VoicePhoneNumber, UserCmdTarget Property

Syntax VoicePhoneNumber = _String

Description Questa proprietà imposta o restituisce il numero per i messaggi vocali dell'Utente. La proprietà è accessibile (lettura/scrittura) solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore e se l'utente referenziato ha un Livello Utente inferiore al valore definito dalla proprietà AllowRuntimeChangesForUsersBelowThisLevel. In caso contrario la proprietà restituisce una stringa vuota.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
Dim myObject As UserAndGroupCmdTarget
Dim myUser As UserCmdTarget
Set myObject = GetUserAndGroup
If Not myObject Is Nothing Then
Set myUser = myObject.GetUserObject("Group1","Ut1")
If Not myUser Is Nothing Then
MsgBox("VoicePhoneNumber is: " & myUser.VoicePhoneNumber , vbOkOnly, GetProjectTitle)
End If
End If
End Sub
```

WebClientAutoLogoffSecs, UserCmdTarget Property

Syntax WebClientAutoLogoffSecs = _Long

Description Questa proprietà consente di leggere o impostare il valore del tempo di auto logoff dell'utente Web Client collegato all'applicazione Server. Il valore impostato sulla proprietà 'Web Client Autologoff (sec.)' dell'utente viene presa in considerazione se è non nulla e la Gestione Password è attiva. Per impostare la proprietà da Basic Script occorre che a runtime sia loggato un utente di livello pari o superiore a quello di amministratore tramite, ad esempio, la funzione 'LogonUser' o il login di un utente eseguito da console.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Public Sub Click()
Dim objUserAndGroup As UserAndGroupCmdTarget
Dim objUserGroup As UserGroupCmdTarget
Dim objUser As UserCmdTarget
```

```

Set objUserAndGroup = GetUserAndGroup
If Not objUserAndGroup Is Nothing Then
    objUserAndGroup.LogonUser("Admin","Admin")
    Set objUserGroup = objUserAndGroup.GetGroupObject("Users")
    If Not objUserGroup Is Nothing Then
        Set objUser = objUserGroup.GetUserObject("User01")
        objUser.WebClientAutoLogoffSecs = 90
    End If
    LogoffActiveUser
End If
End Sub

```

1.18.3. UserGroupCmdTarget

Functions

GetNumUsers, UserGroupCmdTarget Function

Syntax GetNumUsers()

Description Questa funzione restituisce il numero degli utenti appartenenti al gruppo referenziato.

Nota:

Le funzioni e le proprietà dei gruppi utenti sono accessibili solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore.

Parameter	Description
None	None

Result Integer

Example1:

Option Explicit

Sub Click()

```

Dim MyGroup As UserGroupCmdTarget
Dim MyUsersAndGroups As UserAndGroupCmdTarget
Dim MyUser As UserCmdTarget
Dim sUsersList As String
Dim numUsers As Integer
Dim i As Integer
Dim tmpUsers As String

'Login Administrator level
GetPasswordFromLevel(1023)
Set MyUsersAndGroups = GetUserAndGroup
If Not MyUsersAndGroups Is Nothing Then
    Set MyGroup = MyUsersAndGroups.GetGroupObject("Users")
    If Not MyGroup Is Nothing Then
        i = 0
        numUsers = CInt(MyGroup.GetNumUsers) - 1
    For i = 0 To numUsers Step 1
        Set MyUser = MyGroup.GetUserAtPos(i)
        If Not MyUser Is Nothing Then
            sUsersList = sUsersList & MyUser.Name & "|"
            tmpUsers = tmpUsers & MyUser.Name & vbCrLf
        Set MyUser = Nothing
    Next i
End If
End Sub

