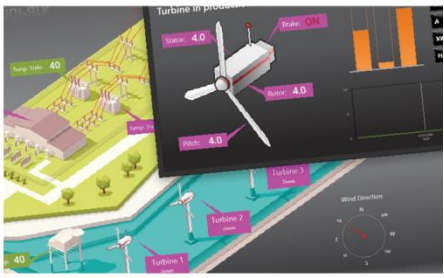
INDUSTRIAL AUTOMATION SOFTWARE

progea

Movicon.next™

Automation Platform.next generation



Movicon NExT

4.2 OPC UA

Ver.3.4.268

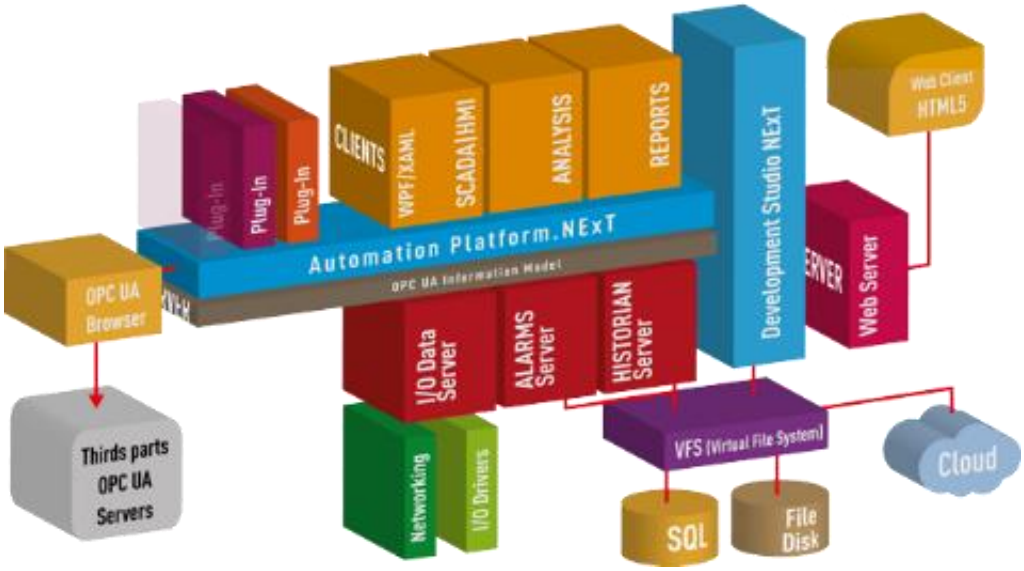
Table of Contents

1. OPC UA - IEC62541

1.1.1. OPC UA Information Model in Platform.NExT

The Automation Platform.NExT framework bases its data information model on the OPC UA technology, both for differentiating the Cient-Server infrastructure and in the complete development its functional parts:

- The I/O Data Server performs the function of Data Access (DA),
- The Alarm Server fulfills the role of Alarms and Conditions (AC)
- The Historian Server manages Historical Access (HA)



This exclusive Progea technology, which is totally transparent to users using the framework, offers the great advantage of using an open, expandible and modular platform that conforms to standards and consents full integration of simple or complex data models deriving from other levels of automation such as IEC61131-3 PLCopen control systems or modern MES/ERP systems.

1.1.2. OPC Unified Architecture (UA) IEC 62541

The OPC Unified Architecture (UA) was released in 2008 as an independent platform based on service-oriented architecture that integrates all the functionality of the individual OPC Classic specifications in one unique and extensible framework. This multi-layered approach makes it possible to achieve the original design specification goals of:

- Functional equivalence: all COM OPC Classic specifications are mapped to UA.
- Platform independence: from an embedded micro-controller to cloud-based infrastructure.

- Secure: encryption, authentication and auditing.
- Extensible: ability to add new features without affecting existing applications.
- Comprehensive information modeling: for defining complex information.

Functional Equivalence

building on the success of OPC Classic, OPC UA was designed to enhance and surpass the capabilities of the OPC Classic specifications. OPC UA is functionally equivalent to OPC Classic, yet capable of much more:

- Discovery: find the availability of OPC Servers on local PCs and/or networks.
- Address Space: all data is represented hierarchically (e.g. files and folders) allowing for simple and complex structures to be discovered and utilized by OPC Clients.
- On-demand: read and write data/information based on access-permissions.
- Subscriptions: monitor data / information and report-by-exception when values change based on client's criteria.
- Events: notify important information based on client's criteria.
- Methods: clients can execute programs, etc. based on methods defined on the server.

Platform Independence

Given the vast range of available hardware platforms and operating systems, platform independence is essential. OPC UA functions on any of the following and more:

- Hardware platforms: traditional PC Hardware, cloud-based servers, PLCs, micro-controllers (ARM etc.)
- Operating Systems: Microsoft Windows, Apple OSX, Android or any distribution of Linux, etc.

OPC UA provides the necessary infrastructure for interoperability across the enterprise, from machine-to-machine, machine-to-enterprise and everything in-between.

Security

Security is one of the most important things to consider when choosing which technology to use. OPC UA is firewall-friendly while addressing security concerns by providing a suite of controls:

- Transport: numerous protocols are defined providing options such as the ultra-fast OPC-binary transport or the more universally compatible SOAP-HTTPS, for example.
- Session Encryption: messages are transmitted securely exactly at 128 or 256 bit encryption levels.
- Message Signing: messages are received exactly as they were sent.
- Sequenced Packets: exposure to message replay attacks is eliminated with sequencing
- Authentication: each UA client and server is identified through OpenSSL certificates providing control over which applications and systems are permitted to connect with each other.
- User Control: applications can require users to authenticate (login credentials, certificate, etc.) and can further restrict and enhance their capabilities with access rights and address-space 'views'.
- Auditing: activities by user and/or system are logges provding an access audit trail

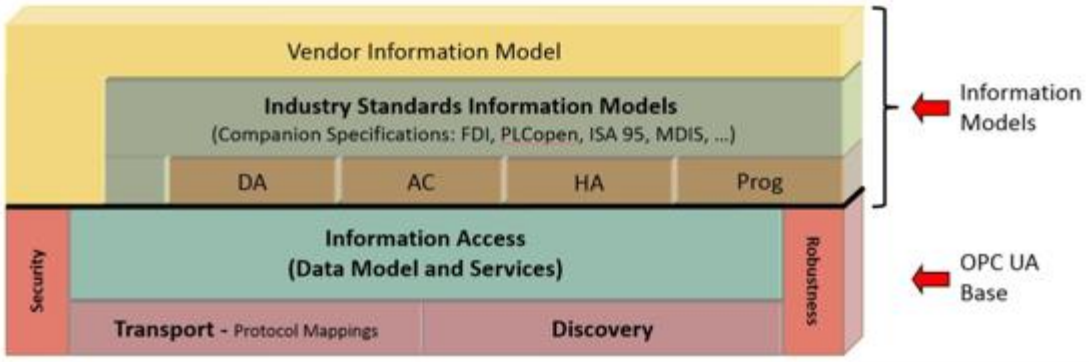
Extensible

The multi-layered architecture of OPC UA provides a 'future proof' framework. Innovative technologies and methodologies such as new transport protocols, security algorithms, encoding standards, or application-services can be

incorporated into OPC UA while maintaining backwards compatibility for existing products. UA products built today will work with the products of tomorrow.

Information Modelling

The OPC UA information modelling framework transforms data into information. With complete object-oriented capabilities, even the most complex multi-level structures can be modelled and extended. Data-types and structures are defined in profiles. For example, the existing OPC Classic specifications were modelled into UA profiles which can also be extended by other organizations. :



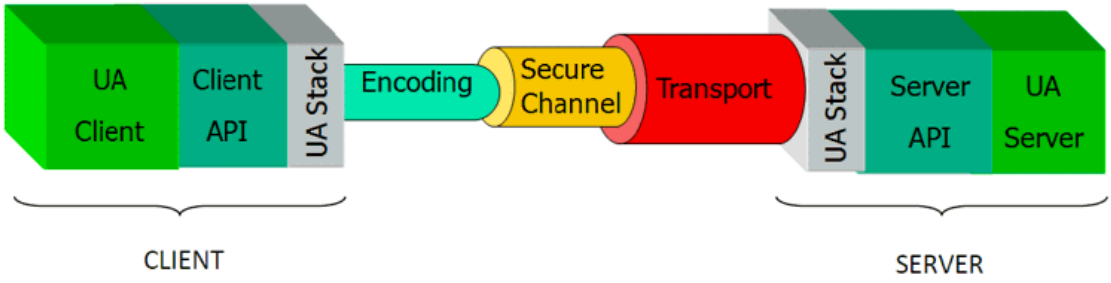
OPC UA Base Services Architecture

More info:
https://opcfoundation.org/about/opc-technologies/opc-ua/

1.2. Independent Data Transport

Transport is one of the essential components of OPC UA data modules. Transport is the infrastructure by means of which data is connected between Client and Server. In Platform.NExT data are independent from the transport you wish to use or need to use for connections towards third party systems in OPC UA.
Different types of transport can be used according to whether data is to be encoded or protected.

