

# Movicon NExT

## 3.0 Data Server

Ver.3.4.268



# Sommario

- 1. I/O DATA SERVER.....3
  - 1.1. CONCETTI GENERALI DEL SERVER DATI .....3
  - 1.2. AREA DI LAVORO DEL SERVER DATI.....3
  - 1.3. GESTIONE STORICI .....4
  - 1.4. IMPOSTAZIONI GENERALI SERVER.....8
  - 1.5. UNITÀ INGEGNERISTICHE ..... 10
  - 1.6. IMPOSTAZIONI AVANZATE DEL SERVER ..... 10
  - 1.7. COMUNICAZIONE CON I DISPOSITIVI..... 14
  - 1.8. DEFINIZIONE TRASPORTI DATI ..... 15
  - 1.9. STATO E LOG DEL SERVER ..... 17
  - 1.10. CONNESSIONE DEI CLIENTS AL SERVER..... 17
  - 1.11. FINESTRA IMPOSTAZIONE CONNESSIONE DB..... 20
- 2. INFO.....23
  - 2.1. CAMBIO ORA DEL PC..... 23



# 1. I/O Data Server

## 1.1. Concetti Generali del Server Dati

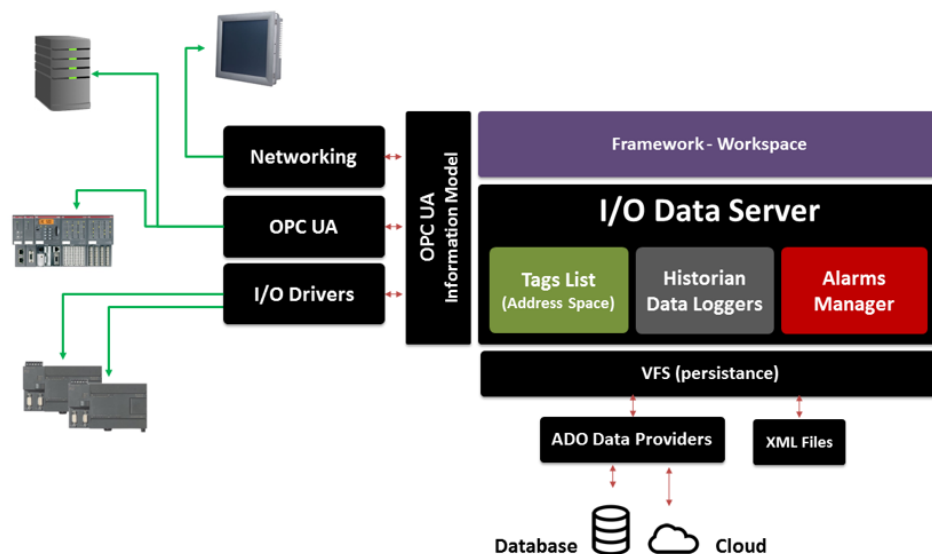
Il modulo I/O Data Server di Platform.NExT è il server di comunicazione e gestione dati, fondamentale per i progetti della piattaforma. Il Server dati provvede a gestire la comunicazione con il campo e con i sistemi di terze parti, definendo e centralizzando le informazioni dinamiche tramite la variabili, definite Tags, descritte nella risorsa Lista Tags (Address Space) del progetto.

Il Server inoltre ha il compito di gestire gli allarmi e la loro registrazione in un archivio storico di eventi, chiamato Log Storico.

Infine, come funzionalità opzionale, il Server Dati ha il compito di gestire la registrazione storica dei dati di processo, acquisiti sulle variabili Tag, attraverso il gestore degli archivi Historian e Data Logger.

Come si può vedere dallo schema illustrato sotto, il modulo Server di Platform.NExT ha il compito di gestire:

- **Comunicazione Dati**
- **Gestione Allarmi**
- **Gestione Registrazione Storici**

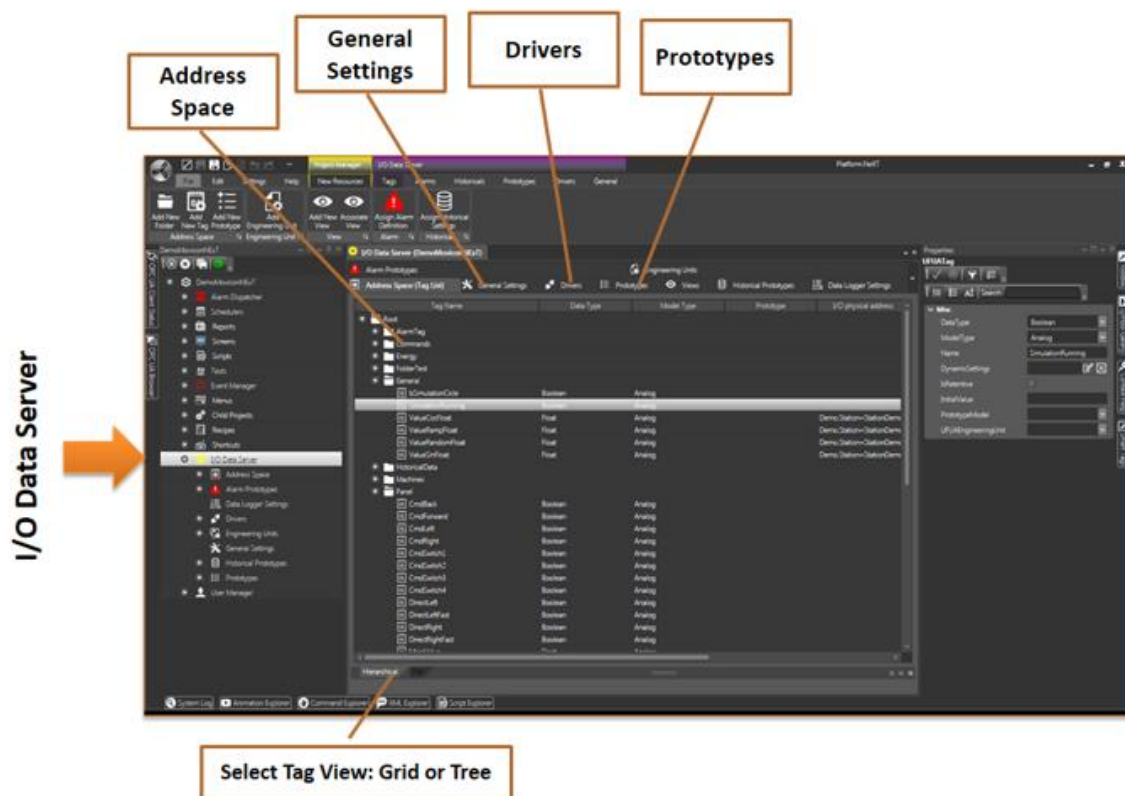


Lo schema illustrativo mostra la struttura del modulo Server dei dati, che prevede la gestione dei compiti di comunicazione, gestione allarmi e gestione storici.

## 1.2. Area di Lavoro del Server Dati

Il modulo funzionale I/O Data Server è presente nella struttura ad albero delle risorse del progetto. Aprendo questa risorsa, verrà visualizzata l'area di lavoro che conterrà le

impostazioni generali e l'accesso a tutte le configurazioni delle funzionalità del Server, inclusa la Lista delle variabili Tags e le altre configurazioni previste per il modulo, descritte in questa guida.