```



```

        End If
        Next i
        MsgBox(tmpUsers,vbInformation + vbOkOnly,"Users List")
    End If
End If
LogoffActiveUser
Set MyUser = Nothing
Set MyGroup = Nothing
Set MyUsersAndGroups = Nothing
End Sub

Example2:
Sub Click()
    Dim MyGroup As UserGroupCmdTarget
    Dim MyUsersAndGroups As UserAndGroupCmdTarget
    Dim MyUser As UserCmdTarget
    Dim sUsersList As String
    Dim numUsers As Integer
    Dim i As Integer
    Dim tmpUsers As String

    Set MyUsersAndGroups = GetUserAndGroup
    If Not MyUsersAndGroups Is Nothing Then
        'administrator login for use the group functions
        MyUsersAndGroups.LogonUser("Daniele","Daniele")
        Set MyGroup = MyUsersAndGroups.GetGroupObject("Users")
        If Not MyGroup Is Nothing Then
            i = 0
            numUsers = CInt(MyGroup.GetNumUsers) - 1
            For i = 0 To numUsers Step 1
                Set MyUser = MyGroup.GetUserAtPos(i)
                If Not MyUser Is Nothing Then
                    sUsersList = sUsersList & MyUser.Name & "|"
                    tmpUsers = tmpUsers & MyUser.Name & vbCrLf
                    Set MyUser = Nothing
                End If
            Next i
            MsgBox(tmpUsers,vbInformation + vbOkOnly,"Users List")
        End If
    End If
    LogoffActiveUser
    Set MyUser = Nothing
    Set MyGroup = Nothing
    Set MyUsersAndGroups = Nothing
End Sub

```

GetUserAtPos, UserGroupCmdTarget Function

Syntax GetUserAtPos(_Position)

Description Questa funzione restituisce l'utente appartenente al gruppo referenziato alla posizione i-esima passata come parametro.

Nota:

Le funzioni e le proprietà dei gruppi utenti sono accessibili solo se si è effettuato il login di un utente (tramite ad esempio la funzione "LogonUser" o il login di un utente da console) con un livello pari o superiore a quello di amministratore.

Parameter	Description
Position As Integer	Indice di riferimento dell'utente i-esimo

Result Object
Viene restituito un oggetto di tipo UserCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Option Explicit
Public Sub Click()
    Dim MyGroup As UserGroupCmdTarget
    Dim MyUsersAndGroups As UserAndGroupCmdTarget
    Dim MyUser As UserCmdTarget
    GetPasswordFromLevel(0)
    Set MyUsersAndGroups = GetUserAndGroup
    If Not MyUsersAndGroups Is Nothing Then
        Set MyGroup = MyUsersAndGroups.GetGroupObject("Users")
        If Not MyGroup Is Nothing Then
            Dim numUsers As Integer
            Dim i As Integer
            i = 0
            numUsers = CInt(MyGroup.GetNumUsers) - 1
            For i = 0 to numUsers Step 1
                Set MyUser = MyGroup.GetUserAtPos(i)
                If Not MyUser Is Nothing Then
                    Debug.Print MyUser.Name
                    Set MyUser = Nothing
                End If
            Next i
        End If
    End If

    Set MyUser = Nothing
    Set MyGroup = Nothing
    Set MyUsersAndGroups = Nothing
End Sub
```

GetUserObject, UserGroupCmdTarget Function

Syntax GetUserObject(_IpszUserName)

Description Questa funzione restituisce l'utente appartenente al gruppo utenti referenziato. La funzione restituisce Nothing se l'utente referenziato non esiste all'interno del gruppo.

Nota:

Le funzioni e le proprietà dei gruppi utenti sono accessibili solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore.

Parameter	Description
IpszUserName As String	Nome dell'utente da reperire

Result Object
Viene restituito un oggetto di tipo UserCmdTarget se la funzione è stata eseguita con successo, altrimenti l'oggetto sarà Nothing.

Example:

```
Option Explicit
Public Sub Click()
    Dim MyGroup As UserGroupCmdTarget
    Dim MyUsersAndGroups As UserAndGroupCmdTarget
    Dim MyUser As UserCmdTarget
    GetPasswordFromLevel(0)
    Set MyUsersAndGroups = GetUserAndGroup
    If Not MyUsersAndGroups Is Nothing Then
```