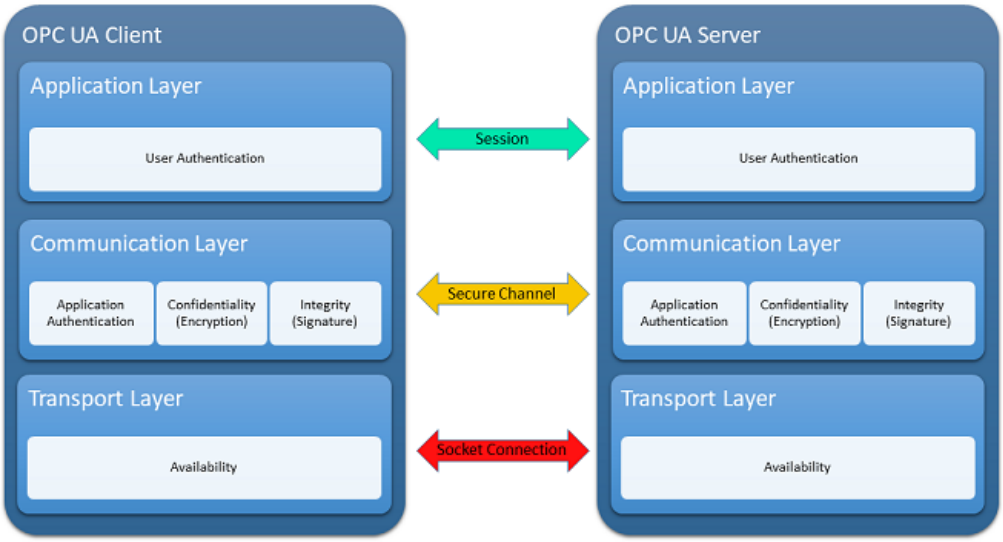
The transports offered by the Data Server are described in the topics on the Platform.NExT I/O Data Server.
In order to obtain communication between OPC UA Server and Client, you should always make sure that the transport you wish to use is appropriate and consistent with the purposes you intend to use it for.



As OPC UA uses a client-server architecture, it is normal that an application assumes both these roles. This is due to the fact that often the Server side is implemented in physical devices (device to device communication). A typical OPC UA application is composed of three software layers as shown in the above image.

1.2.1. OPC UA Security

The OPC UA security architecture is a solution that consents the implementation of security functions in various parts and has a multi-layer structure: Application Layer, Communication Layer and Transport Layer.



The Application Layer is used to transmit information between Client and Server which have established an OPC-UA Session on a Secure Channel (which is found at the Communication Layer). The Secure Channel secures the data exchanged in a session. The Transport Layer is responsible for transmitting and receiving data.
The security in OPC UA is realized through the defining certain security mechanisms that consent the Application Authentication in a Secure Channel and User Authentication in the Session.

Secure Channel definition

The Secure Channel, where the Application's Authentication takes place, ensures that data is exchanged safely in the following two modes:

- **Sign**: securing data integrity by using digital signatures. The security mechanism providing use of the digital signatures requires that:
-
    - Each side (Client and Server) possesses two keys (Public/Private Key)
    - Each side (Client and Server) makes their public key known
    - Each message exchanged between both sides is signed using the sender's private key and verified by the receiver with the sender's public key.

- **Encrypt**: securing the confidentiality of data by using asymmetric cryptography. The security mechanism used for asymmetric cryptography requires that:
-
    - Each side (Client and Server) possesses two keys (Public/Private Key)
    - Each side (Client and Server) makes their public key known
    - Encryption: Each message exchanged between both sides is encrypted using the public key of the counterpart and decrypted with each sides own private key.

The Secure Channel definition is created by applying a Security Profile and a Security Policy made available by the Server. The Security Profiles (also called Security Modes) are a set of three predefined security profiles:

- None: no security, channel is not safe.
- Sign: data integrity secured with digital signature (Application Certificate X.509). The digital signature is applied by the sender when exchanging data. The receiver can then verify whether the data actually comes from the expected sender.
- Sign & Encrypt: Data integrity and confidentiality secured with digital signature and encryption. First the sender's digital signature is applied to exchanged data which are then encrypted (Application Instance Certificate X.509).

The Security Policy indicates the length of the key and the algorithm used in the Security Profile; for example: Basic128Rsa15, Basic256, Basic256Sha256, Aes256-Sha256-RsaPss, etc...
The Security Policy and Security Profile combination forms the Security Level which indicates the degree of security of the Communication Level, where Level 0 is the lowest level. Clients can then choose the Security Policy and Security Profile simply by comparing them with the Security Levels.

Session definition.

At Application Level however, the security mechanisms that consent user authentication (and as a consequence their authorization) ensure access for specific users (and therefore their roles) during the Session configuration. There are three different authentication modes at this level: anonymous, with credentials (username/password), with digital signature (User Certificate X.509)

Secure Channel and Session establishment

The creation of a SecureChannel is principally based on the choice of Session Endpoint. Each OPC UA Server offers one or more Session Endpoints as follows:

- Endpoint Url: the endpoint's network address used by client to establish a SecureChannel
- Server's Application Instance Certificate: contains the Server's public key
- Security Policy: key length and algorithm used in the Security Mode
- Security Mode: none, Sign or Sign&Encrypt
- User Authentication: anonymous, username/password, User Certificate X.509
- Transport Protocol: specifies the stack characteristics used by the EndPoint: encoding, security, transport.

In order to establish a Session in a secure connection between an OPC UA Client and a OPC UA Server you need to carry out this steps:

1. First step: choose a Session Endpoint. The client selects a Session Endpoint and proceeds with validating the Server's Application Instance Certificate. If the certificate is trustworthy, the client can go ahead with the next steps.

2. The Client sends an Open Secure Channel request in accordance to the selected Session Endpoint Security Mode.
    - When Security Mode level is "None", the request is sent without using any security mechanisms.
    - When Security Mode level is "Sign" , the request is sent using the Private key associated to Client's Application Instance Certificate as the signature.
    - When the Security Mode level is "Sign&Encrypt", the request is sent using the Private key associated to Client's Application Instance Certificate as the signature along with the code using the Public Key of the Server's Application Instance Certificate (in addition to the signature).

3. The Server receives the Open Secure Channel message and validates the Client's Application Instance Certificate contained in the message. When the certificate returns valid, the Server proceeds with interpreting the request in accordance with the selected Security Policy and Security Mode.
    - When the Security Mode level is "None", the request is received without applying any security mechanisms.
    - When the Security Mode level is "Sign" , the request's signature is verified using the Public Key associated to the Client's Application Instance Certificate.
    - When the Security Mode level is "Sign&Encrypt" , the request is verified using the Public Key associated to the Client's Application Instance Certificate and then decrypted using the Private Key associated to the Server's Application Instance Certificate del Server

    When the Server accepts, it sends the response to the Client using the same method used for the request and a Secure Channel is then established.

4. Once the Secure Channel has been established, the Client proceeds with sending a Create Session request to the Server. Once created it then has to be activated by passing the User Credentials to the Server.

Appendix: asymmetric encryption and digital signatures

To respond to different needs, messages between two interlocutors can be encrypted or signed (or signed and then encrypted).

The asymmetric encryption, also known as a pair of cryptographic keys or a cryptographic public key is a type of encryption where messages are encoded by the agent who possesses a pair of keys:

- the 'private key', which is personal and secret, is used for decoding an encrypted document;
- the 'public key', which must be distributed, is used to encrypt a document destined to the person who possesses the relative private key.