La visualizzazione della Lista Tag può essere impostata a seconda della propria preferenza, sia come visualizzazione "ad albero", con l'elenco gerarchico delle cartelle e dei Tags, oppure come "griglia", che consente le operazioni quali selezioni, filtri, ecc.

## 1.3. Gestione Storici

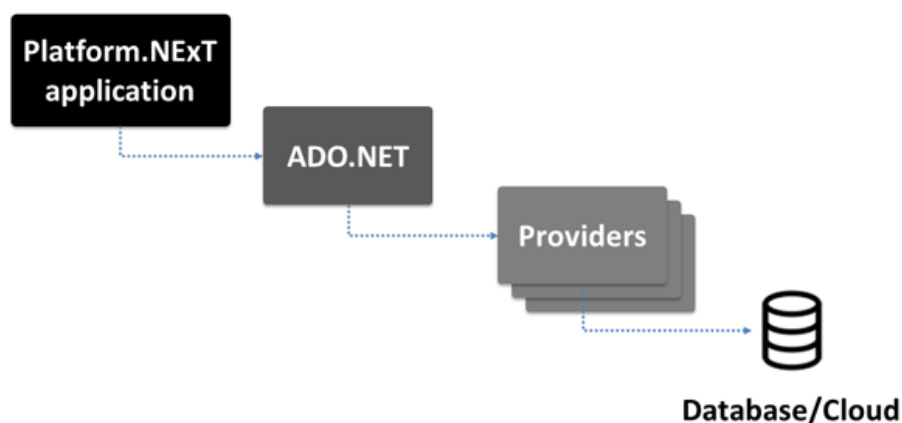
Il modulo I/O Data Server provvede, tra le altre cose, a gestire la registrazione degli archivi storici, attraverso le modalità e le configurazioni descritte negli appositi capitoli. La registrazione dei dati gestiti dal Server si articola attraverso i seguenti moduli funzionali:

- **Registrazione Eventi ed Allarmi**
- **Registrazione Dati di Processo Historian**
- **Registrazione Data Loggers**
- **Gestione Ricette**
- **Oggetti della Toolbox Collegabili ai DB (griglia, datalogger ecc...)**

Ciascun modulo funzionale svolgerà in modo indipendente la gestione dei propri dati e l'accesso al proprio database, utilizzando l'infrastruttura della piattaforma Platform.NExT.

## Connessione ai Database

Il modulo I/O Data Server di Platform.NExT utilizza la tecnologia di connessione ai database basata sui **Providers Dati** di ADO.NET. Grazie a questa tecnologia ogni progetto ed ogni risorsa di registrazione dati può essere indipendente dal formato dei dati che si desidera utilizzare.



## Le stringhe di Connessione

Grazie ai **Data Providers**, il Server dei dati è completamente indipendente dal tipo di database utilizzato nelle proprie architetture. E' possibile quindi personalizzare la stringa di connessione, nelle apposite proprietà di configurazione del Server Dati di Platform.NExT, per utilizzare il tipo di database preferito.

Se non diversamente specificato, il Server Dati prevede già, per default, una stringa di connessione che garantisce la registrazione dei dati automaticamente su database SQL Server.

Personalizzando però la stringa di connessione, è possibile definire un diverso Provider dati ed accedere quindi al database preferito.

La Stringa di Connessione contiene quindi le informazioni necessarie al Provider per stabilire la connessione al file di database e per gestirne i comandi, in modo totalmente trasparente ed indipendente dal progetto.

Siccome vi sono diversi Data Provider, è opportuno che l'utente verifichi correttamente, in funzione del Provider desiderato, come va impostata la stringa di connessione.

Queste infatti possono cambiare a seconda se ci si vuole connettere con i Provider per Oracle, MySQL o altri.


Come spiegato in precedenza tramite il pulsante a fianco possiamo effettuare la selezione tra le sorgenti dati "Sql Server", "SQL Azure", "MS Access", "XML file".

E' anche possibile però modificare manualmente le stringhe di connessione per ampliare la possibilità di connessione verso altri DataBase.

Database Engine	Version(s)	XPO Data Store Adapter	Stringa Di Connessione
Advantage	Server v11.1	AdvantageConnectionProvider	XpoProvider=Advantage;Data Source=\\myserver\\myvolume\\mypat\\mydd.add;ServerType=local;UserID=ASSSYS;TrimTrailingSpaces=true

ASA	SQL Anywhere 16, SQL Anywhere 17	AsaConnectionProvider	XpoProvider= Asa;Uid=MyUsername;PWD=MyPassword;DBF=c:\ mydatabase.db;Persist Security Info=true
ASE	Sybase Adaptive Server 12, Sybase Adaptive Server 15.5, Sybase Adaptive Server 16	AseConnectionProvider	XpoProvider= Ase;Port=5000;Data Source=MyAseServer; User ID=MyUserName;Password=MyPassword;Initial Catalog=MyDatabase;Persist Security Info=true
DB2	DB2 9.7.4	DB2ConnectionProvider	XpoProvider= DB2;Server=MyAddress:MyPortNumber; User ID=MyUserName;Password=MyPassword;Database=MyDatabase;Persist Security Info=true
Firebird	Firebird 1.5, Firebird 2.5.7	FirebirdConnectionProvider	XpoProvider= Firebird;DataSource=localhost;User=SYSDBA; Password=masterkey;Database=MyDatabase.fdb;ServerType=0;Charset=NONE
MSAccess (32bit Systems)	Access Database Engine	AccessConnectionProvider	<p>XpoProvider=MSAccess;Provider=Microsoft.ACE.OLEDB.12.0;Mode=Share Deny None;data source=C:\Temp\TestDB1.accdb;userid=Admin;password=;</p> <p> sarà necessario installare i driver acces a 32bit (<b>AccessDatabaseEngine.exe</b>) per le versioni  <a href="https://www.microsoft.com/en-us/download/details.aspx?id=13255">https://www.microsoft.com/en-us/download/details.aspx?id=13255</a> (Access 2010)  <a href="https://www.microsoft.com/en-us/download/details.aspx?id=54920">https://www.microsoft.com/en-us/download/details.aspx?id=54920</a> (Access 2016)</p> <p> Per impostare invece la connessioni di datalogger è possibile agire dalla finestradi configurazione ottenedo una stringa di configurazione come questa:</p> <p>DataProvider=System.Data.OleDb;  Provider=Microsoft.ACE.OLEDB.12.0;Data  Source=C:\Temp\TestDB1.accdb</p>
MSAccess (64bit Systems)	Access Database Engine		<p>XpoProvider=MSAccess;Provider=Microsoft.ACE.OLEDB.12.0;Mode=Share Deny None;data source=C:\Temp\TestDB1.accdb;userid=Admin;password=;</p> <p> sarà necessario installare i driver acces a 64bit (<b>AccessDatabaseEngine.exe</b>) per le versioni</p>



