



Movicon NExT

17.0 Tec. specifications

Ver.3.4.268

Table of Contents

1. REQUIREMENTS	1
2. SERVER SPECIFICATIONS	5
2.1. TAGS	5
2.2. ENGINEERING UNITS.....	7
2.3. ALARM AND EVENT LOG.....	7
2.4. HISTORIAN.....	9
2.5. RECIPES.....	11
2.6. DRIVERS	12
2.7. OPC UA SERVER.....	13
2.8. EVENTS.....	14
2.9. REDUNDANCY	14
3. CLIENT SPECIFICATIONS	17
3.1. SCREENS	17
3.2. SHORTCUTS	19
3.3. MENU	20
3.4. OPC UA CLIENT	20
3.5. MULTI TOUCH.....	21
4. WEB CLIENT SPECIFICATIONS	23
4.1. WEB CLIENT	23
5. GENERAL SPECIFICATIONS	25
5.1. USERS AND GROUPS	25
5.2. SCRIPTS.....	25
5.3. TEXT AND STRING RESOURCE.....	26
5.4. WINDOWS SERVICES	27
5.5. CHILD PROJECTS	27
5.6. ALARM DISPATCHER.....	28
5.7. SCHEDULERS.....	28
5.8. NETWORKING.....	29

1. Requirements

Setup Requirements

In order to install Movicon.NExT the following pre-requisites must be met:

- Microsoft .NET Framework 4.7
- DirectX 9
- Reactive Extension v1.0.2856.0
- Microsoft SQL Server Express 2012

If the previous requisites are not satisfactory, the setup will download the necessary packets from internet and startup the relative installation.



The tables below are purely indicative. As it is impossible to know the details of applications created with Movicon.NExT in advance, it is the design engineer's sole responsibility to evaluate and use a hardware that is most suitable to the characteristics of the project to be created.

Editor Requirements

The table below reports the software and hardware requirements for developing Movicon.NExT projects using the editor.

Software Requirements	Hardware Requirements
Minimal O.S. Windows 8.1 Windows Server 2012 R2Windows 7 SP1 Windows Server 2012	Minimal CPU Intel® Core™ i5-5xxx (5th generation) RAM: 4 GB HD: Hard Disk with 4 GB of free space Graphic Card: integrated
Recommended O.S. Windows 10 Windows Server 2019 (*)Windows Server 2016	Recommended CPU Intel® Core™ i5-7xxx (7th generation) RAM: 8 GB HD: Solid State Disk with 4 GB of free space Graphic Card: dedicated

Server Requirements

The table below reports the software and hardware requirements for the Server side in Movicon.NExT projects.

Software Requirements	Hardware Requirements
Minimal O.S. Windows 8.1 Windows Server 2012 R2 Windows Server 2012 Windows 7 SP1	CPU Intel® Core™ i3-5xxx (5th generation) RAM: 4 GB HD: Solid State Disk with 4 GB of free space

Windows 7 Embedded Standard Windows 10 IOT Enterprise (*)	Graphic Card: Integrated
Recommended O.S. Windows 10 Windows Server 2019 (*) Windows Server 2016	Recommended CPU Intel® Core™ i5-5xxx (5th generation) RAM: 4 GB, HD: Solid State Disk with 4 GB of free space Graphic Card: integrated

Client Requirements

The table below reports the software and hardware requirements for Client side in Movicon.NExT projects.

Software Requirements	Hardware Requirements
Minimal O.S. Windows 8.1 Windows Server 2012 R2 Windows Server 2012 Windows 7 SP1 Windows 7 Embedded Standard Windows 10 IOT Enterprise (*)	Minimal CPU Intel® Core™ i3-7xxx (7th generation) RAM: 4 GB HD: Solid State Disk with 4 GB of free space Graphic Card: Integrated
Recommended O.S. Windows 10 Windows Server 2019 (*) Windows Server 2016	Recommended CPU Intel® Core™ i5-7xxx (7th generation) RAM: 8 GB, HD: Solid State Disk with 4 GB of free space Graphic Card: Integrated

Web Client - WPF (App) requirements

The table below reports the minimum software requirements and recommendations to publish Movicon.NExT projects for Web Client (WPF) access.

Server side Software requirements, Smartphone/Tablet Software Requirements.

Software Requirements	Hardware Requirements
Minimal O.S. Windows 8.1 Windows Server 2012 R2 Windows Server 2012 Windows 7 SP1	Minimal O.S. Android 4.1 and later iOS 9.3 and later Windows 10 Mobile (build 1607 and later)
Recommended O.S. Windows 10 Windows Server 2019 (*) Windows Server 2016	Recommended O.S. Android 7 and later iOS 10.3 and later Windows 10 Mobile (build 1709 and later)

Web Client - HTML5 (Browser) Requirements

The table below reports the minimum software requirements and recommendations to publish Movicon.NExT projects for Web Client (HTML5) access.