In a cryptographic public key system, anyone can encrypt a message using the receiver's public key, but this message can only be decrypted with the receiver's private key.
The basic idea of asymmetric encryption becomes clearer when using a postal analogy where:

- the sender is Alice
- the receiver is Bob
- the locks are the public keys
- the keys to the locks are the private keys.

Communication between Alice and Bob takes place according to these steps:

1. Alice asks Bob to send him her padlock already opened. However the key to the lock is jealously guarded by Bob.
2. Alice receives the padlock and uses it to close the parcel then she sends it to Bob.
3. When Bob receives the parcel, he can open it with the key of which he is the only owner.
4. On the other hand, if Bob wants to send a parcel to Alice, he must ask her for the padlock so that he can use it to close the parcel. Once Alice receives the parcel, she can open it with her private key of which she is the only owner.

As you can see, in order to lock the parcel you need the receiver's padlock (public key) and to decrypt or in other words open the padlock you need to use the key of the padlock's owner (private key). This is how the asymmetrical encryption/decryption process works.

The digital signature is usually identified as the technology with which a document's authenticity is certified or data signed by who issued it. The digital signature system requires that the sender who issued the message uses their private key to generate information in association with the message to certify its origin.

It is important to note how both the encryption and the signature are based on the exchange of public keys. When the message is encrypted, the public key used is the one belonging to the receiver. When the message is signed, it is the sender's key that is used. In both cases, the value of the public keys is not confidential and the critical part lies in guaranteeing their authenticity. It must therefore be made certain that a specific public key effectively belongs to the interlocutor for whose message needs to be encrypted or whose signature needs to be verified. If a third party discovers the receiver's public key and replaces it with their own, the encrypted message contents will be unveiled and it will not be possible to verify the validation of a digital signature.
The distribution of public keys is a crucial problem in public key technology and is resolved by deploying electronic certificates. The public key certificates (X.509) are a reliable and secure tool with which public keys can be distributed and made known to end users while guaranteeing data authenticity and integrity.

The structure of an X.509 certificate is as follows:

- Version Number
- Serial Number
- Signature Algorithm ID: encryption algorithm used
- Issuer Name: identifies the Certificate Authority (CA) who issued and signed the certificate

- Valid from/to: validity period
- Public Key: Certificate's public key
- Signature: digital signature created by issuer. The digital signature imposed by using the issuer's Private Key.

Each X.509 certificated is associated with a Private Key.

2. OPC-UA Client

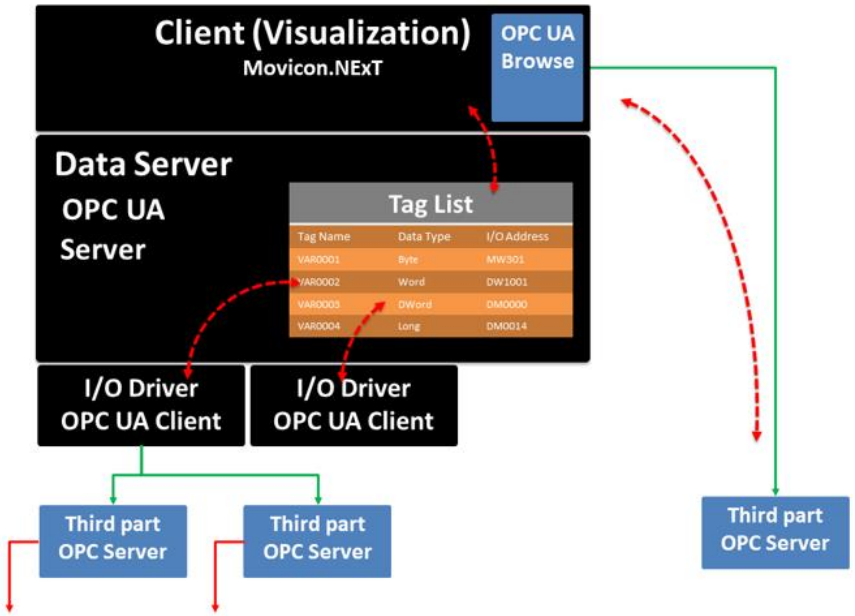
2.1.1. OPC UA Client

The Movicon.NExT client conforms to the OPC UA specifications and consents connectivity towards any other thirdy party OPC UA Server platform or device that supports these specifications:

- DA (Data Access)
- AC (Alarms and Conditions)
- HA (Historical Access)

Third party OPC UA Server connectivity

Client-Server connectivity according to the OPC UA Standard can occur on the Movicon.NExT platform's Client side and Server side as needed.



The integrity and confidentiality of data exchanged with third party OPC UA Servers secured by the exchanging of Application Certificates in the Application Authentication phase (Sign or SignAndEncrypt). Authentication by means of using X.509 Certificates is not supported at User Authentication level which continues to use the Anonymous and Username/Password modalities.

Client side connection using OPC UA Browser

In the visualization client it is possible to associate symbols and graphical objects on screen to Tags in the Server's Address Space as well as connect them directly to a third party OPC UA Server using the OPC UA Browser.
This modality lets you associate symbols or graphical objects to Tags which can be selected by browsing OPC UA Server items live.
In this way, the 'Client only' architecture can also be used by involving the platform's graphical part only: Movicon.NExT visualization client.

Server side connection using the OPC UA Client Driver

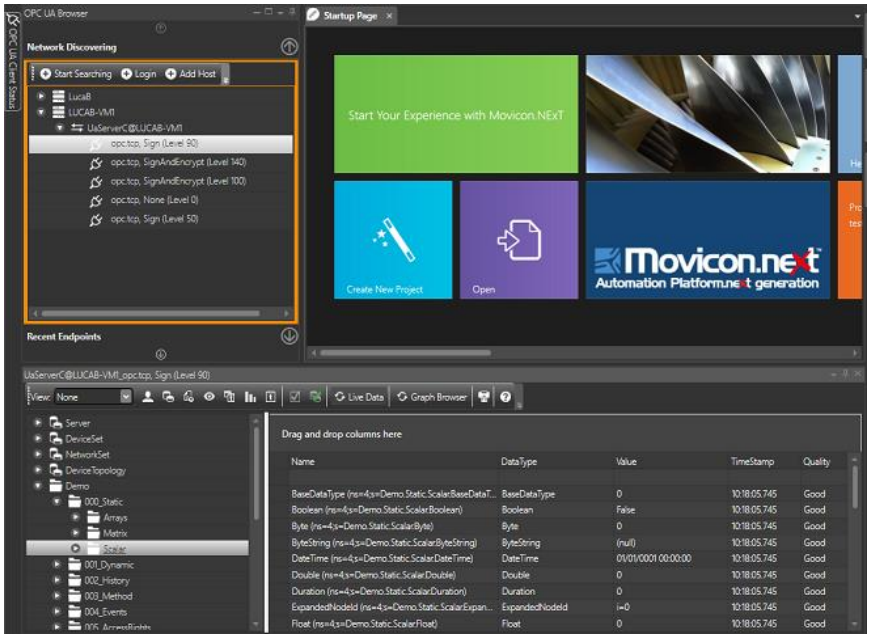
In order to make data deriving from other OPC UA Servers available to others by means of the Movicon.NExT Server, you will need to use the OPC UA Client communication driver provided for this purpose.