			<p><a href="https://www.microsoft.com/en-us/download/details.aspx?id=13255">https://www.microsoft.com/en-us/download/details.aspx?id=13255</a> (Access 2010)</p> <p><a href="https://www.microsoft.com/en-us/download/details.aspx?id=54920">https://www.microsoft.com/en-us/download/details.aspx?id=54920</a> (Access 2016)</p> <p></p> <p>Per impostare invece la connessioni di datalogger è possibile agire dalla finestradi configurazione ottenedo una stringa di configurazione come questa:</p> <p>DataProvider=System.Data.OleDb;  Provider=Microsoft.ACE.OLEDB.12.0;Data Source=C:\Temp\TestDB1.accdb</p>
MSSqlServer	Microsoft SqlServer 7.0, Microsoft SqlServer 2000, MS SQL Server 2000 Desktop Engine (MSDE 2000), Microsoft SQL Server 2005, SQL Server 2005 Express Edition, SQL Server 2008, SQL Azure™ Database, SQL Server 2008 R2, SQL Server 2008 R2 Express, SQL Server 2012, SQL Server 2012 Express (including LocalDB), SQL Server 2014, SQL Server 2014 Express (including LocalDB), SQL Server 2016, SQL Server 2016 Express (including LocalDB)		XpoProvider=MSSqlServer;Data Source=(local);User ID=username;Password=password;Initial Catalog=database;Persist Security Info=true
MSSqlServerCE	Microsoft SqlServer 2005 Mobile, SQL Server 2005 Compact Edition (Microsoft SqlServer 2005 Everywhere Edition CTP), SQL Server Compact 3.5, SQL Server Compact 4.0		XpoProvider=MSSqlServerCE;Data Source=MyDatabase.sdf;Password=MyPassword
MySql	MySQL Server 4.1, MySQL Server 5.0, MySQL Server 5.1, MySQL Server 5.7		XpoProvider=MySQL;Server=MyServerAddress;User ID=MyUserName;Password=MyPassword;Database=MyDatabase;Persist Security Info= true;Charset=utf8
Oracle	Oracle 9i, Oracle 10g, Oracle 11g, Oracle 12c		XpoProvider=Oracle;Data Source=TORCL;User ID=MyUserName;Password=MyPassword

Pervasive	Pervasive PSQL 9, Pervasive PSQL 10, Pervasive PSQL 11		XpoProvider= Pervasive;Server=MyServerAddress; UID=MyUserName;PWD=MyPassword;ServerDSN=MyDatabase
PostgreSQL	PostgreSQL 7, PostgreSQL 8, PostgreSQL 9, PostgreSQL 10		XpoProvider= Postgres;Server=127.0.0.1;User ID=MyUserName;Password=MyPassword;Database=MyDatabase;Encoding=UNICODE
SQLite	SQLite 3		XpoProvider= SQLite;Data Source=filename
VistaDB	VistaDB 4, VistaDB 5		XpoProvider= VistaDB;Data Source=C:\mydatabase.vdb4 XpoProvider= VistaDB5;Data Source=C:\mydatabase.vdb5



Se si utilizza MySQL dovrà essere installata la versione di Connector/NET 6.9.9



La stringa di "Connessione default Historian" verrà utilizzata anche dagli Historian, Data logger, Report e tutti gli oggetti dello screen che visualizzano dei dati prelevati da DB, a patto che non sia stata specificata una "connection string" personalizzata sulle proprietà di questi oggetti .

## 1.4. Impostazioni Generali Server

La finestra **Impostazioni Generali** del Server permette di configurare le impostazioni generali che determinano il funzionamento del Server Dati.

### Configurazione Impostazioni

La finestra di Configurazione Impostazioni dell'I/O Data Server contiene le impostazioni generali del modulo. Ogni singolo parametro di impostazione ha già preimpostato un valore di default che, nella maggior parte dei casi, è considerato ottimale, ma naturalmente i parametri si possono cambiare a seconda delle specifiche esigenze dell'utente.

#### Nome Applicazione

Nome dell'applicazione Server. E' possibile quindi personalizzare il nome con il quale il Server verrà visualizzato nella lista di selezione (browsing) degli eventuali Client (es. Client OPC UA di terze parti).

Oltre al Nome Applicazione si possono impostare altre informazioni quali "Manufacturer Name", "Product Name", "Product URI", "Software Version" e "Build Number" che verranno semplicemente pubblicate dal Server come informazioni aggiuntive o per la gestione dei Certificati di sicurezza.

#### Nome Produttore

Permette di definire il nome del produttore, questa informazione verrà pubblicata dal Server.

#### Nome Prodotto

Permette di definire il nome del prodotto, questa informazione verrà pubblicata dal Server.

**Indirizzo Prodotto**

Permette di definire un indirizzo URL riferito al prodotto o al produttore, questa informazione verrà pubblicata dal Server.

**Versione Software**

Permette di definire la versione software del prodotto, questa informazione verrà pubblicata dal Server.

**Numero Build**

Permette di definire la versione build del progetto, questa informazione verrà pubblicata dal Server.

**Connessione di default Historian**

Stringa di connessione che verrà utilizzata per default, se non diversamente specificato, dai Prototipi di Historian e DataLogger per la registrazione dei dati tramite il modulo Historian del Server.

Lasciando l'impostazione di default, il server utilizza la tecnologia di Microsoft SQL Server. Se si desidera connettersi ad un altro database, occorre utilizzare la stringa di connessione DB Provider ADO.NET adeguata, secondo le informazioni relative alle stringhe di connessione verso il DB desiderato.



Vedere i concetti sulla connessione ai Database attraverso Data Providers ADO.NET di questo manuale

**Connessione di Default Log Eventi**

Stringa di connessione che verrà utilizzata per default, se non diversamente specificato, dal motore di Registrazione Eventi e Allarmi (Log Storico).

Lasciando l'impostazione di default, il server utilizza la tecnologia di Microsoft SQL Server. Se si desidera connettersi ad un altro database, occorre utilizzare la stringa di connessione DB Provider ADO.NET adeguata, secondo le informazioni relative alle stringhe di connessione verso il DB desiderato.



Vedere i concetti sulla connessione ai Database attraverso Data Providers ADO.NET di questo manuale

**Max. Periodo Log Eventi**

Durata massima dei dati storici registrati nel Database degli Eventi ed Allarmi (Log Storico), permettendo quindi di dimensionare il periodo di tempo dei dati contenuti nell'archivio. Il valore di default è impostato a 365 giorni, e significa che raggiunto tale periodo di tempo, il sistema inizierà a riciclare sul dato più vecchio, che verrà quindi sovrascritto.

**Abilitazione Diagnostica**

Permette di abilitare la pubblicazione e la gestione di informazioni di diagnostica sul server a cui i client possono accedere.



Attenzione! L'attivazione di questa opzione potrebbe avere un impatto sulle performances del server, va quindi usata con cautela e soltanto se strettamente necessario e a fini di debug. Ad esempio, con l'opzione abilitata, in progetti con un numero corposo di Tag definiti nell'Address Space del Server l'avvio del Server potrebbe impiegare anche parecchi minuti.

## 1.5. Unità Ingegneristiche

La finestra "Unità Ingegneristiche"(Engineering Units) dell'I/O Data Server permette di inserire e configurare dei modelli di conversione dei dati, da un valore di entrata ad un valore di uscita calcolato e scalato.



Le unità Ingegneristiche sono importanti anche perchè permettono di configurare automaticamente i valori di scala di molti oggetti dell'interfaccia utente. Infatti, se presenti in associazione ad un Tag, e questo Tag è associato ad esempio ad un oggetto come il Gauge (visualizzatore a lancetta), questo adatterà automaticamente i valori di scala sul Min. e Max. secondo quanto impostato nei campi EU Range Low ed EU Range Max dell'unità ingegneristica.