Server side software requirements, Client side software requirements.

Software Requirements	Hardware Requirements
Minimal O.S. Windows 8.1 Windows Server 2012 R2 Windows Server 2012 Windows 7 SP1 Other necessary software: Micorsoft IIS (Internet Information Service)	The Web technology is baed on the HTML5 standard and therefore is supported by any HTML5 compatible browser with any operating system, for example, such as: <ul style="list-style-type: none">• Desktop: Windows, Apple iOS, Linux, etc.• Mobile: Windows Phone, Apple iOS, Android, etc.
Recommended O.S. Windows 10 Windows Server 2019 (*) Windows Server 2016 Other necessary software: Micorsoft IIS (Internet Information Service)	

(*) Validation of hardware requirements, in evaluation phase.

2. Server Specifications

2.1. Tags

The Tags contain dynamic information and process data. They are managed in the platform's I/O Data Server module.

The Tags can be associated to field devices using drivers or connected to several systems using OPC UA.

Tags that are not connected externally are internal tags that are not connect for license purposes.

Functionality

- Tag names and descriptions are customizable
- Tag data types can be defined
- Structure variable types can also be defined with multiple levels (structures of structures)
- Array tag types can be defined for data types as defined below.
- String tag types can be defined.
- Tags can be made retentive (value is maintained when system shuts down).
- Two tags can be defined with the same name in two different folders: the tag's absolute name takes into account the complete path.
- Engineering Units can be assigned to tags (conversion and normalization and format rule).
- Tag Quality management for checking whether the tag has been updated correctly by the driver.
- default Initial values can be associated to tags
- Tags can be imported from the PLC database when communication driver provides this functionality.
- Generic Tag List import and export (e.g. in Excel files)
- Dynamic physical addresses of field devices can be associated directly to tags for connecting (driver, OPC or networking).
- Tags can be removed or updated automatically in various points of the project where they are being used (the refresh command in each screen is used to do this).
- Local project tags can be created. These tags will be used by objects on screen and can all be referenced using scripts.
- Aliases can be managed with parameter files
- Cross Reference can be used to see where tags are being used within the project and to trace project screen navigation commands.
- Digital and Enumerated models can be managed to display strings to substitute each number belonging to a list of numeric values.
- The "Method" model can be used to invoke methods (function) exposed by a driver.

Data Types

When using tags selected with Analog or Variable as model type will need to defined one of the following data types in their properties:

- Boolean
- Byte with sign (8 bit)
- Byte without sign (8 bit)
- 16 bit integer with sign (Int16)

- 16 bit integer without sign (UInt16)
- 32 bit integer with sign (Int32)
- 32 bit integer without sign (UInt32)
- 64 bit integer with sign (Int64)
- 64 bit integer without sign (UInt64)
- Float (32 bit single precision)
- Double (long) (64 bit double precision)
- String (Each byte character + 0 termination character). An Array can be created for each data type listed. Structure tags can be created when "ObjectType" model type is selected.
- When "Structure Prototypes" are presented on the card they will be listed as the Structure types available for applying to tags.

Tag pointing

Platform.NExT also permits tags to be addressed in the following modes:

Pointing to bits	By using the "Input Expression" field, accessible from any tag linkable object, you will be able to point to any one of the tag's bits by specifying the number using the < BitNumber> notation where the bit number starts with the "0".
Pointing to structure members	Any one of the structure tag members can be pointed by simply dragging the structure tag member from the 'Project's Resource' to the object that it is to be linked to. Alternatively, you can select the tag from the "tag item" menu which is access from the object's adorning.
Pointing to a array element	By using the "Input Expression" field, accessible from any tag linkable object, you will be able to point to an array tag element by specifying the number using the <[ElementNumber]> notation where the element starts from the "0" index.

Limitations

Maximum number of tags	Max. 128.000 Tags can be declared in one individual project. This is the total sum of Tag Elements or Structure Members or Array Elements Data exchanged with the field, according to license activation, is a Max. 100.000 Tags. Numbers exceeding this limit can be managed with special licenses.
Tag Names	The name of a tag must be composed of alphanumeric characters without spaces or special characters except for the ("_") character. The tag name cannot start with a special character or a number. The tag name can be a maximum of 64 characters long.

Structure Tag Size	Structure tags cannot exceed a total size of 32,767 bytes.
No. of Retentive Tags	Max 16,000 retentive tags per project. Please keep into account that the system saves the states of each retentive tag in a XML file. Each value variation is written in this file for each retentive tag. Excessive use may compromise system performance and physical memory duration.
No. of tags connected via Networking	Max. 32,000 Tags connected to other Platform.Next projects by means of using Networking
No. of tags with Statistics	Currently unavailable.