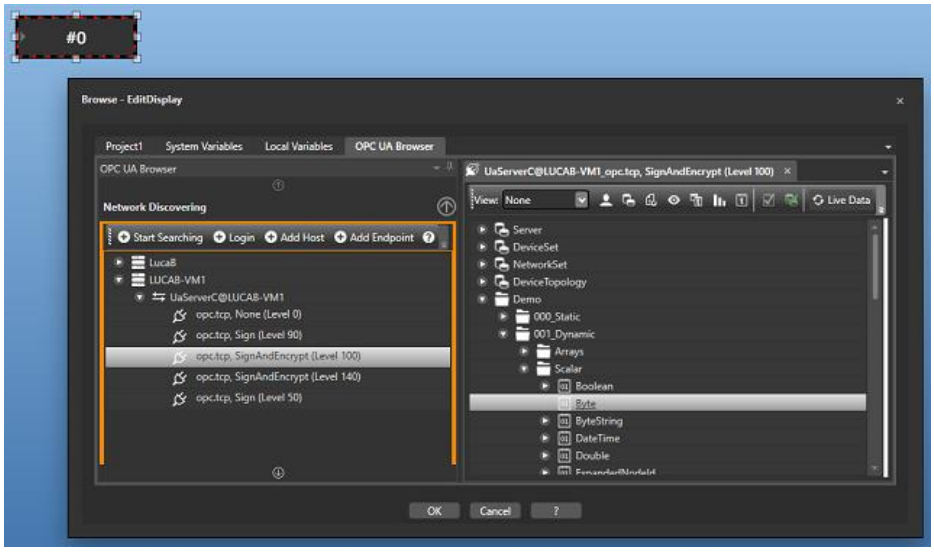
In this way, data will be available by means of Tags defined/imported in the platform's Address Space and therefore available to the entire system.

2.1.2. Client Connectivity using OPC UA Browser

It is possible to directly associate an item, that has been exposed by any compatible OPC UA Server, from any display object, symbol or command inserted on screen. In this way, the graphical element will be associated to the OPC UA Server item directly and not to a project Tag.

This is done through the Tag selection window by selecting the Tab corresponding to the OPC UA Browser.



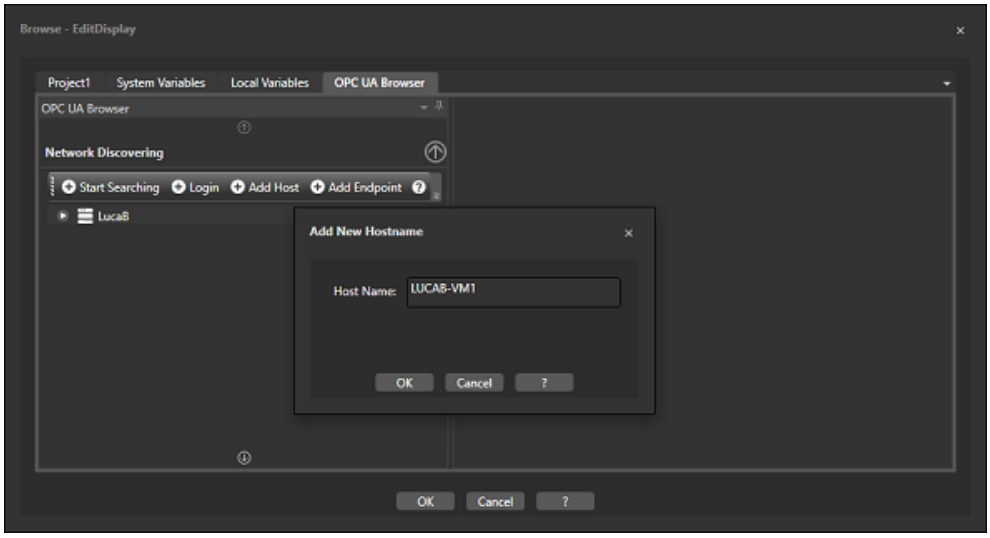


The OPC UA Browser is divided into two sections: the Endpoints selection window provided by the OPC UA Server is on the left and the Tag selection window relating to the established Secure Channel is on the right.
The above screenshot shows two Local Discovery Servers called "LUCAB" and "LUCAB-VM1". In the "LUCAB-VM1" Local Discovery Server an OPC UA server has been registered with the "UaServerC@LUCAB-VM1" Application Name and exposes five different Endpoint Sessions. Each Endpoint Session is identified by means of the triad: "Transport Protocol, Security Mode (Security Level)" for example "opc.tcp, SignAndEncrypt (Level 100)"
The following commands are available in the Endpoint selection window:

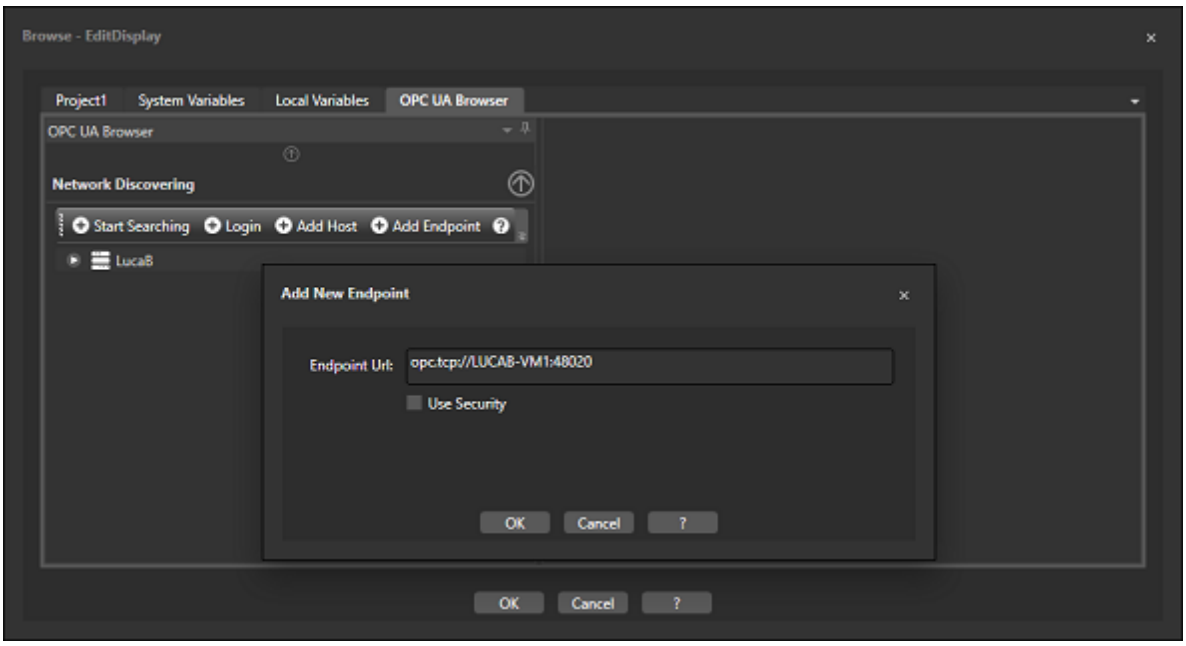
- Start Searching: scans the local network for any OPC UA Local Discovery Servers
- Login: permits User Access to the selected Endpoint by using User Credentials (username/Password) if not automatically requested during the connection phase.
- Add Host: permits Hostname to be defined manually (or IP address) of a Local Discovery Server from which to obtain all the Server's available Endpoint Sessions.
- Add Endpoint: permits you to define the url manually of the Endpoint you wish to connect to.

Endpoint connection

To end the procedure used to select an item from the OPC UA Server using the selection window, you will need to establish a Secure Channel by means of activating an Endpoint Session.
The Endpoint can be selected using one of three commands: "Start Searching", "Add Host" or "Add Endpoint".
A Host should be added if you do not know the Endpoints made available by the Server.