```

        Set MyGroup = MyUsersAndGroups.GetGroupObject("Users")
        If Not MyGroup Is Nothing Then
            Set MyUser = MyGroup.GetUserObject("Guest")
            If Not MyUser In Nothing Then
                MsgBox MyUser.GetXMLSettings,vbOkOnly,""
            End If
        End If
    End If
    Set MyUser = Nothing
    Set MyGroup = Nothing
    Set MyUsersAndGroups = Nothing
End Sub

```

GetXMLSettings, UserGroupCmdTarget Function

Syntax GetXMLSettings

Description Questa funzione restituisce una stringa con il contenuto XML del file di progetto relativo al gruppo referenziato.

Nota:

Le funzioni e le proprietà dei gruppi utenti sono accessibili solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore.

Parameter	Description
None	None

Result String

Example:

```

Option Explicit
Public Sub Click()
    Dim MyGroup As UserGroupCmdTarget
    Dim MyUsersAndGroups As UserAndGroupCmdTarget
    Dim MyUser As UserCmdTarget
    GetPasswordFromLevel(0)
    Set MyUsersAndGroups = GetUserAndGroup
    If Not MyUsersAndGroups Is Nothing Then
        Set MyGroup = MyUsersAndGroups.GetGroupObject("Users")
        If Not MyGroup Is Nothing Then
            MsgBox MyGroup.GetXMLSettings,vbOkOnly,""
        End If
    End If
    Set MyUser = Nothing
    Set MyGroup = Nothing
    Set MyUsersAndGroups = Nothing
End Sub

```

Properties

CommandListLogoff, UserGroupCmdTarget Property

Syntax CommandListLogoff

Description Questa proprietà restituisce una stringa contenente il codice XML della Lista Comandi di LogOff associata al gruppo referenziato.

Parameter	Description
None	None

Result String

Example:

```
Sub Main()  
    Dim objUserAndGroup As UserAndGroupCmdTarget  
    Dim objGroup As UserGroupCmdTarget  
    Set objUserAndGroup = GetUserAndGroup  
    If Not objUserAndGroup Is Nothing Then  
        objUserAndGroup.LogonUser("Administrator", "Administrator")  
        Set objGroup = objUserAndGroup.GetGroupObject("Users")  
        If Not objGroup Is Nothing Then  
            MsgBox "CommandListLogoff = " & objGroup.CommandListLogoff,  
                vbInformation, GetProjectTitle  
            Set objGroup = Nothing  
        End If  
        Set objUserAndGroup = Nothing  
    End If  
End Sub
```

CommandListLogon, UserGroupCmdTarget Property

Syntax CommandListLogon

Description Questa proprietà restituisce una stringa contenente il codice XML della Lista Comandi di LogOn associata al gruppo referenziato.

Parameter	Description
None	None

Result String

Example:

```
Sub Main()  
    Dim objUserAndGroup As UserAndGroupCmdTarget  
    Dim objGroup As UserGroupCmdTarget  
    Set objUserAndGroup = GetUserAndGroup  
    If Not objUserAndGroup Is Nothing Then  
        objUserAndGroup.LogonUser("Administrator", "Administrator")  
        Set objGroup = objUserAndGroup.GetGroupObject("Users")  
    End If  
End Sub
```

```

        If Not objGroup Is Nothing Then
            MsgBox "CommandListLogon = " & objGroup.CommandListLogon,
                vbInformation, GetProjectTitle
            Set objGroup = Nothing
        End If
        Set objUserAndGroup = Nothing
    End If
End Sub

```

DefaultAccessLevel, UserGroupCmdTarget Property

Syntax DefaultAccessLevel = _Long

Description Questa proprietà imposta o restituisce il Livello Predefinito di Accesso assegnato agli utenti appartenenti al gruppo referenziato.

Nota:

Le funzioni e le proprietà dei gruppi utenti sono accessibili solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore.

Parameter	Description
None	None

Result Long

Example:

```

Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserGroupCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetGroupObject("Group1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.DefaultAccessLevel
            Set obj2 = Nothing
        End If
    End If
End Sub

```

DefaultEnableAutoLoggoff, UserGroupCmdTarget Property

Syntax DefaultEnableAutoLoggoff = _Boolean

Description Questa proprietà imposta o restituisce la configurazione di AutoLoggoff per gli Utenti appartenenti al Gruppo referenziato. Se posta al valore booleano True, verrà eseguito il Logoff automaticamente da Movicon dopo il tempo impostato nella proprietà "Logoff Automatico dopo (sec)". Il conteggio del tempo inizierà dal momento in cui l'utente non eseguirà più alcuna operazione sulla tastiera o con il mouse.

Nota:

Le funzioni e le proprietà dei gruppi utenti sono accessibili solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore.

Parameter	Description
None	None

Result Boolean

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserGroupCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetGroupObject("Group1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.DefaultEnableAutoLoggoff
            Set obj2 = Nothing
        End If
    End If
End Sub
```

DefaultExpiringDaysPassword, UserGroupCmdTarget Property

Syntax DefaultExpiringDaysPassword = _Long

Description Questa proprietà permette di definire il numero di giorni di default dopo i quali le Password degli Utenti appartenenti al Gruppo referenziato scadranno e non saranno più utilizzabili. Una volta scaduta, alla successiva autenticazione dell'Utente, verrà chiesto di inserire una nuova password. L'inserimento di un valore in questo campo ha significato soltanto se la proprietà dell'Utente "Modifica Password Obbligatoria" è stata abilitata. Se il valore di questa proprietà viene lasciato a zero, verrà ereditato il valore impostato nella proprietà "Durata Password (Giorni)" del Gruppo di appartenenza.

Nota:

Le funzioni e le proprietà dei gruppi utenti sono accessibili solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore.

Parameter	Description
None	None

Result Long

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserGroupCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetGroupObject("Group1")
    End If
End Sub
```

```

        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.DefaultExpiringDaysPassword
            Set obj2 = Nothing
        End If
    End If
End Sub

```

DefaultLevel, UserGroupCmdTarget Property

Syntax DefaultLevel = _Integer

Description Questa proprietà imposta o restituisce il Livello Predefinito di Password da associare al Gruppo.

I valori possibili per i livelli sono:

```

enum_UserLevel0 = livello 0
enum_UserLevel1 = livello 1
enum_UserLevel2 = livello 2
enum_UserLevel3 = livello 3
enum_UserLevel4 = livello 4
enum_UserLevel5 = livello 5
enum_UserLevel6 = livello 6
enum_UserLevel7 = livello 7
enum_UserLevel8 = livello 8
enum_UserLevelAdministrator = livello amministratore
enum_UserLevelDeveloper = livello sviluppatore

```

Nota:

Le funzioni e le proprietà dei gruppi utenti sono accessibili solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore.

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserGroupCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetGroupObject("Group1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.DefaultLevel
            Set obj2 = Nothing
        End If
    End If
End Sub

```

DefaultLogoffScript, UserGroupCmdTarget Property

Syntax DefaultLogoffScript = _String

Description Questa proprietà imposta o restituisce il nome dello script da eseguire al LogOff degli Utenti appartenenti al Gruppo referenziato.