2.2. Engineering Units

The Engineering units are conversion tools that can be associated to Tags. They are used to convert the Tag's raw data to into a scaled engineering unit to be displayed in the project.

The unit to be displayed can also be defined > as a tag value suffix in display objects.

Functionality

Converts the data value to an engineering unit, by performing a mathematical conversion using a ratio between the min. and max. of the raw data and the min. and max of the converted data.

Each single Engineering Unit can be associated to one or several Tags.

Texts can be associated and used as "Measure Units" that can automatically be displayed in the linked display object.

Limitations

Max. number of U.I.	Max. 16,000 Engineering Units per project.
UI in Structures	The Engineering Units can be associated to structure members.
Associable Texts	Max. 64 Characters

2.3. Alarm and Event Log

The Alarm manager is used from the I/O Data Server module. The Alarm Manager is used to define thresholds that can be associated to Tag that establish the condition type used to generate Alarm notifications according to configurable functionalities.

Functionality

- Alarms incorporate one or several event threshold definitions.
- Option to group Alarms by Areas and Source.
- Option to enable/disable alarms during runtime based on tag values.
- Option to insert an alarm activation delay time (delay activation filter).
- Option to set exclusive or nonexclusive thresholds for analog alarms with several thresholds.
- Option to manage alarms only when the associated tag quality is good.
- Option to associate a text and help string to alarm thresholds and String ID (multilanguages) can also be used.
- Option to assign a fixed alarm activation value or manage a dynamic value by means of using a tag.
- Option to select different alarm activation logic conditions such as "GreaterThanOrEqual", "LessThanOrEqual", "Equals", "NotEqual", etc..
- Option to associate each alarm with a severity level.
- Option to associate each alarm with a command-state tag.
- Option to associate a command list to be executed on alarm event (ON, OFF, ACK and RESET).
- Option to define whether alarm should be acknowledged and reset or a simple message.
- Millisecond data precision.
- Option to change colors for displaying alarm based on severity in the Alarm Window, for activation state and for alarm acknowledgement.
- Option to force a comment by the operator when acknowledging the alarm. The comment is linked to the alarm's severity level ≥ 100 . The comment will be recorded in the historical log.
- Option to send alarm text as SMS and emails using the Alarm Dispatcher server. The notification can be sent once a certain alarm conditions is verified. It can also be sent to one or several preselected recipients.
- Option to associate Script code to the alarm window for managing ACK and Reset events.
- Alarms can be defined as Templates. This means that an alarm (template) can be created and associated to several tags.
- Option to use an Alarm Statistics Manager to analyze alarm events that occurred within certain time ranges (Statistical Downtime Analysis) with custom reports.

A Status File in XML format is created in the "ALARM" folder for each alarm defined in the project in runtime. This file is used for saving alarm information when the project is closed for example the alarm's status (ON, OFF, etc.), its total ON time, its unique and transaction ID.

Historical Event Log

- Automatic recording (excludible in the log properties) of each event linked to alarms or messages
- Records all system events
- Historical is displayed with filters such as 'by time range'.
- Print management
- Archive format in relational database (SQL Server is used for default). Open to different DB formats
- Supports Cloud (Azure)

Limitations

The below limits are to be taken into consideration when inserting alarms:

Alarms	Limitations
Alarms	Max. 16,000 alarm objects per project
Thresholds	Max. 4 thresholds for each individual alarm Thresholds are currently fixed at 1 (trip alarm) or 4 (exclusive / non exclusive level, rate of change, deviation) for OPC UA specification conformity. The 4 thresholds can be enabled/disables individually.
Text	Max. 256 characters for alarm text strings
Severity	Max. 1000 alarm severities

Historical	Limitations
Historical Log	Max. 999 days in archive. Note that this file capacity limit depends on the limits of the databased being used as indicated below.

Memory	<p>The Microsoft SQL Server Express database (free) is used if not specified otherwise in the settings. In this case and as indicated by the product specifications (https://msdn.microsoft.com/library/cc645993.aspx) the limitations are: Max 10 GB for archive files, max. 1 GB of RAM used.</p> <p>When using the Microsoft SQL Server standard version, the limitations are: Max 64 GB of archive files, Max 524 PB, 1 GB of RAM used.</p> <p>Other limitations depend on the file structure according to the producer of the Database format used and set in the project.</p>
--------	--

2.4. Historian

The Historian Manager, available as an option from I/O Data Server module, offers the Historian and Data Logger functionality. Both have the task to record process data on relational database using two data recording modes respectively.