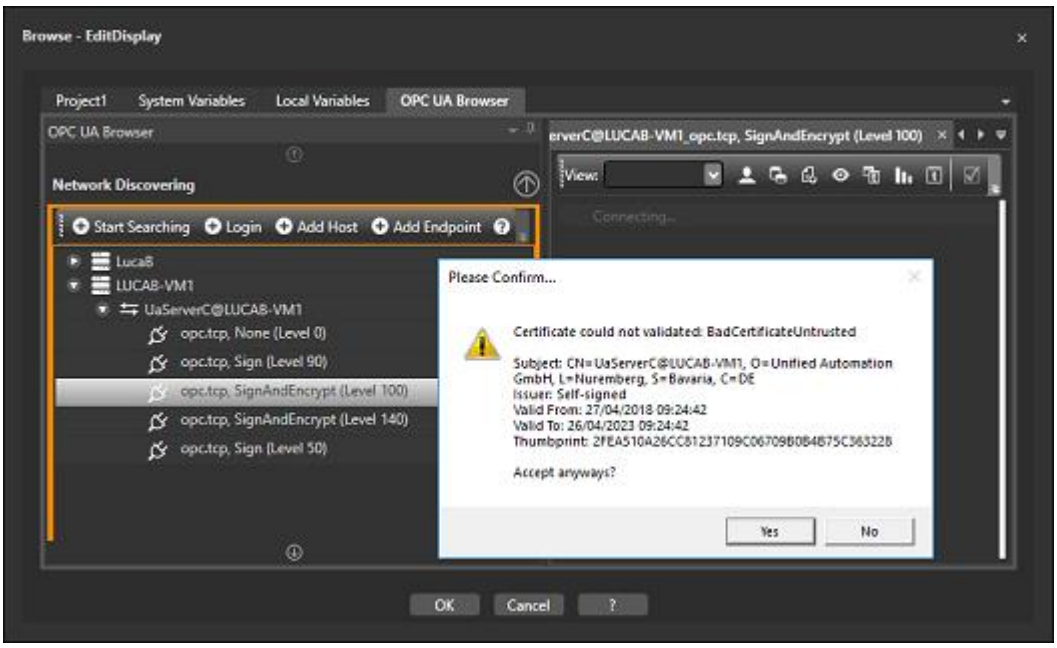
Otherwise, if you already know the Endpoint's url, you can select "Add Endpoint" to define the url using this format: "<Transport Protocol>://<hostname>:<port>", for example, "opc.tcp://server1:48020".



When you select the "Use Security" option, the Endpoint with a security level that is higher than those made available by the Server with a Sign or SignAndEncrypt Security Mode, will be selected.

It would be preferable to use a hostname instead of an IP address to manually define the Host and Endpoint especially in those cases where the selected Endpoint uses the Sign or SingAndEncrypt Security Modes. Creating a Secure Channel, in these cases, involves the exchanging of public certificates between the Client and Server. The necessity to use a hostname (or IP address) depends on how the OPC UA Server's security certificate to be connected to has been defined.

After having added an Endpoint or a Host that exposes several Endpoints, you will need to select one from the tree structure on the left with a double click to activate the communication channel towards the OPC UA Server.
If the security mode is Sign or SignAndEncrypt type, the Channel will open after the Application Instance Certificates have been exchanged. The Movicon.NExT OPC UA Browser receives and accepts the OPC UA Server certificate interactively by displaying a dialog window as shown in the screenshot below:



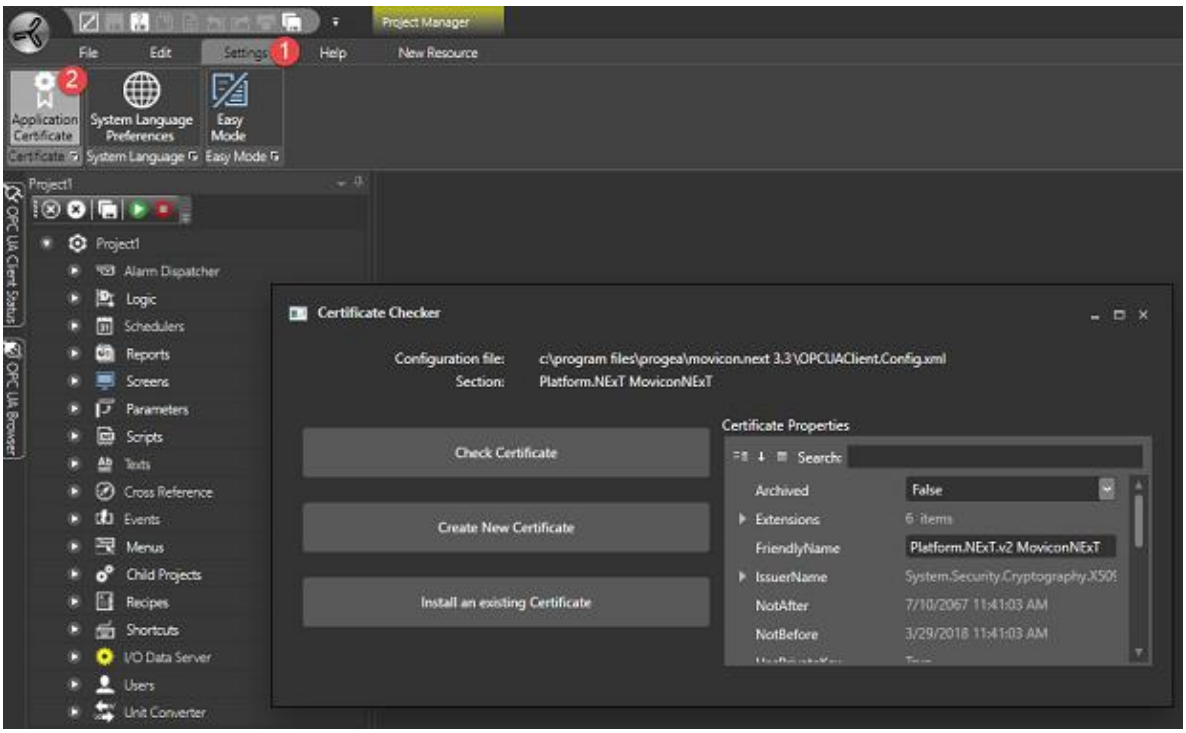
In this way the Movicon.NExT development environment, acting as Client, will accept the validity of the certificate for whole duration of the session.

The OPC UA Browser can be made to accept Untrusted certificates automatically by setting this option: <AutoAcceptUntrustedCertificates>true</AutoAcceptUntrustedCertificates> nel file "OPCUAClient.Config" present in the Movicon.NExT installation folder

In addition to receiving the certificate containing the Server's public key, Movicon.NExT also returns the certificate back to the same Server. However, according to the configuration, the OPC UA Server might refuse the certificate and place it in its list of rejected certificates. If this happens you will need to move the certificate from the list of Rejected Certificates and put it in the list of Trusted Certificates. Otherwise you can copy the Movicon.NExT cerficate file to the Server according to the relative procedures.
The certificate to be used in this case is the "Platform.NExT.v2 MoviconNExT" file which is located in the "%ProgramData%\OPC Foundation\CertificateStores\MachineDefault\certs\" directory

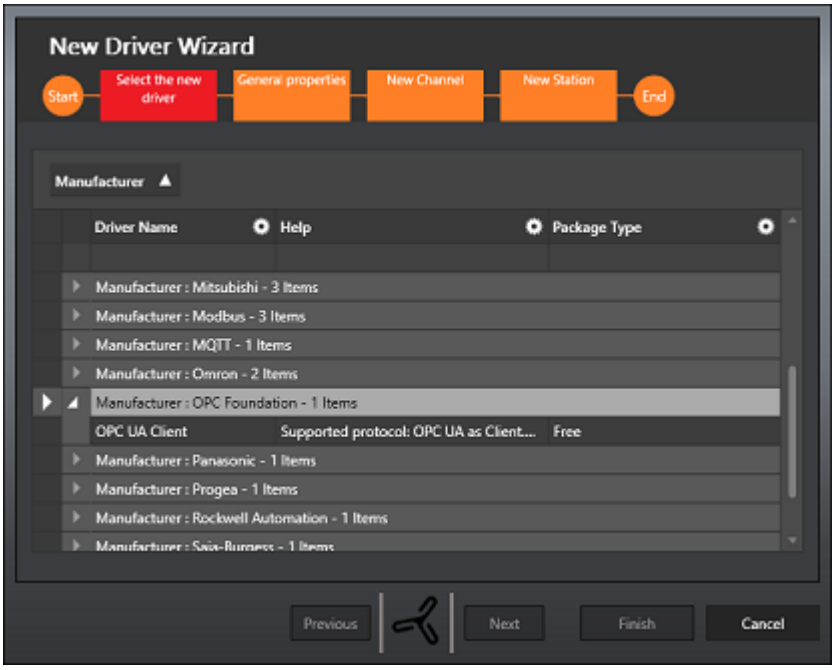
Certificate management

The Movicon.NExT certificate named "Platform.NExT.v2 MoviconNExT" can be controlled, removed or replaced using the configuration tool which can be accessed from the Ribbon's Setting (1) Tab and then selecting the Application Certificate (2) item.