### Generale

#### Nome Unità Ingegneristica

Definisce il nome da assegnare all'unità ingegneristica

#### Nome dell'unità di misura

Permette di definire un valore di testo (o una sigla) da assegnare all'unità ingegneristica

### Scala

#### Valore Minimo Normalizzato

Definisce il valore minimo di uscita, calcolato in corrispondenza ed in rapporto al valore grezzo di entrata

#### Valore Massimo Normalizzato

Definisce il valore massimo di uscita, calcolato in corrispondenza ed in rapporto al valore grezzo di entrata

#### Valore Minimo Grezzo

Definisce il valore minimo grezzo in entrata, sul quale verrà calcolato il valore ingegneristico di uscita

#### Valore Massimo Grezzo

Definisce il valore massimo grezzo in entrata, sul quale verrà calcolato il valore ingegneristico di uscita

## 1.6. Impostazioni Avanzate del Server

Sono disponibili una serie di **Impostazioni Avanzate** del Server, accessibili solo tramite la Finestra delle Proprietà, che permettono ad un utente esperto di andare a modificare alcune tra le impostazioni di dettaglio che determinano il funzionamento del Server Dati.

### Generale

per maggiori informazioni vedi il capitolo "impostazioni generali del server".

## Opzioni OPC UA Server

parametri di configurazione per la gestione delle sessione che il server può gestire nelle comunicazioni tra client e server.

### **Tempo Massimo Aggiornamento Dati**

Tempo massimo in ms. di pubblicazione dati supportato dal server.

Il server reimposta le sottoscrizioni che hanno un intervallo superiore, su questo valore.

Il valore ritornato dal server su una sottoscrizione, indica con quale tempo garantisce che il valore nel server venga aggiornato con il valore del campo.

### **Intervallo minimo di Aggiornamento**

Parametro che indica, in ms, la differenza minima tra l'intervallo di pubblicazione supportato.

Viene sempre usato per calcolare l'intervallo di sottoscrizione che viene ritornato dal server rispetto a quello che ha richiesto il client.

### **Durata Massima Sottoscrizione**

Definisce per quanto tempo le sottoscrizioni restano aperte senza una pubblicazione dal client.

### **Dimensione Max. Coda Messaggi**

Definisce il numero massimo di messaggi salvati nella coda per ogni sottoscrizione.

### **Dimensione Massima Coda Notifiche**

Definisce il numero massimo di notifiche salvate nella coda per ogni item monitorato.

### **Max. Nr. Notifiche per pacchetto**

Definisce il numero massimo di notifiche salvate nella coda per ogni pubblicazione.

### **Intervallo min. campionamento metadati**

Definisce il l'intervallo di campionamento minimo per i metadati.

### **Tempo Minimo Aggiornamento Dati**

Tempo minimo in ms. di pubblicazione dei dati supportato dal server.

Il server reimposta le sottoscrizioni che hanno un intervallo inferiore, su questo valore. Il valore ritornato dal server su una sottoscrizione, indica con quale tempo garantisce che il valore nel server venga aggiornato con il valore del campo.

### **Numero Max. di Operazioni "Browse"**

Numero massimo di "continuation points" usato per le operazioni di browsing da parte dei clients verso il Server.

Il "continuation point" è la necessità di effettuare una successiva richiesta al server per ottenere le restanti informazioni. Infatti il numero di informazioni ricevute dal client potrebbe essere incompleto a causa di un limite nel numero massimo di informazioni che il client può ricevere su ogni singola interrogazione.

### **Tempo Massimo Sessione**

Tempo massimo in ms. nel quale una sessione Client resterà aperta anche in assenza di comunicazione col Server. Si consiglia di lasciare il valore di default.

### **Tempo Minimo Sessione**

Tempo minimo in ms. nel quale una sessione Client resterà aperta anche in assenza di comunicazione col Server. Si consiglia di lasciare il valore di default.

**Max. Nr. di Sessioni**

Indica il numero massimo di sessioni client che possono essere aperte sul server.

**Esecuzione**

Raggruppano alcune delle impostazioni per la diagnostica e la gestione delle proprietà di sintesi vocale.

**Valore di Default Intervallo di Campionamento**

Imposta il tempo di aggiornamento dati minimo per la notifica dei valori Tag dal server al client. Se il client non specifica un tempo di aggiornamenti differente il server utilizzerà questo valore, espresso in msec.

**Abilita Diagnostica**

Permette di abilitare la pubblicazione e la gestione di informazioni di diagnostica sul server a cui i client possono accedere. L'attivazione di questa opzione potrebbe avere un impatto sulle performances del server, va quindi usata con cautela e soltanto se strettamente necessario e a fini di debug.

**Sintesi Vocale Abilitata**

Abilita la sintesi vocale per le notifiche dei messaggi di sistema.

**Nome sintesi vocale**

Imposta il nome del motore di sintesi vocale

**Ridondanza**

In questa sezione potremmo configurare le varie opzioni della ridondanza. per maggiori info. riguardo alle proprietà editabili in questa sezione vedere il capitolo "Impostazioni della Ridondanza".

**Impostazioni Avanzate DB**

Selezionando il Server Dati ed accedendo alla Finestra delle Proprietà, sono visibili alcune impostazioni aggiuntive, di tipo "avanzato" per configurare i dettagli dell'I/O Data Server. Ogni singolo parametro di impostazione ha già preimpostato un valore di default che, nella maggior parte dei casi, è considerato ottimale, ma naturalmente i parametri si possono cambiare a seconda delle specifiche esigenze dell'utente.

**Max. Durata Richiesta**

Durata massima delle richieste in ingresso. a notare che le richieste più vecchie verranno eliminate.

**Numero Max. di Operazioni "history Read"**

Numero massimo di "continuation points" (vedi sopra) usato per le operazioni di HistoryRead.

**Nr. Max. Accessi DB Storici**

Numero massimo di accessi contemporanei alla parte storica del server. Superato questo limite, la richiesta viene sospesa fino alla conclusione di uno degli accessi in corso.

**Nr. Max. Record in Errore**

Dimensione massima della cache dei dati per cui è fallita la scrittura. Al superamento di questo valore, tutti i dati che sono in attesa di un nuovo tentativo di riscrittura sul

database, verranno immediatamente scaricati su un file di testo in formato XML, nel disco locale in cui è in esecuzione il server.

**Nr. Min. Record in Cache**

Dimensione della cache dei dati in attesa di essere scritti sul database, sotto alla quale si può rientrare da una condizione di flush su file di testo locale in formato XML.

**Nr. Max. Record in Cache**

Dimensione massima della cache dei dati in attesa di essere scritti sul database. Raggiunto questo limite, i nuovi dati verranno inseriti nella cache predisposta per effettuare il flush sul file di testo locale in formato XML. Nel caso in cui venga raggiunto il limite indicato anche per questa seconda cache, i dati saranno irrimediabilmente persi.

**Num. Max Record da Eliminare**

Numero massimo di record eventualmente eliminati dal database su ogni controllo di età massima del database.