Historian, Functionality

- The Historian uses a model that records a series of data with a column relating to the tag name and one relating to the value. These are typically used for displaying data in graphics based on time such as Trends and DataAnalysis (Time-series data)

- Uses XPO Providers for direct DB connections
- Historical recordings are performed in Databased tables that are completely transparent to the user.
- Data can be recorded 'on time' with relative sampling time settings.
- Recordings can be set with maximum and minimum recording rates based on fast changing or unchanged tag value rates. This will avoid over cramming data files or recording nothing if value should remain stationary.
- Recording can be set to record data 'on change'.
- Option to set custom connections to the preferred database (default SQL Server).
- Option to use the Microsoft Azure technology for recording in the Cloud.
- Option to set the max. age with which to keep data in the historical
- The Historian is associated as a Template to the tags to be recorded
- Option to record only when quality is good
- Option to record data based on absolute variations, tag value percent of the previous tag value or with reference to the ranges defined in the Engineering Units.

Data Loggers, Functionality

- Uses a data recording model where each Tag corresponds to a DB table column, typically adapted to data groups (e.g. statistics, production, Reports).
- Uses XPO providers for direct DB connection
- Each DataLogger records data on a table with the same name of the Datalogger
- Option to select to record data "on time" with relative sampling time settings
- Option to select to record data 'on command' using an appropriate Record tag
- Option to enable recording based on tag state
- Option to reset the table (cancel all records) based on a tag's state
- Option to use a custom connection to the preferred database (default SQL Server).
- Option to use the Microsoft Azure technology for recording in the Cloud.
- Option to set the max. age with which to keep data in the historical
- Option to customize the database table and column names
- Option to add extra columns for each tag to be recorded to report: Tag Quality, Tag TimeStamp relating ot the server and time of the device connected with the driver.

Limitations

Functionality	Limitation
Historians and Data Loggers	Max 1024 Historians - Data Loggers per project
Tags for Historian	Max. 1024 Tags for each Historian
Columns for Data Logger	Max 1024 columns for DataLogger. (This number may be reduced to a max. of 341 if using aggregated tables).
Archive age	Max 9999 days can be used for archive age. The maximum age will always depend on the database engine capacity chosen (eg. SQL Express uses Max. 2GByte).
Sampling time	Not lower than 300 milliseconds. The rate in which samplings can be taken is strictly tied to the quantity of data to be recorded and the

	driver's sampling time. These two parameters should be taken into consideration.
--	--

Memory	<p>The Microsoft SQL Server Express database (free) is used if not specified otherwise in the settings. In this case and as indicated by the product specifications (https://msdn.microsoft.com/library/cc645993.aspx) the limitations are: Max 10 GB for archive files, max. 1 GB of RAM used.</p> <p>When using the Microsoft SQL Server standard version, the limitations are: Max 64 GB of archive files, Max 524 PB, 1 GB of RAM used.</p> <p>Other limitations depend on the file structure according to the producer of the Database format used and set in the project.</p>
--------	--

2.5. Recipes

The Recipe Manager is used to create and manage data archives typically used for recording setpoint values linked to product names indexed in the archive.

The Manager permits users to set all the data values and settings and record them on DB files to be activated with the appropriated commands later when needed.

Functionality

- Used to build Recipe structures and define the graphical layout desired to display data.
- Records field values that relate to the recipe on a connected relational database (SQL Server is used for default)
- Supports data file format on custom DBs
- Supports data management in the Cloud with SQL Azure
- The recipe fields are divided into groups where each one can be set a task towards the field.
- Supports data transfer management from DB to Visualization and/or DB to PLC with direct data transfer management in one unique data block
- Tasks can be redefined in each member of the group they belong to
- Recipes can be called using a command or displayed through a RecipeViewer object
- Supports recipe management via Web Client
- Supports Import - Export to common text files

Limitations

Functionality	Limitation
No. of Recipes per project	Max. 512 recipes per project
No. of Columns per Recipe	Max. 512 Columns per Recipe

Memory	<p>The Microsoft SQL Server Express database (free) is used if not specified otherwise in the settings. In this case and as indicated by the product specifications (https://msdn.microsoft.com/library/cc645993.aspx) the limitations are: Max 10 GB for archive files, max. 1 GB of RAM used.</p> <p>When using the Microsoft SQL Server standard version, the limitations are: Max 64 GB of archive files, Max 524 PB, 1 GB of RAM used.</p> <p>Other limitations depend on the file structure according to the producer of the Database format used and set in the project.</p>
--------	--

2.6. Drivers

The Platform.NExT I/O Data Server Communication Drivers are functional modules, in the form of plug-ins, that manage communications according to the field device's data protocol to be connected to for exchanging data. The information contained in the device's memory areas are referred to by a "Physical Address" whose value is associated to the Tag.

Functionality

- Link to PLC addresses can be managed directly in the Tag properties
- Communications are made more efficient and optimized automatically. Automatic aggregation of data in Dynamic tasks and data exchange only with those tags that are effectively in use to improve and optimize performances
- Option to customise communications, connection management and the time used to interrogate the field for tags in use or and those not in use.
- Channel concept used for speeding up communication settings for devices with the same IP address or using the same serial port.
- Automatic Import from PLC's database/program or device
- Advanced debugging and Tracing Log functions
- Statistical communication information visualization (jobs being run for which channel, station...)
- Communication quality displayed directly in each object associated to a field tag

Performances

The communication performances depend on the quantity of data being exchanged simultaneously and the protocol being used.