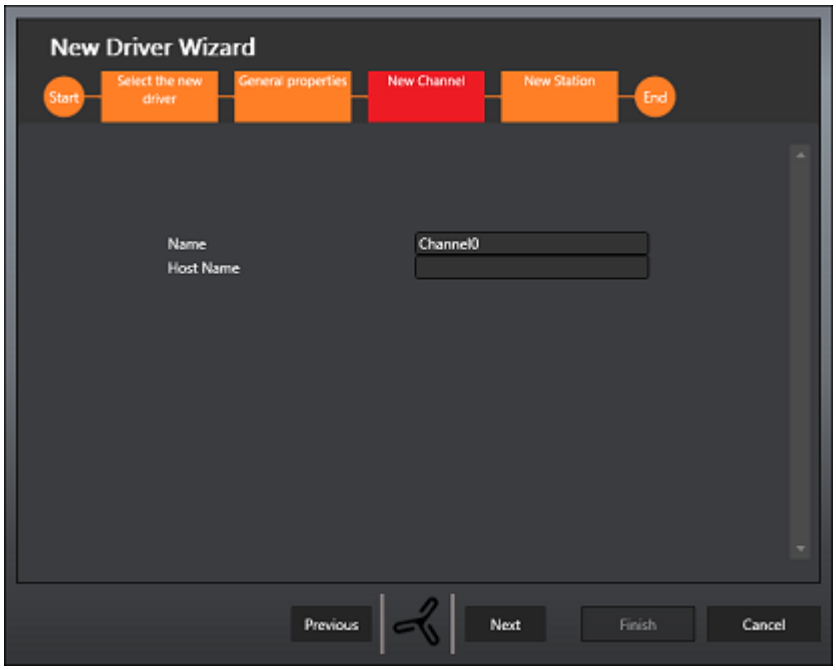


2.1.3. Connecting to Server with an OPC UA Client communication driver

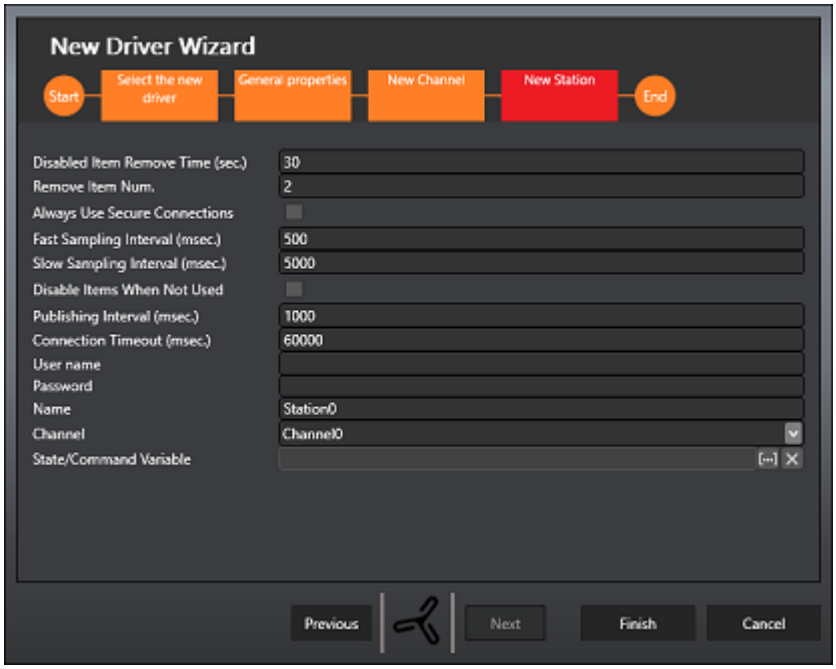
Connectivity towards an OPC UA server is usually accomplished by using an OPC UA Client communication driver.
In this way the Movicon.NExT Server becomes Client of a second OPC UA Server.
By means of using the Client OPC UA driver, the selected items will the available as Tags in the project's I/O Data Server's Address Space.



In addition to defining the name of the Channel, you must also define a Hostname to be used when connecting to the items. When tags are imported with the OPC UA Browser, this Hostname will be used to replace the one defined in the item's url.



The following parameters are available for defining the Station:



1. Disabled Item Remove Time (Sec.): this is used to set a delay time in seconds with which the non active OPC UA subscriptions will be removed.
- Remove Item Num.: this is used to set the number of non active OPC UA items to be removed at each time interval.
- Always Use Secure Connections: this is used to select the Endpoint with a Security Level higher than those made available by the Server with a Sign or SignAndEncrypt Security Mode type.

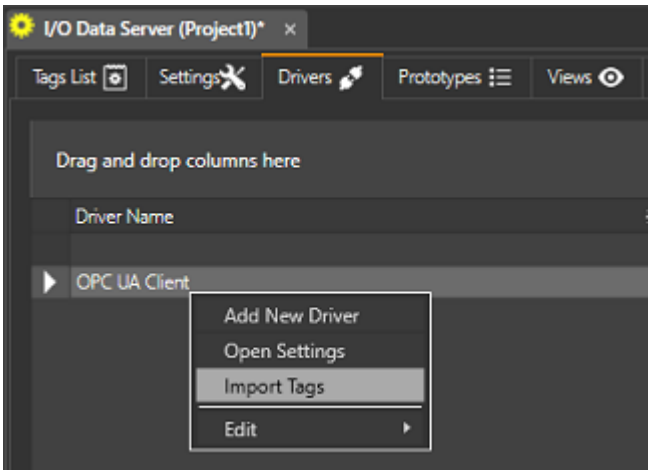
The certificate will need to be exchanged with public keys when using the Security Sign or SignAndEncrypt Security Modes. Therefore, the OPC UA Server must be provided with the certicate used by the OPC UA Client called "Platform.NExT.v2

IOServer", which is available in the "%ProgramData%\OPCFoundation\CertificateStores\MachineDefault\certs\" folder. While the OPC UA Server's ".der" certificate must be copied to the "%ProgramData%\OPC Foundation\CertificateStores\UA Applications\certs\" folder.

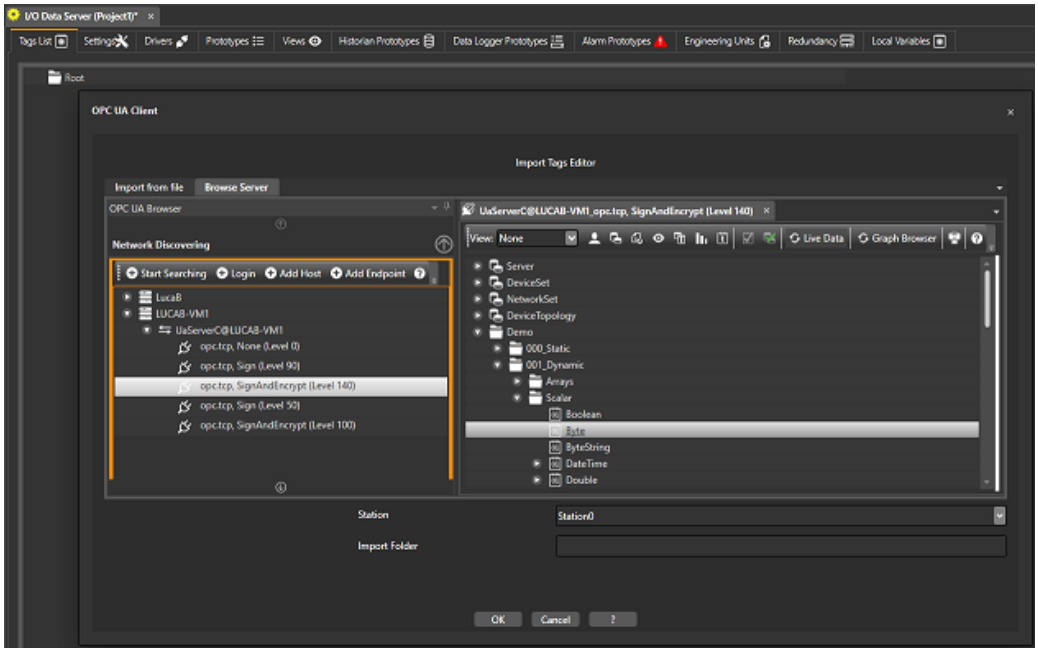
- Fast Sampling Interval: Defines the frequency with which existing and not-in-use tags are updated.
- Slow Sampling Interval: Defines the frequency with which existing tags that are going out-of-use are updated.
- Disable items When Not Used: Sets Tags as "Inactive" when not in use.
- Publishing Interval: Time in seconds of tag notifications towards Server.
- Connection Timeout: Connection timeout to Server.
- Username: Username to use for OPC UA Authentication at user level.
- Password: Password to be used for OPC UA Authentication at user level.
- Name: Name of Station.
- Channel: Channel that Station refers to.
- State/Command Variable: this property is used to assign a name to a supervision numeric variable (Byte type recommended) in order to control the communication status of the selected channel.