Nota:

Le funzioni e le proprietà dei gruppi utenti sono accessibili solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserGroupCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetGroupObject("Group1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.DefaultLogoffScript
            Set obj2 = Nothing
        End If
    End If
End Sub
```

DefaultLogonScript, UserGroupCmdTarget Property

Syntax DefaultLogonScript = _String

Description Questa proprietà imposta o restituisce il nome dello script da eseguire al LogOn degli Utenti appartenenti al Gruppo referenziato.

Nota:

Le funzioni e le proprietà dei gruppi utenti sono accessibili solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
```



```

Dim obj2 As UserGroupCmdTarget
Set obj1 = GetUserAndGroup
If Not obj1 Is Nothing Then
    Set obj2 = obj1.GetGroupObject("Group1")
    Set obj1 = Nothing
    If Not obj2 Is Nothing Then
        Debug.Print obj2.DefaultLogonScript
        Set obj2 = Nothing
    End If
End If
End Sub

```

DefaultSecsAutoLoggoffTimeout, UserGroupCmdTarget Property

Syntax DefaultSecsAutoLoggoffTimeout = _Integer

Description Questa proprietà imposta o restituisce il tempo dopo il quale Movicon eseguirà l'Auto Logoff dell'eventuale Utente attivo appartenente al Gruppo referenziato. Naturalmente questa impostazione ha significato soltanto se la proprietà "Abilita Auto Logoff" è stata attivata. Il conteggio del tempo inizierà dal momento in cui l'utente non eseguirà più alcuna operazione sulla tastiera o con il mouse.

Nota:

Le funzioni e le proprietà dei gruppi utenti sono accessibili solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore.

Parameter	Description
None	None

Result Integer

Example:

```

Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserGroupCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetGroupObject("Group1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.DefaultSecsAutoLoggoffTimeout
            Set obj2 = Nothing
        End If
    End If
End Sub

```

Description, UserGroupCmdTarget Property

Syntax Description = _String

Description Questa proprietà imposta o restituisce un testo descrittivo relativo al Gruppo referenziato. La descrizione serve soltanto come promemoria per il programmatore e apparirà soltanto nella "Finestra delle Proprietà" del Gruppo.

Nota:

Le funzioni e le proprietà dei gruppi utenti sono accessibili solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserGroupCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetGroupObject("Group1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.Description
            Set obj2 = Nothing
        End If
    End If
End Sub
```

Language, UserGroupCmdTarget Property

Syntax Language = _String

Description Questa proprietà permette di leggere o impostare la lingua di default per il Gruppo Utenti selezionato. Impostando la lingua a livello di Gruppo la "Lingua" di default sarà valida per tutti gli Utenti appartenenti al Gruppo e che non hanno la stessa proprietà impostata. Inserendo una lingua in questa proprietà quando l'Utente appartenente al Gruppo esegue il Logon verrà automaticamente attivata quella lingua, a meno che la stessa proprietà "Language" non sia stata definita anche a livello dell'Utente, nel qual caso avrà la priorità la proprietà dell'Utente.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim oGroup1 As UserGroupCmdTarget

    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set oGroup1 = obj1.GetGroupObject("Group1")
        Set obj1 = Nothing
        If Not oGroup1 Is Nothing Then
            MsgBox("Group Language = " & oGroup1.Language, vbOkOnly,
                GetProjectTitle)
            Set obj2 = Nothing
        End If
    End If
```

```
End If
End Sub
```

Name, UserGroupCmdTarget Property

Syntax Name = _String

Description Questa proprietà imposta o restituisce il nome del Gruppo referenziato.

Nota:

Le funzioni e le proprietà dei gruppi utenti sono accessibili solo se si è effettuato il login di un utente (tramite ad esempio la funzione 'LogonUser' o il login di un utente da console) con un livello pari o superiore a quello di amministratore.