For example, the table below shows various performance tests:

System	Details
PC	Win8.1, CPU i5 4 MB RAM
PLC	SIMATIC S7-314
I/O Driver	Siemens S7-TCP-IP

Exchanged Data	10,000 Tags (16 bit) simultaneously
Performances	1.15 secs refresh time

Limitations

The limitations to be taken into consideration while defining drivers are reported in the table below. Note: consistent data aggregation and good configuration will improve performances.

System	Limitations
Win64	Max 32 drivers simultaneously
Number of Stations	Max 128 Stations for each individual driver used.
Data Structure	Data structure support depends on the protocol being used.

2.7. OPC UA Server

The Platform.NExT I/O Data Server module is a native OPC UA server, certified according to the OPC Foundation standard.

OPC UA is defined as the international IEC 62541 standard.

Functionality

- Permits connection of any OPC UA client to read or write data
- Completely supports the data Information Model according to the OPC UA standard
- OPC Server Certification by the OPC Foundation test laboratories.
- Supports the DA specifications (Data Access)
- Supports the AC specifications AC (Alarms and Conditions)
- Supports the HA specifications (Historical Access)
- Supports different types of Transports

Limitations

System	Limitations
Server and connected Items	See paragraph: "OPC UA Connections and Networking"

2.8. Events

The Even Manager is a Platform.NExT resource that is used to activate the execution of one or more commands based on programmed events such as on status change, on events in Tags, or cyclic events according to scheduled times.

Functionality

- Possibility to enable events dynamically using tags
- Possibility to set condition in tags to determine events ; ">", ">=", "<", "<=", "<>", "on change".
- Possibility to managed events based on a predefined cyclic time schedule using minutes, hours, specific days, months and years.

Limitations

Functionality	Limitations
Server	Max 1024 Event Object per project
Server	Max 16 Sequential commands per each Event Object

2.9. Redundancy

The Redundancy management (optional) provides Hot Backup and Fault Tolerance for redundant mission critical station redundant networks. It permits an Active Server to perform as Primary and other active Servers to perform as Secondaries with full operativity and complete synchronization of all functional states and historicals being managed. Any connected Clients can manage eventual server switch over according to operativity mode and server load balancing purposes.

Functionality

- Fully automated surveillance and entry into redundant server array service management
- Fully automated data synchronization management
- Redundant communication driver management
- Historical and Datalogger synchronization management
- Active Alarms and Historicals redundancy management
- Archive and Server alignment functions
- Server names listed in hierarchical order of activation in redundancy mode.
- Option to use a number of threads dedicated to synchronizing historicals
- Redundancy tag to indicate the active server
- Option to align historicals to the time set or with a time range
- The last server that took over control can be kept active when the primary server is restored (managed only with two servers)

Limitations

Functionality	Limitations
Server	Max 4 servers managed simultaneously
Server	Scripts, recipes, shedulers and the alarm dispatcher are not managed in redundancy mode.
Tags	Max 100,000 Tags managed in redundancy
Tags	System tags are not managed in redundancy mode

3. Client Specifications

3.1. Screens

The screen resources are one of the essential elements of the Platform.NExT Client module projects defined Movicon.NExT. The screens are those windows that contain the Graphical User Interface (GUI) made up of graphical drawings, symbols and objects. Process Supervision or part of it is performed by means of these screens using graphical command for animations associated to the field tags.

The screen windows interface with the operator and can be activated as a whole screen or as pop-ups. Movicon also provides Embedded Screens where screens can be displayed inside other screens.

Screen Functionalities

- Screens are based on WPF with XAML vectorial graphics.
- Top quality Graphics Library and XAML toolbox
- Integrated Vector Graphics Editor
- Top level graphical effects (blur, trasparency, gradients, shadow, outer glow, etc.).
- Supports 3D graphics.
- Supports bitmap images (BMP, GIF, JPG, PNG) and multimedia formats
- Vector and 3D symbol settings in XAML format
- Centralized symbol library repository also can be managed in the Cloud
- Expandable and customizable libraries
- Screen contents are independent from screen resolutions
- Screen window sizes, background color and images can be set in their properties
- Supports multi-monitor.
- Full support to Multi-touch.
- Pop-up window management (modal or synchro).
- Option to keep screen loaded in memory when not displayed.
- Native Antialiasing functionality
- Possibility to use 32 display layers for screen contents
- Page navigation with system functions (Tiles, Gallery, Geolocalization) or traditional management with startup screen.
- Screen parameterization with alias.
- Engineering unit values can be used where provided in symbols and objects

Object and Symbol graphics functionality

Each graphical element can be assigned with dynamic animation properties. The dynamic functions are:

- Blink
- Opacity
- Composed Movement
- Horizontal Movement
- Vertical Movement
- Width
- Height
- Scale
- Rotation
- Border Color

- Back Color
- Border line
- Filling
- Storyboard
- Text
- Enable
- Visual State

Command Functionality for Toolbox Objects

Each object from the Toolbox library is predisposed to interact with the system by means of assigning tags or operation commands. The functionalities can be assigned to the object according to its type.