**Max. Dimensione File di Flush**

Dimensione massima (MBytes) di tutti i file di flush. Raggiunto questo limite i file di flush più vecchi che eccedono questa dimensione verranno definitivamente rimossi. Nel log viene tracciato con questo messaggio : "The safely size files exceeded the maximum allowed and some files has been definitively removed from path '{0}' !"

**Nr. Massimo Eventi Allarme**

Imposta il numero massimo di eventi allarme da caricare nel buffer.

## Impostazioni DB

Sono impostazioni che consentono di avere una connessione verso il DB, per la registrazione dei dati storici.

**Connessione Predefinita DB Storici**

Imposta la stringa di connessione predefinita a cui faranno capo tutte le risorse basate su DB storici.

**Connessione Predefinita DB Eventi**

Imposta la stringa di connessione predefinita a cui faranno capo tutte le risorse basate su DB Eventi.

**Durata Massima Dati in DB Eventi**

Imposta il tempo massimo di persistenza dei dati sul DB Eventi.

**Tempo di Attesa DB Storici**

Imposta il tempo di attesa tra un tentativo di connessione al DB e il seguente (espresso in secondi).

**Num. Massimo Tentativi di Scrittura**

Permette di impostare il numero massimo di tentativi di accesso al DB degli storici, prima di iniziare a scaricare i dati in cache su un file di testo locale.

## Audit

**Connessione di Default Audit Trail**

Consente di impostare l'eventuale stringa di connessione alternativa a quella di default del progetto che è rappresentata dalla stringa di connessione Historian Default connection nelle impostazioni generali del IO Data Server. Se impostata definirà un

database alternativo nel quale verrà creata una tabella UFUAAuditDataItem per tenere traccia delle registrazioni sulle tag sottoposte ad audit.

#### **Abilita Protezione Dati su File**

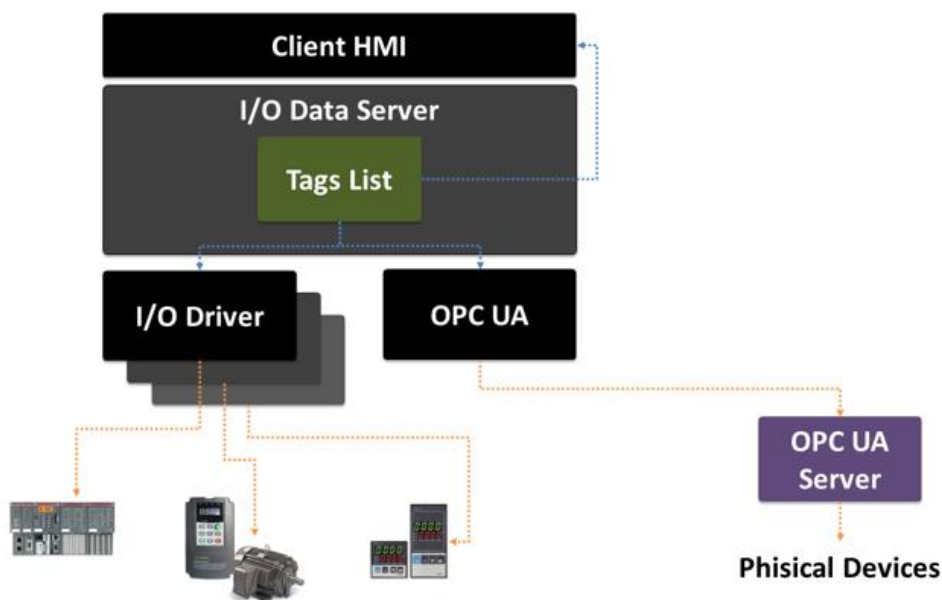
Consente di scegliere se proteggere le registrazioni di audit da manomissioni, involontarie o intenzionali che siano, introducendo una colonna con dati criptati nella tabella UFUAAuditDataItem. Le informazioni contenute in questa colonna impediscono la manomissione dei dati sensibili delle registrazioni senza compromettere l'integrità della cifratura.

## **1.7. Comunicazione con i dispositivi**

Uno dei compiti principali del modulo I/O Data Server è quello di gestire la comunicazione con i dispositivi in campo, attraverso i diversi canali di comunicazione che gestiscono i protocolli verso gli I/O fisici delle periferiche esterne.

La piattaforma quindi è in grado di comunicare in real-time con i dispositivi in campo, utilizzando opportuni drivers di comunicazione inclusi nel prodotto, oppure la tecnologia di connessione ai server OPC UA di terze parti. In ogni caso, tutte le informazioni raccolte dagli indirizzi fisici in campo sono associate alle variabili, definite Tags, che sono elencate nella risorsa Lista Tags, che in sostanza definisce quello che viene chiamato Address Space del Server.

La Lista dei Tag rende tutte le informazioni real-time visibili a tutte le risorse ed i moduli dell'intera piattaforma Automation Platform.NExT, sia nella parte Server che nella parte Client.



L'illustrazione mostra un esempio di architettura di comunicazione attraverso gli I/O Drivers oppure la tecnologia OPC UA

Nelle apposite sezioni della documentazione sono descritte le specifiche configurazioni dei Drivers, che dipendono da ogni specifico protocollo di comunicazione (Es. Siemens S7. Ethernet-IP, Modbus, Omron ecc.), oppure la tecnologia di comunicazione standard OPC UA.





I Drivers di Comunicazione dell'I/O Data Server sono installati con il server stesso, ma nuovi driver potrebbero essere disponibili successivamente all'installazione, scaricabili dal sito web di Progea, oppure potrebbero essere realizzati da terze parti.



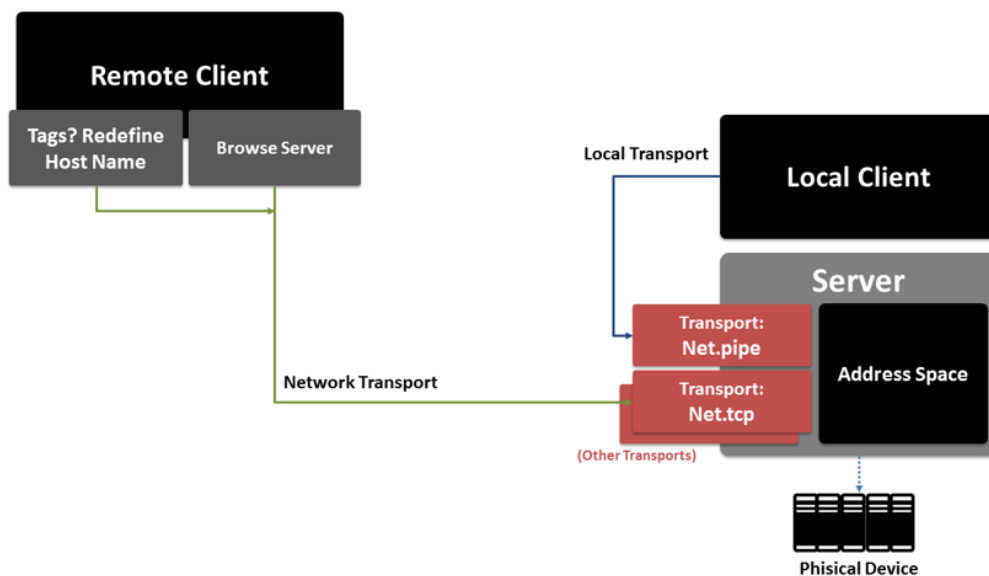
Platform.NExT è concepito affinché vi sia la possibilità per gli utenti più esperti di realizzare un proprio specifico driver di comunicazione, ed integrarlo nella piattaforma. Progea fornisce un apposito SDK per lo sviluppo del proprio protocollo "custom", con eventuale supporto e consulenza se necessario.