Bit 0 (State)	Connection Channel 0= connected 1= not connected
Bit 1 (State)	Primary Host Error State 0=Active 1=Inactive
Bit 2 (State)	Backup Host Error State 0=Active 1=Inactive
Bit 3 (State)	Connected Host 0=Active 1=Inactive

Once the OPC UA Client driver configuration has been defined, Tags can be imported from the OPC UA Server by selecting Import Tags from the contextual menu or from the Ribbon:



Tag import can be done using the "Import from file" or "Browse Server" which used the OPC UA Browser as seen previously.



In cases when the Client OPC UA driver is used for communicating with a OPC UA Server and Tags are imported in the project using the OPC UA Browser, it may be necessary to copy both the "Platform.NExT.v2 IOServer" (used by the driver) and "Platform.NExT.v2 MoviconNExT" (used by the OPC UA Browser) certificates to the OPC UA Server. Both these certificates can be found in "%ProgramData%\OPC Foundation\CertificateStores\MachineDefault\certs\"

Certificate management

The certificate used by the OPC UA Client driver is the same one used by the Movicon.NExT Server called "Platform.NExT.v2 IOServer". Please refer to the OPC UA Server topic's section entitled 'Certificate Management'.

2. OPC-UA Server

2.1.1. OPC UA Server

The Movicon.NExT Servers conforms to the OPC UA specification and consents connectivity to any other third+ party OPC UA Client platform or device supporting these specifications:

- DA (Data Access)
- AC (Alarms and Conditions)
- HA (Historical Access)

The OPC UA Server functionality is optional and needs to be enabled on your software product license.

Server Certification

Platform.NExT's OPC UA Server has been successfully certified by the OPC Foundation's OPC Certification Test Lab. The Test Lab carried out validation tests to verify full conformity with the standard (rel. 1.02) and Performance Stress and Load. Platform.NExT obtained Certification from the OPC Foundation to ensure users with guaranteed:

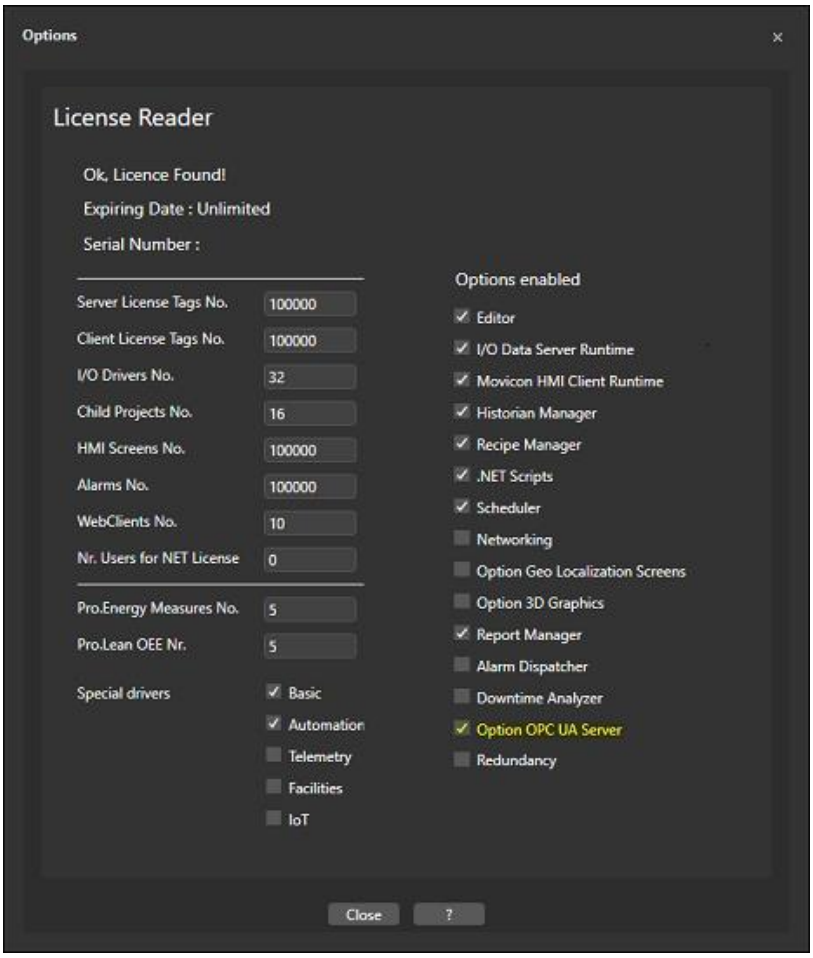
- Compatibility
- Interoperability
- Robustness
- Usability
- Efficiency

1. **Certificate Number: 1506CS006A**

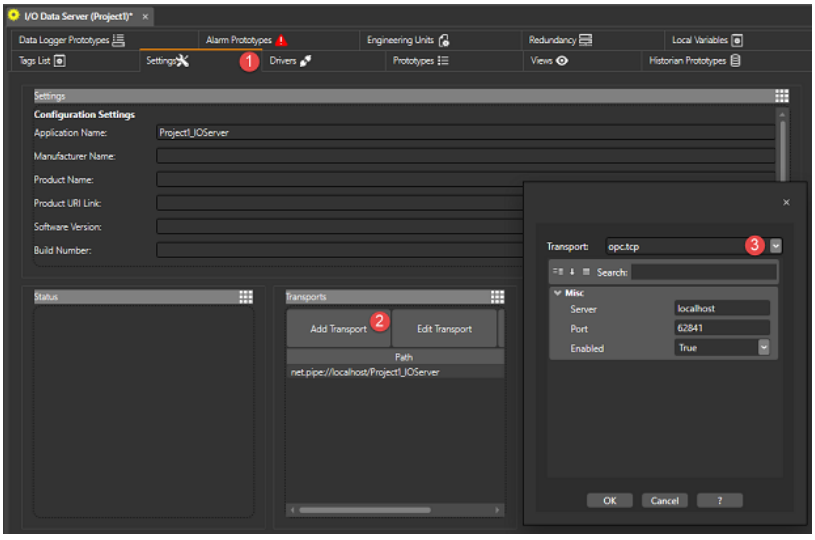
Third party OPC UA Client Access

In order for the Platform.NExT's I/O Data Server Tags to be visible and connected by means of using a third party OPC UA Client browser, the following points must be taken into consideration:

1. The OPC UA Server must be enabled on the Server's Runtime license



2. A network transport type must be configured in the I/O Data Server Settings that is adaptable for OPC UA communications such as tcp, http or https. As shown in the screenshot below, open the I/O Data Server Settings tab (1) and select Add Transport (2) in the transport definition area. In the network transport definition dialog field (3) select the transport desired and configure its properties. We recommend that you keep the 'localhost' as Server so that the hostname defined by the operating system is used.