The object are divided into three categories:

1. Check Boxes = These objects are predisposed to manage the assigned tag values which are normally boolean type
2. Analog = These are object that are predisposed to manage the assigned analog variable value ((integer, float, etc.)
3. Commands = These are objects predisposed to execute operation commands (typically buttons or such like)

Below is a list of operation commands that can be assigned to command, menu, accelerator and event objects:

- Set value in Tag Actions
- Command actions in Alarms
- Command actions in Screens
- Command actions in Reports
- Command actions in Recipes
- Command actions in Users
- Command actions for change language
- Command actions in operating system or in projects
- Execution and launching script actions
- Calling methods with expressions actions
- Call 3D Views Actions

Limitations

The limitations to keep in mind when defining screens are listed in the table below:

System	Limitations
Screens	Max. 8192 screens in project. Max. 32 screens open simultaneously
Pop-up Screens	The screen background color transparency cannot be set when screen is in Modal and Frame mode. This option is however available for use in Syncro. screens.
Screen Contents	Max. 1024 vectorial elements per screen Max. 256 symbols or objects per screen Max. 256 basic elements for each symbol Max. 16 symbols contained in one symbol (nesting)

	Max. 16 animations simultaneously for each graphical symbol Max. 16 commands for each object
3D	Max. 1 3D model for each screen Max. 999 Elements for each 3D model Max. 32 Animations for each 3D model Max. 1 Inner Screen for each model Max. 32 Views for each model

Expressions instead of Tags

Objects and symbols also come with the option to receive logic expressions to determine the value being managed both in Input and Output. This functionality allows Tag values, formulae and calculations to be combined in order to represent values according to how the logic has been customized to be linked to object. The Expression method uses simplified syntax by using the same syntax used in the Microsoft Excel 2013 expressions.

System	Limitations
Expressions	Max 2 nesting levels and 8 operators.

3.2. Shortcuts

The Shortcuts resource is used to associate one or several commands or command lists to physical keyboard keys or key combinations for their immediate activation.

Functionality

- Activation of a command or a list of commands in association to keys or combo keys.
- Each shortcut object can contain a set of different keys, each one with a specific command.
- Shortcuts can also be associated to a specific screen for activating command keys based on which screen is active.
- All main screens can be set with a Generic set of command keys.

Limitations

Sytem	Limitation
Shortcuts	Max. 512 shortcuts per project.
	Max. 32 keys or combination keys for each shortcut.
	Max. 16 commands for each key or combination keys.

3.3. Menu

Platform.NExT has a Menu resource which can be used for creating interactive command menus. Each menu can be composed of command items in order to create user interaction by means of using symbol or object oriented contextual menus or customized general system menus or menus for each screen window.

Functionality

- Execution of a command or list of commands in association to each item on each individual menu
- Each Menu object can contain a set of items and dynamic items, each one with their own commands.
- Menus can be activated from any symbol or object, or from simple "hot regions" in each screen.
- Menus can be associated to specific screens to display in the active screen window's system menu bar.
- Menu Windows can be generic to all 'Main' screens

Limitations

System	Limitations
Menu	Max. 512 menu objects per project
	Max. 32 Items for each menu.
	Max. 16 commands for each menu item

3.4. OPC UA Client

The Platform.NExT OPC UA Client functionality permits connections to the OPC UA Server according to two different modes:

- Connecting as Client from the I/O Data Server module
- Connecting directly to display objects on the Client

The Movicon.NExT vizualization Client is natively a OPC UA Client and permits native browsing of third party OPC UA Server Items.
OPC UA is the International IEC 62541 standard

Functionality

- Permits connection to and browsing of any Server OPC UA Server to read or write data

- Completely supports the data Information Model according to the OPC UA standard
- Supports the DA (Data Access) specifications
- Supports the AC (Alarms and Conditions) specifications
- Supports the HA (Historical Access) specifications
- Supports different types of Transports

Limitations

System	Limitations
Server and connected Items	see paragraph: "OPC UA Connection and Networking"

3.5. Multi Touch

The multitouch operating system gestures and functionality are native to and supported in Platform.NEXT.