## 1.8. Definizione Trasporti Dati

Nelle Impostazioni dell'I/O Data Server è disponibile la finestra che elenca i Trasporti inseriti e disponibili per la comunicazione dati del progetto Server.

Tramite questa finestra si possono inserire quindi i "trasporti", ovvero i criteri di consegna dei dati che verranno poi utilizzati per la comunicazione tra Server e Client. Una volta avviato, il Server metterà a disposizione gli end-point secondo i trasporti abilitati nel progetto.

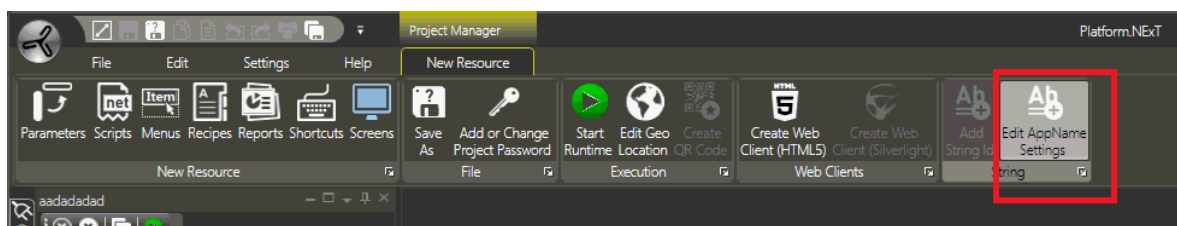
I Client, quindi (qualsiasi Client) potrà quindi utilizzare il trasporto preferito, o quello più idoneo al tipo di applicazione, in modo indipendente dal modello dei dati utilizzato.



Il Server Dati di Platform.NExT è appositamente concepito per rendere i dati indipendenti dal Trasporto che si desidera utilizzare, consentendo quindi di utilizzare architetture distribuite Client-Server basate sui trasporti più idonei al tipo di applicazione che si intende realizzare.

Dalla finestra di Impostazione del Server Dati, è possibile inserire, eliminare o modificare i Trasporti disponibili per un Server Dati, che potranno quindi essere utilizzati dai Client che intendono connettersi.

Le operazioni sui Trasporti sono impostabili attraverso l'apposita finestra di "Aggiunta Trasporto" o "Modifica Trasporto":



Questa finestra consente quindi di scegliere il Trasporto che si desidera aggiungere nel Server Dati, ed eventualmente configurare:

- Tipo di Trasporto
- Nome Server
- Porta
- Abilita/Disabilita

## Il Trasporto locale net.pipe

Per default, un progetto disporrà del trasporto "net.pipe". Questo trasporto è il trasporto che viene utilizzato, se non diversamente specificato, per connettere in una **applicazione locale** il Client al Server. Questo significa che il Client di visualizzazione, attraverso l'assegnazione dei Tags definiti nel Server agli oggetti grafici dei sinottici, verranno collegati utilizzando il Trasporto Locale definito net.pipe.

Questo trasporto è uno dei trasporti messi a disposizione da Microsoft WCF (Windows Communication Foundation) ed utilizza la memoria locale, garantendo le massime performances disponibili.

**Tuttavia, questo trasporto è solo "locale", quindi non consente la connettività in rete, che nel caso dovrà essere gestita attraverso l'inserimento perlomeno di uno degli altri trasporti previsti.**



Attenzione! Il trasporto "net.pipe" è un trasporto locale, che prevede che il Client e il Server siano avviati all'interno della stessa sessione di lavoro. Non è quindi possibile utilizzare tale trasporto se ad esempio il progetto Client e il progetto Server girano su macchine differenti.

In caso di utilizzo di Terminal Server o Remote Desktop, l'utilizzo del trasporto net.pipe può richiedere l'autenticazione utente con il livello di privilegio adeguato.

## I Trasporti di rete

Per consentire la connessione dei Client remoti verso il Server, occorre che questo disponga di uno dei Trasporti di rete, che deve quindi essere introdotto nel Server. Esistono diversi Trasporti di rete, e ciascuno presenta caratteristiche che ne privilegiano le performances oppure la sicurezza. E' quindi il progettista che può decidere quali trasporti utilizzare per le proprie architetture, in funzione delle proprie necessità o delle necessità del cliente.

L'ordine con i quali vengono utilizzati, ad esempio quando si collega una variabile ad un display, è il seguente:

- Net.Tcp
- Opc.Tcp
- Https
- Http
- NoSecurityHttp

Se non sono stati abilitati tutti i trasporti verrà utilizzato il primo disponibile nel progetto e, se non ne è configurato nessuno, verrà preso il primo che è presente nel file di configurazione del Server.

## 1.9. Stato e Log del Server

Nelle Impostazioni dell'I/O Data Server è disponibile la finestra che permette di visualizzare lo stato per la diagnostica della comunicazione dati del progetto Server. E' una finestra di log che si attiva quando si avvia manualmente il Server tramite il comando "Start Server" e che riporta i messaggi di stato o di errore del Server.

## 1.10. Connessione dei Clients al Server

La piattaforma Platform.NExT utilizza una architettura basata su Client e Server, con modello di informazione dati basato sullo standard OPC UA. Secondo questo modello, viene garantito all'utente la massima interoperabilità di scambio dati, consentendo ai Clients di connettersi al Server Dati secondo le proprie necessità, con la piena indipendenza dal Trasporto che si desidera utilizzare, dal modello di dati da scambiare e dalla connettività di Client NExT o di Client di terze parti.



Attenzione! La connessione dei Clients al Server richiede sempre l'abilitazione della Licenza sul Server. La funzione di Networking è generalmente abilitata salvo licenze speciali, e consente l'accesso dei Clients Movicon.NExT ad un Server di Platform.NExT.  
L'accesso di Clients OPC UA di terze parti richiede sempre l'opzione OPC UA Server abilitata sulla licenza acquistata.



Ricordiamo che la connessione del Client Locale è sempre consentita e prevista per default. Una applicazione locale Client-Server disporrà pertanto di una licenza Client-Server e di una connessione locale basata sul trasporto "net.pipe" sempre disponibile nelle configurazioni di default dei progetti.

### La definizione del Trasporto

La prima cosa da definire sul Server, per la connessione di eventuali stazioni Client Remote (oltre alla abilitazione della licenza, ovviamente), è la disponibilità del Trasporto desiderato, che deve quindi essere disponibile sulle "Impostazioni Generali" del Server Dati.

Tipicamente, una connessione tra un Client Movicon.NExT ed un Server Dati di un progetto Platform.NExT si potrebbe basare sul trasporto "Net.tcp", che è quello comunemente più utilizzato. Ma in base alle esigenze di sicurezza o a specifiche

necessità, il Server potrebbe disporre di altri Trasporti, come indicato nel paragrafo corrispondente.



Attenzione! Occorre definire almeno un trasporto di rete nel Server affinché un Client possa connettersi (come ad esempio il "net.tcp").