The OPC Foundation's OPC UA Local Discovery Server (LDS) gets installed with the Movicon.NExT Setup. This Server provides the infrastructure

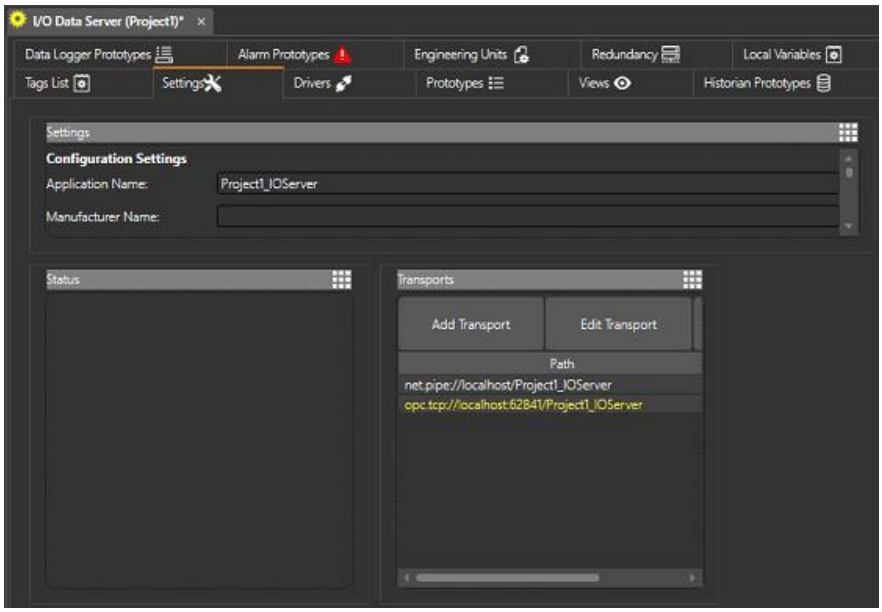
needed to expose OPC UA Servers; which have been started up in a machine, to make them visible to OPC UA Clients. If any problems with connecting or browsing between Client and Server occur, you should verify whether the OPC UA Local Discovery Server (opcualds.exe) has been started up and is working correctly (this process is found in the Windows Task Manager list headed UALDS).

Session Endpoint, Security Mode and Certificates

Movicon.NExT's I/O Data Server exposes three Endpoints with the following Application Authentication options:

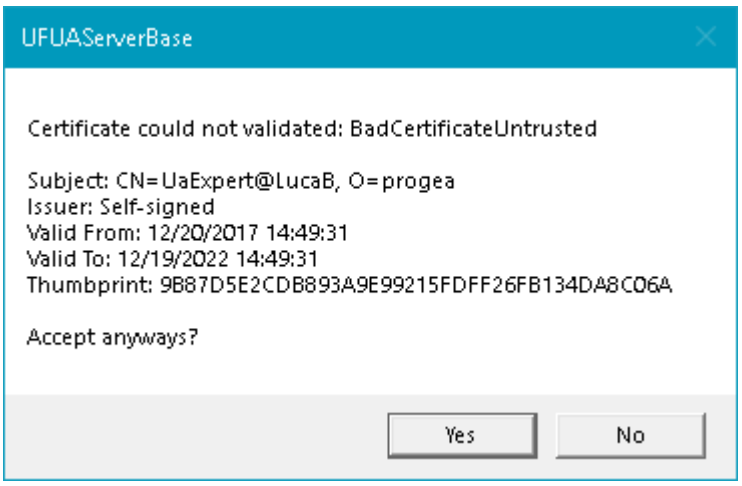
Security Mode	Security Policy	Security Level
None	-	Level0
Sign	Basic256	Level2
SignAndEncrypt	Basic128Rsa15	Level3

All authentication types defined by the OPC UA standard are supported for user authentication: Anonymous, Username/Password, X.509 User Certificates
These options can be changed by means of editing "UFUAServer.UAServer.Config" file which resides in the Movicon.NExT installation folder.
The Session Endpoint to use in the OPC UA Client to activate a Secure Channel with the Server is one of those defined in the list of Transports in the I/O Data Server Settings.



In reference to the previous screenshot, the session url to use would be:
opc.tcp://<hostname_del_server>:62841

When the Security Mode is Sign or SignAndEncrypt type, the Channel will open after the Application Instance Certificates have been exchanged. The Movicon.NExT Server automatically sends its OPC UA certificate to the OPC UA Client and receives and accepts the OPC UA Client certificate interactively by proposing a dialog as shown below:



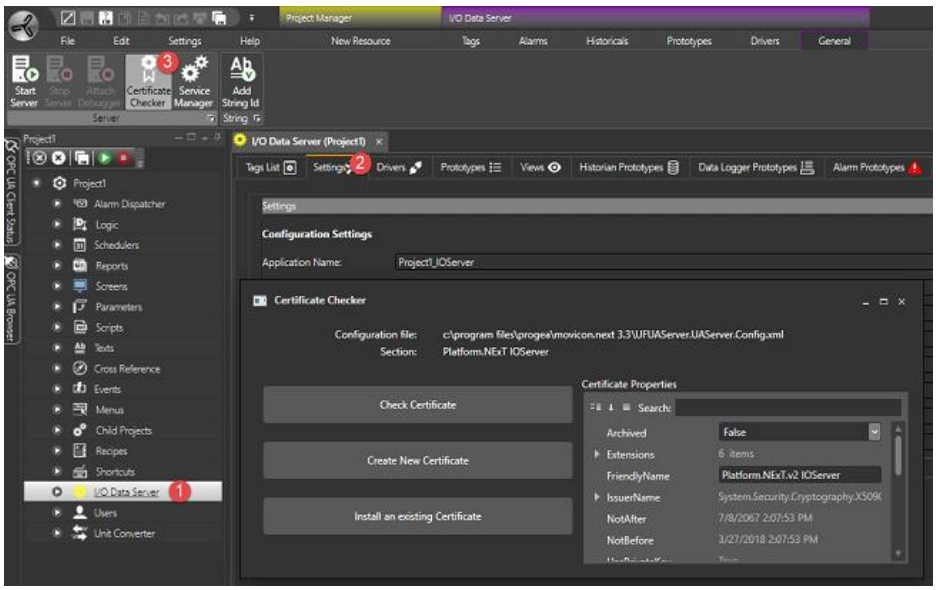
The Client certificate confirm dialog window is only available when the I/O Data Server has not been started up as service. However, it is possible to make the Server automatically accept Untrusted certificates by setting the <AutoAcceptUntrustedCertificates>true</AutoAcceptUntrustedCertificates> option in the "UFUAServer.UAServer.Config" file residing in the Movicon.NExT installation folder.

If you wish to make the OPC UA Client Certificate permanently Trusted on the Server, you will need to copy the Client's Certificate to the directory "%ProgramData%\OPC Foundation\CertificateStores\UA Applications\certs" directory.

If you need to provide the I/O Data Server's Application Instance Certificate to the OPC UA Client, you will find it in the file called "Platform.NExT.v2 IOServer" from the "%ProgramData%\OPC Foundation\CertificateStores\MachineDefault\certs\" folder

Certificate Management

The Movicon.NExT Server's certificate called "Platform.NExT.v2 IOServer" can be controlled, renewed or replaced using the configuration tool which is accessed by expanding the I/O Data Server (1) and selecting the Settings tab (2) and then selecting the Certificate Checker (3).


Project Manager
I/O Data Server
File
Edit
Settings
Help
New Resource
Tags
Alarms
Historicals
Prototypes
Drivers
General
Start Server
Stop Server
Attach Debugger
Certificate Checker
Service Manager
Add String Id
Server
String
3
Project1
I/O Data Server (Project1)
Tags List
Settings
2
Drivers
Prototypes
Views
Historian Prototypes
Data Logger Prototypes
Alarm Prototypes
Alarm Dispatcher
Logic
Schedulers
Reports
Screens
Parameters
Scripts
Texts
Cross Reference
Events
Menus
Child Projects
Recipes
Shortcuts
I/O Data Server
1
Users
Unit Converter
Configuration Settings
Application Name:
Project1_IOServer
Certificate Checker
Configuration file:
c:\program files\progea\movicon.next 3.3\UFUAServer.UAServer.Config.xml
Section:
Platform.NExT IOServer
Check Certificate
Create New Certificate
Install an existing Certificate
Certificate Properties
Search:
Archived
False
Extensions
6 items
FriendlyName
Platform.NExT.v2 IOServer
IssuerName
NotAfter
7/8/2067 2:07:53 PM
NotBefore
3/27/2018 2:07:53 PM

INDUSTRIAL AUTOMATION SOFTWARE

progea

Progea Srl
Via D'Annunzio, 295
I-41123 Modena
info@progea.com
Tel +39 059 451060

Progea International SA
via Sottobisio, 28
6828 Balerna (CH)
international@progea.com
Tel +41 91 96 76 610

Progea Deutschland GmbH
Marie-Curie Str., 12
D-78048 VS Villingen
info@progea.de
Tel +49 (0)7721 99838 0

Progea North America Corp.
2380 State Road 44, Suite C
Oshkosh, WI 54904
info@progea.us
Tel. +1 (888) 305-2999