Functionality

- Independent from the hardware being used.
- Supports data scrolling, navigation and zooming according to the multitouch gestures.
- Supports object manipulation: each graphical object can be dragged, customized and zoomed as pleased within the screen. All operations are memorized according to the user who performed them.
- Supports object conditioned multitouch: commands can be activated by pressing two different objects at the same time (e.g. commands for security system)

Limitations

Functionality	Limitations
Multitouch	Pages are scrolled using 3 fingers to avoid making mistakes by touching wrong part of screen.
	Max. 32 manipulable objects per screen
	Max. 2 objects can be used for managing double security commands.

4. Web Client Specifications

4.1. Web Client

Platform NExT supports the web architecture to allow remote users to connect to Data Servers using the HTML5 technology. Screens can be created to deploy web pages using the integrated application module that uses the Microsoft IIS Web Server and a ASPX application module, which connects to the Data Server via OPC UA, to deploy screen pages using HTML5. The deploy feature creates HTML5 pages and manages connections to the Data Serve independently from the fact that the screens are displayed in the Movicon.NExT Client as well.

The mechanism to deploy and connect data is totally automatic and transparent. It offers the advantage of being able access plant system screens from any connected system using remote control with maximum interation and security.

Functionality

- The OPC UA Web Server integrated with automatic HTML5 screen deployment
- Cross platform: possible to access from PC, tablet, smartphone (e.g. Android, iOS, Linux..) by using various browers
- Deploy Wizard for automatically generating HTML5 Web server
- Access to all functionalities and controls on screen (with authentication option)
- Authentication and access to users defined in the project using the ASP.NET Membership provider.
- Total configurability in project designing objects, data and controls that are visible on web side
- Supports "pop-up" windows
- Complete Support to Alarms and Historical Events
- Supports Reports management
- Supports Trends and Historical Analysis
- Supports IP videocamera images managed from server
- Supports Web Sockets (full duplex) communications

Limitations

Functionality	Limitations
Screens	<p>The Multidocument is not supported by the page structure: therefore screen used as frames are not supported on the web side. However "pop-up" windows are supported.</p> <p>Objects that are not supported:</p> <ul style="list-style-type: none">• Web Browser• Media Elements• Earth 3D• Layout Windows• Digital Clock with transparent background• Combo Box: managed as displays• Slider: only read

Commands	<p>Commands not supported among those associated to graphical events are:</p> <ul style="list-style-type: none"> • System Commands • User Commands • Alarm Commands (use commands provided on web side) • Report Commands (use commands provided on web side) • Impulsive Tag command Value
3D	3D graphics not supported on Web side
Geo Scada	Maps are displayed on Web side in static mode. Interactive geographic and cartographic maps not supported
Connections	<p>Max. 10 instances simultaneously are supported on desktop operating systems via the web.</p> <p>Max. 100 instance simultaneously are supported on server type operating systems in servers via the web.</p>
Menù	The 'menu' resources are not supported.

5. General Specifications

5.1. Users and Groups

Platform.NExT manages user security according to strict standards using ASP.NET. Memberships. The system is therefore independent from the authentication type being used which for default is based on a SQL Server repository with the relative Microsoft authentication Provider. However other providers can be used for managing authentication (eg. biometric systems).

The project's Users management is used to define users with a series of parameters in addition to hierarchical level of privileges and access areas.

Functionality

- User management based on ASP.NET Membership provider, with maximum security and independent from authentication provider
- The authentication provider is based on the Microsoft SQL Server repository, If not specified otherwise.
- User rights can be edited and assigned in both edit and runtime modes (settable limit of users that can be inserted in runtime)
- Supports Groups and Users management
- Electronic Signature Management
- Supports the CFR21 Part 11 and GAMP5 specifications
- Users edited in runtime uses the same list of users defined in edit mode.
- User Authentication in Web Client HTML5 architecture as well.
- Option to share Windows Operating System users (local or network domain users) for authentication
- Option to protect the project with passwords and encryption (depends on the user management).
- Option to request authentication for Operating System operations: "ALT+TAB", "CTRL+ALT+CANC".

Limitations

System	Limitations
Users	Max. 9999 Users per project
Groups	Max. 512 User Groups
Levels	Max. 999 hierarchical levels and 31 access areas for each individual user.

5.2. Scripts

Platform.NExT offers the possibility to use the powerful VB.NET scripting language in your project applications on both Client and Server sides.

Functionality

- VB.NET scripting language, multithreading with Unicode support
- Editor powerful with guide, intellisense, debugger, simulazione
- Client side run with numerous API to resources, symbols and objects.
- Supports graphical object events and methods
- Option to encapsulate code in objects and create Power Templates
- Supports Dropping Code
- Option to create methods relating to tags: this code is run within the Server context
- Support to external Assembly and User Control .NET
- Runs script code started up as Windows service
- Script can be used to reference the project's global tags and local tags by writing the name of the PlatformNext tag or specific functions.
- API and specific interfaces used for accessing functions relating to screen resources, string tables, alarms, scripts and server.