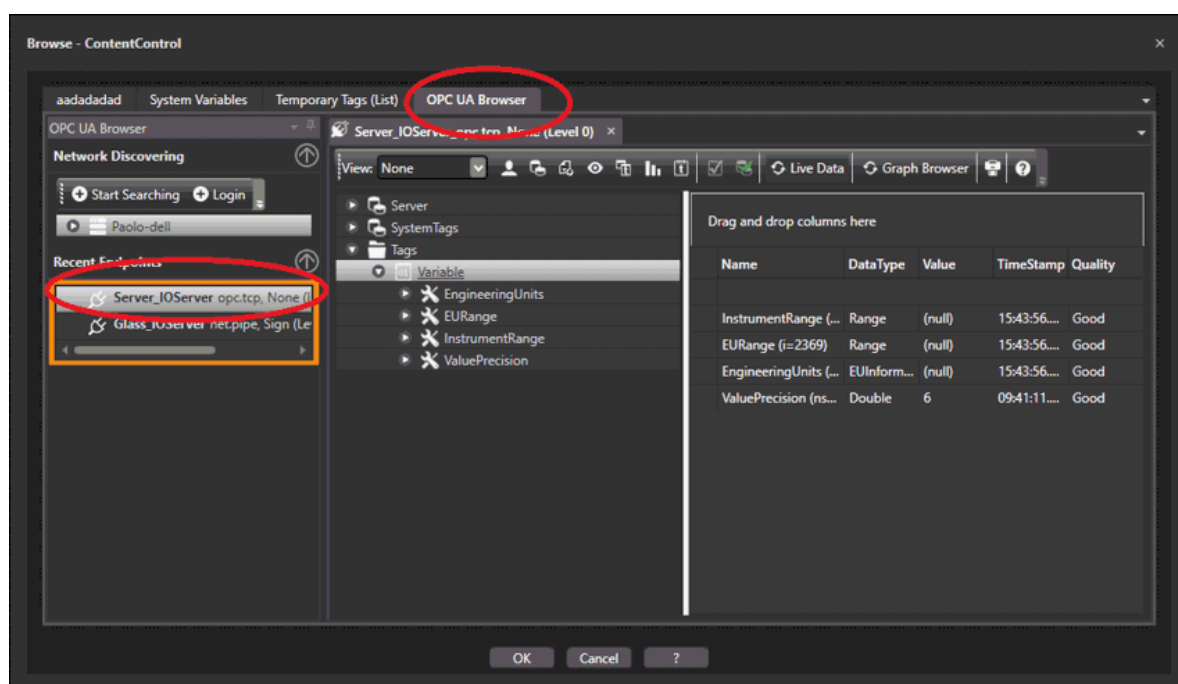
## Associazione dati da un Client verso un Server Remoto

Il Server Dati, attraverso il Trasporto di rete, esporrà le proprie variabili Tag utilizzando il Trasporto di rete preferito ed installato nel Server, secondo il modello di informazione dello standard OPC UA.

Tutte le informazioni disponibili nei Tags del Server saranno quindi accessibili da qualsiasi Client, sia dai sinottici di Movicon.NEXt che dagli eventuali Clients OPC UA di terze parti. Lo standard di connessione è lo stesso, seppure si possano utilizzare trasporti diversi.

Se si volesse quindi realizzare un progetto Client di visualizzazione, utilizzando i sinottici di Movicon.NEXt, è possibile associare le variabili agli oggetti grafici utilizzando il Browse delle Tag. Infatti, per ogni simbolo grafico od oggetto, è possibile associare una Tag dell'eventuale Server Locale oppure utilizzare lo standard di connessione e, mediante la medesima finestra di associazione dei Tags, selezionare il "Browse OPC UA", quindi selezionare il server dei dati tra quelli remoti disponibili e selezionare l'item (corrispondente al Tag del Server) al quale connettersi.

- **In questo caso, il progetto Client non avrebbe alcuna definizione locale del dato, ma sarebbe connesso direttamente al Server per gli item selezionati.**
- **Se invece si disponesse di un Client quale "copia" di un progetto con la definizione di dati Tag precedentemente collegate ad un server locale, occorre utilizzare le informazioni del paragrafo successivo.**



L'illustrazione mostra la finestra di associazione dei Tags agli oggetti, utilizzando la funzione di browsing OPC UA

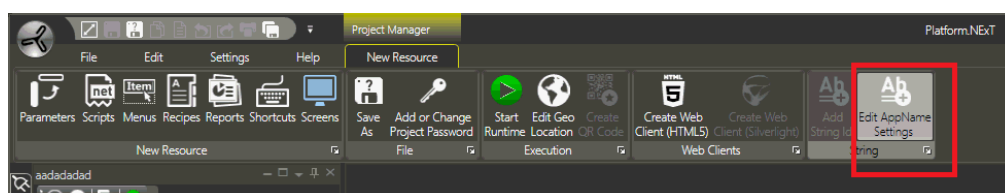
## Ridefinizione dei Tags locali di un Client verso un altro Server

Qualora si fosse nella necessità di ridefinire l'indirizzamento di un Server remoto al quale puntare, da un progetto Client già esistente e già quindi collegato a Tags che erano stati definiti localmente nell'address space, è possibile utilizzare la funzione di riassegnazione del Server al quale "puntare".

Infatti, tutte le variabili dei sinottici collegate ad una Tag List di un Server Locale utilizzano per default il trasporto "net.pipe" e sono collegate al Server locale di riferimento del progetto. Qualora il Client venisse "spostato" su un'altra stazione, avrebbe la necessità di puntare, in quanto Client Remoto, al server della stazione nella quale risiedono i dati.

La definizione del dato però è l'insieme di una serie di parametri secondo la specifica OPC UA, e il modo corretto e semplice per informare il progetto Client di dove puntare per accedere al Server, è quello di utilizzare lo strumento "Edit AppName Settings", che provvede ad indirizzare le connessioni non al server nella posizione originale, ma al nuovo indirizzo a cui puntare.