Parameter	Description
None	None

Result String

Example:

```
Option Explicit
Public Sub Click()
    Dim obj1 As UserAndGroupCmdTarget
    Dim obj2 As UserGroupCmdTarget
    Set obj1 = GetUserAndGroup
    If Not obj1 Is Nothing Then
        Set obj2 = obj1.GetGroupObject("Group1")
        Set obj1 = Nothing
        If Not obj2 Is Nothing Then
            Debug.Print obj2.Name
            Set obj2 = Nothing
        End If
    End If
End Sub
```

WebClientAutoLogoffSecs, UserGroupCmdTarget Property

Syntax WebClientAutoLogoffSecs = _Long

Description Questa proprietà consente di leggere o impostare il valore del tempo di auto logoff dell'utente Web Client collegato all'applicazione Server. Il valore impostato sul gruppo degli utenti viene preso in considerazione se la Gestione Password è attiva e la proprietà "Web Client Autologoff (sec.)" dell'utente è nulla. Per impostare la proprietà da Basic Script occorre che a runtime sia l'oggetto un utente di livello pari o superiore a quello di amministratore tramite, ad esempio, la funzione "LogonUser" o il login di un utente eseguito da console.

Parameter	Description
None	None

Result Long

Example:

```

Option Explicit
Public Sub Click()
Dim objUserAndGroup As UserAndGroupCmdTarget
Dim objUserGroup As UserGroupCmdTarget
    Set objUserAndGroup = GetUserAndGroup
    If Not objUserAndGroup Is Nothing Then
        objUserAndGroup.LogonUser("Admin","Admin")
        Set objUserGroup = objUserAndGroup.GetGroupObject("Users")
        objUserGroup.WebClientAutoLogoffSecs = 90
        LogoffActiveUser
    End If
End Sub

```

1.18.4. WorkspaceCmdTarget

Functions

OpenProject, WorkspaceCmdTarget Function

Syntax OpenProject(_IpszFilePath)

Description Questa funzione apre il progetto relativo al percorso passato come parametro. La funzione può essere eseguita solamente in design mode.

Parameter	Description
IpszFilePath As String	Percorso del progetto da aprire

Result Boolean

Example:

```

Option Explicit
Public Sub Click()
    OpenProject("C:\Movicon\Project1.movprj")
End Sub

```

OpenScreen, WorkspaceCmdTarget Function

Syntax OpenScreen(_IpszScreenName)

Description Questa funzione apre il sinottico referenziato. La funzione può essere eseguita solamente in design mode.

Parameter	Description
IpszScreenName As String	Nome risorsa sinottico

Result Boolean

Example:

```

Option Explicit
Public Sub Click()

```

```
        OpenScript("Screen1")
End Sub
```

OpenScript, WorkspaceCmdTarget Function

Syntax OpenScript(_IpszScriptName)

Description Questa funzione apre lo script referenziato. La funzione può essere eseguita solamente in design mode.

Parameter	Description
IpszScriptName As String	Nome risorsa script

Result Boolean

Example:
Option Explicit
Public Sub Click()
 OpenScript("Script1")
End Sub



Movicon è un sistema SCADA/HMI per Windows™ interamente progettato e realizzato da Progea.

© 2016 - Tutti i diritti riservati.

E' vietata la riproduzione o la divulgazione del presente manuale o del programma senza la preventiva autorizzazione scritta di Progea Srl.

Le informazioni contenute in questo manuale sono soggette a modifica senza preventiva segnalazione e senza comportare alcun vincolo all'ente realizzatore.



Via D'annunzio 295
41123 Modena - Italy
Tel. +39 059 451060
Fax +39 059 451061
Email: info@progea.com
Http://www.progea.com



Via XX Settembre, 30
Tecnocity Alto Milanese
20025 Legnano (MI) Italy
Tel. +39 0331 486653
Fax +39 0331 455179
Email: willems@progea.com



Progea Deutschland GmbH
Marie-Curie-Str. 12
D-78048 VS-Villingen
Tel: +49 (0) 7721 / 99 25 992
Fax: +49 (0) 7721 / 99 25 993
info@progea.de



Progea International Ltd
via Sottobisio 28
6828 Balerna - Switzerland
tel +41 (91) 9676610
fax +41 (91) 9676611
international@progea.com



Progea North America Corp.
2380 State Road 44 suite C
Oshkosh, WI 54904
Tel. +1 (888) 305 2999
Fax. +1 (920) 257 4213
info@progea.us