Limitations

System	Details
Win64	Max 512 Script objects per project, of which t4 can be run in runtime simultaneously (multithreading).
Win64	Instances from the same script are not supported simultaneously
Web Client	Not available (run on server).

5.3. Text and String Resource

The String table and the files where the text used in the project is centralized.

Functionality

- Supports dynamic multilanguages
- Supports Unicode
- Automatic text translation (requires internet connection)
- Option to filter inserted strings
- Import-export strings to csv files
- Text contained in the String Table can be copied to the clipboard and pasted in Microsoft Excel, and viceversa

Limitations

Functionality	Details
Columns	Max 64 Languages (text columns)
Strings	Max 32,000 Strings for each language (records)
Characters	Max 1024 characters for each string

5.4. Windows Services

Platform.NExT supports the Windows Services as I/O Server and for running Script logic.

Functionality

- I/O Data Server Service support
- Scripts Service support
- The service can be set with user credentials predefined in system or by indicating a specific user.
- When using users defined in the operating system, the service can be protected from closing by those users who are not system administrators by hooking on to the users' set rules.
- This service, as such, results already active before logging on to the operating system.

Limitations

Feature	Limitations
Services	As a Windows service it inherits the features and limitations of these services.

5.5. Child Projects

The child projects render project development modular. This is done by means of associating and linking other projects to the project, both locally or in remote to create a modular and distributed project architecture.

Functionality

- Allows the project to be subdivided in modules: structured design.
- Simultaneous project development team work on 'child' projects to be inserted in the 'Parent' project
- The Parent project dynamically displays the resources defined in Child projects.

Limitations

Functionality	Limitations
	Max 32 Child Projects per Parent Project

5.6. Alarm Dispatcher

Alarm Dispatcher is an optional Platform.NExT Server module which sends Alarm and Event notifications to user recipients by using SMS and emails.

Functionality

- Options to use SMS or Email for sending notifications
- Notifications can be activated or deactivated according to variable condition
- Notifications via E-mail with attachments
- Notifications to individual recipients or group recipients

Limitations

System	Limitations
Server	Max 2048 notification events per project
	Max 32 Users can be associated to each individual notification
	Max. size email attachments is 512 Kbytes
	Max. 10 attachments per email

5.7. Schedulers

The Scheduler is a Server module that permits commands to be managed according to scheduled time plans or calendar dates using sophisticated settings.

Functionality

- The Scheduler Server is an integrated server module and as a plugin it is independent from the platform and based on OPC UA in transparent mode.
- Reoccurring or cyclic expiration can be configured as needed and in runtime as well.
- Events can be managed according to a predefined calendar or a weekly plan and modified in runtime as well.
- Integrated User and Password management for display schedules and modifying settings in runtime.
- ON and OFF events can be managed with start and end dates.
- Project Tags can be associated to schedulers
- Schedulers can be displayed and set from Web Clients.
- Event recordings and scheduler server log

Limitations

Functionality	Limitations
Server	Max 512 Scheduler Objects per project
	Commands that can be associated to a Scheduler are a set of values in Tags.

5.8. Networking

The OPC UA connectivity and the Networking consent remote Movicon.NExT clients to connect to Servers and viceversa.

The remote clients and servers can be either Platform.NExT application or third party OPC UA systems.

The connectivity is nevertheless based on the OPC UA data model and the Networking feature connects applications based on Platform.NExT.

When the networking feature is enabled on the license, a server will allow a local client to connect by means of using the "netpipe" transport.

Transports

- Net.Pipe: This is the Microsoft Transport that uses shared memory. It is the most performing transport due to being the only local one. In addition it is very safe to use because it does not open any listening port on the network card and therefore no one can enter from the outside. It is used when the client and the server reside in the same PC.
- Net.Tcp : This is a Microsoft transport based on the TCP protocol to consent communication over the net. This protocol is preferential for Movicon Server and Client as an alternative to using Opc.Tcp.
- Opc.Tcp : This transport, developed by the OPC Foundation, is based on the TCP protocol. Third party OPC UA servers or devices that implement the OPC UA server inside, can provide the use of this protocol.
- HTTP and HTTPS : This transport can be used in specific configurations in public networks with or without security management.
- NoSecurityHttp : HTTP transport without security management for transmitted data.

Limitations

Systems	Details
Net.Pipe	Max 100 concurrent sessions Max 10.000 monitored items
Net.Tcp	Max 50 concurrent sessions Max 5000 monitored items
Opc.Tcp	Max 50 concurrent sessions Max 5000 monitored items
HTTP e HTTPS	Max 25 concurrent sessions Max 2500 monitored items
NoSecurityHttp	Max 25 concurrent sessions Max 2500 monitored items