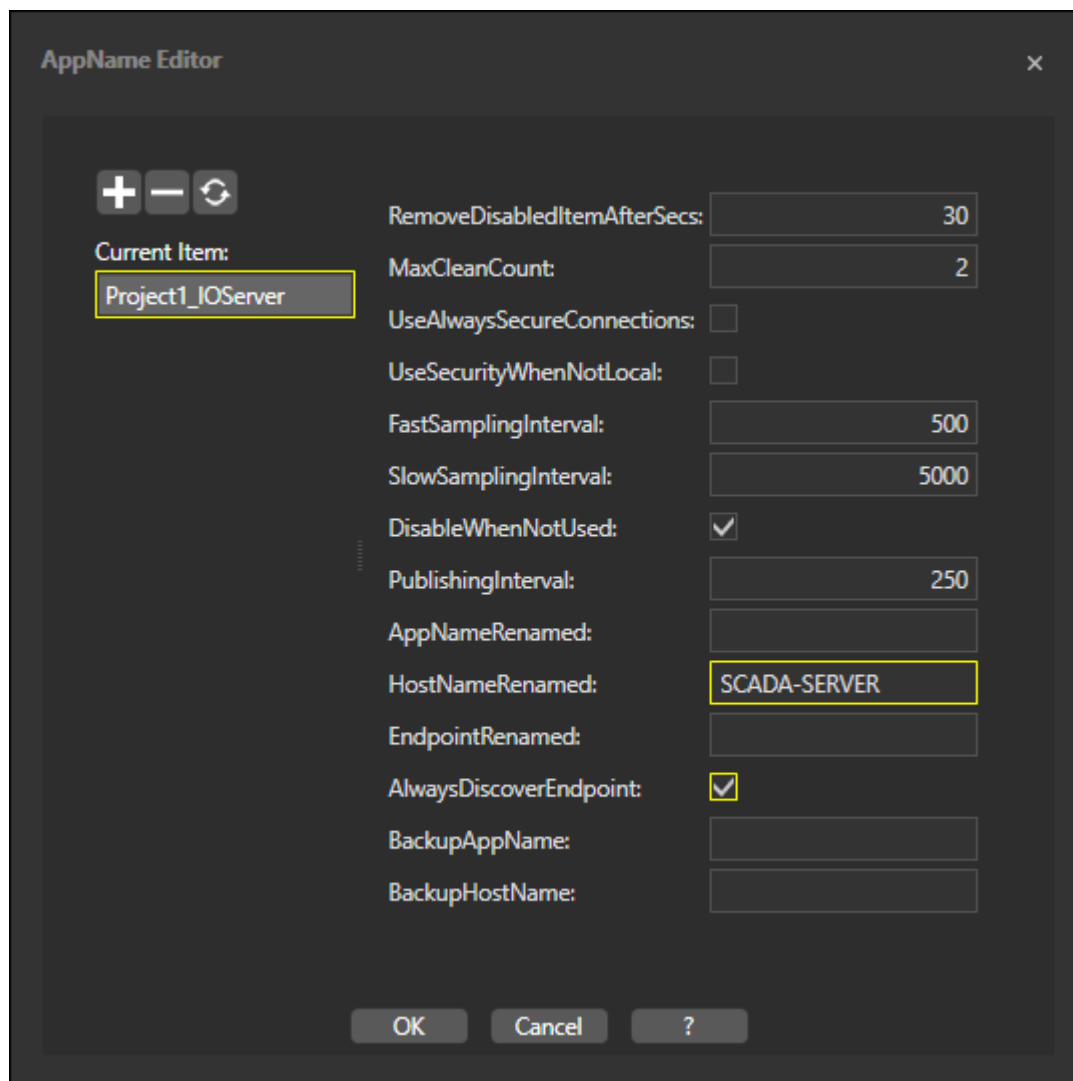
Questo comando è disponibile dall'apposito Ribbon del progetto come illustrato in figura.



Utilizzando il comando, apparirà una finestra nella quale occorre eseguire la "ricerca" del Server desiderato (se in esecuzione), oppure utilizzando il comando [+] per inserire il nome del server se già conosciuto.

Qui, è necessario definire i parametri :

- **Elemento Corrente:** usare il Nome Applicazione del progetto in uso
- **HostName Renamed :** definire il nome di rete del Server al quale occorre puntare
- **Always Discover EndPoint:** selezionare questa impostazione per fare in modo che il Client individui e sostituisca automaticamente gli endpoint precedenti con i nuovi ai quali puntare.



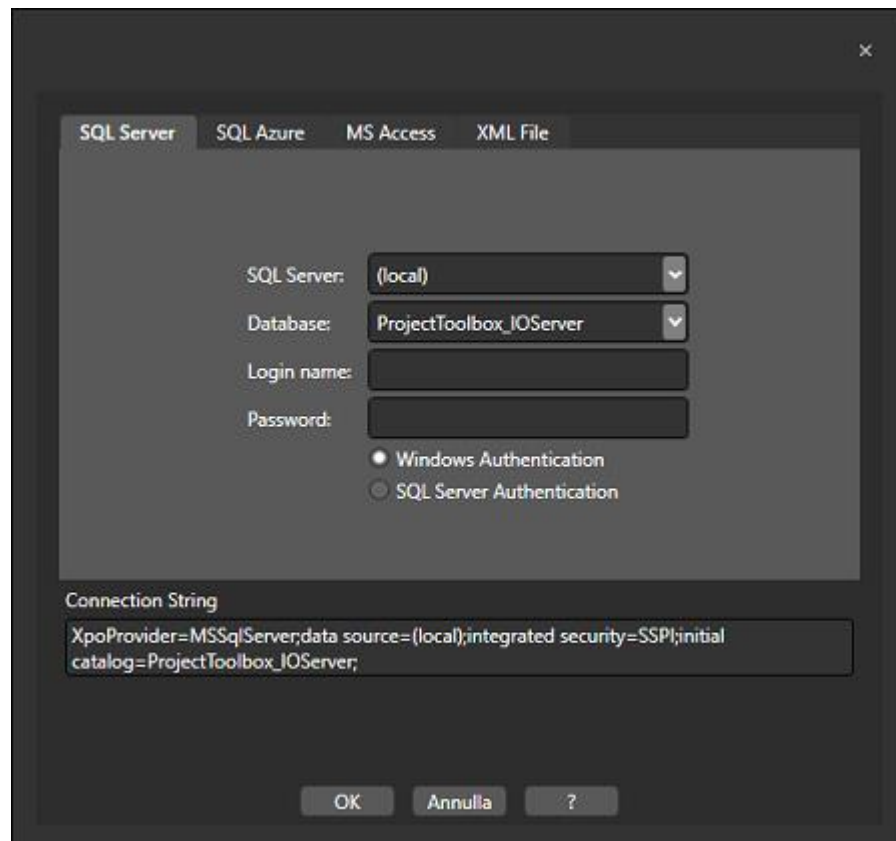
La figura mostra la finestra che permette di assegnare il nuovo nome del server al quale puntare per il progetto Client.

## Uso dei Progetti Figlio nelle architetture distribuite Client-Server

Nelle architetture di rete Client-Server di progetti Platform.NExT è disponibile la tecnologia dei progetti Padre e Figlio, per creare progetti distribuibili in rete che automaticamente saranno connessi al proprio Server di Riferimento. Consultare il Capitolo dei Progetti Figlio di Platform.NExT per maggiori informazioni al riguardo.

## 1.11. Finestra impostazione connessione DB

Tramite questa finestra è possibile definire la stringa di connessione verso il Data Base Selezionato tra quelli disponibili. Tale finestra sarà visualizzata dai vari elementi del progetto che dispongono della possibilità di gestire la sorgente dati a cui collegarsi , come ad esempio i Dati Storici Server I/O o oggetti di visualizzazione del Client (data Analysis, Visualizzatore log Storico...).



Sarà quindi possibile configurare il provider tramite le opzioni indicate (vedi figura).





## 2. Info

### 2.1. Cambio Ora del PC

Quando si esegue un cambio orario nel PC il Server scollega i tag e poi li ricollega. Questo comportamento è voluto. Sarebbe un problema di sicurezza se al collegamento fosse permesso di continuare quando si cambia la data (stepped forward). Molti elementi di sicurezza collegati potrebbero andare in timeout o scadere, la connessione deve essere chiusa e ricreata.

Attenzione però che in una situazione come questa, che normalmente non dovrebbe verificarsi se non quando cambia l'orario da legale a solare e viceversa, alcune risorse come ad esempio gli script potrebbero andare in errore se tentano di accedere ai tag nel momento in cui vengono scollegati. In questo caso bisogna prevedere la gestione dell'errore all'interno dello script.

