Matrikon®
Powering Interoperability.

Matrikon® Data Broker

Version 2.0

End User How-To Guide

LIMITATIONS

Matrikon has made its best effort to prepare this manual. Matrikon makes no representation or warranties of any kind with regard to the completeness or accuracy of the contents herein and accepts no liability of any kind including without limitation warranties of merchantable quality, satisfactory quality, merchantability, and fitness for a particular purpose on those arising by law, statute, usage of trade, course of dealing or otherwise. Matrikon shall not be liable for any losses or damages of any kind caused or alleged to be caused directly or indirectly from this manual.

LICENSE AGREEMENT

This document and the software described in this document are supplied under a license agreement and may only be used in accordance with the terms of that agreement. MatrikonOPC reserves the right to make any improvements and/or changes to product specifications at any time without notice.

About this manual

SOFTWARE VERSION

Version: 2.0

DOCUMENT VERSION

Version: 1

Document revision history

Revision	Supported Release	Date	Description
1.0	2.0	17/06/2022	Initial Release

Scope

This How-to Guide focuses on specific end-user tasks for the Matrikon Data Broker (**MDB**) and provides illustrated instructions for carrying them out. These end-user tasks are the basic tasks associated with the features of MDB, and do not include external tasks required to install or configure additional components such as bridging applications, network traversal applications, or external information models. This document describes only those specific fields and functions required to carry out the described task.

MDB is a complex application that implements many of the functionalities defined in the OPC UA specification. Some of these require advanced knowledge of the UA specifications and the components described therein. For example, some tasks associated with information modeling and data mapping require knowledge of programming methods in addition to knowledge of the address space components described in the specification. They are therefore beyond the scope of this document.

This document is intended as a supplement to the User Manual rather than a replacement for it. For a full description of the functionality of MDB please refer to the User Manual

Related documents and resources

MDB User Manual

Intended audience

This document is intended primarily for MDB end-users, administrators, and anyone responsible for day-to-day use of MDB and the operation of the system in which it is installed.

It can also be used as a reference for any customer considering the purchase of MDB.

Admonishments

NOTE - This symbol indicates additional information that should be considered when using this product or feature of the product.

INFO - This symbol indicates additional information about the product or feature of the product that may be of interest to the user.

CAUTION – This symbol indicates that an incorrect action may result in an error or unintended or unwanted operation of the product or a feature of the product.

WARNING – This symbol indicates that care must be taken to avoid an incorrect action that may result serious system errors which can include catastrophic failure!

Contents

About this manual ..... **i**

Document revision history ..... **i**

Scope ..... **ii**

Related documents and resources ..... **ii**

Intended audience ..... **ii**

Admonishments ..... **iii**

Introduction **1**

Connecting Matrikon OPC UA Explorer to MDB 2

Changing the user password in MDB 5

Connecting third-party clients to MDB 7

Connecting MDB to OPC UA servers 11

Accepting application instance certificates in MDB 15

Creating reverse connections across firewalls 17

Importing information models into MDB 23

Creating instances of a type 28

Mapping data in MDB 32

Installing an MDB container on a Linux platform 38

Publishing data from MDB to the cloud 41

Introduction

The Matrikon Data Broker (MDB) is a fully compliant OPC Unified Architecture (UA) application that supports both server and client functionality. Server aggregation, reverse connection, data modeling and mapping, and MQTT Publisher are all supported in MDB 2.0. These features include many configuration and control items required for proper operation. This requires logging into the MDB as an administrator (*admin*).

OPC UA Explorer is a fully compliant OPC UA client application which supports all the features implemented in MDB. UA Explorer is also designed as the configuration interface for MDB. All tasks involving the configuration of MDB features and functions are carried out through UA Explorer.

This being the case, the first task that must be carried out is to connect UA Explorer to MDB. Upon first login, the password for the admin account should be changed. These are the first two tasks described in this guide.

Connecting Matrikon OPC UA Explorer to MDB

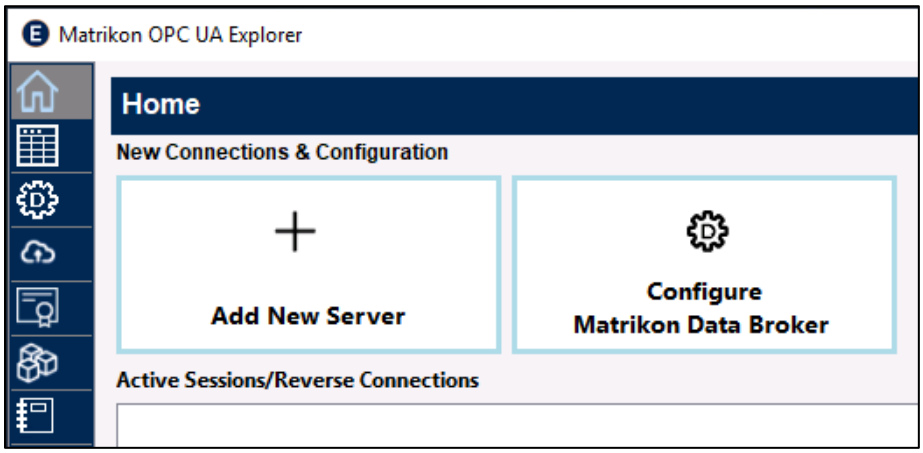
Use Case

The Matrikon OPC UA Explorer is a fully compliant OPC UA client application that is designed to act as the configuration utility for MDB. It can be used to connect to any OPC UA server application. In the context of this document UA Explorer provides access to all the features of MDB.

Procedure

1. Open OPC UA Explorer.

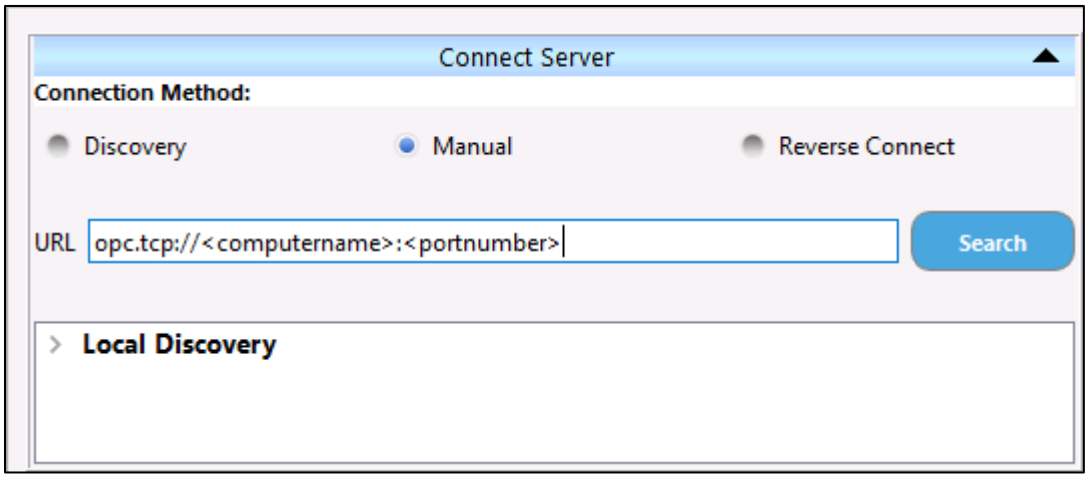
2. On the Home page, click on the Add New Server tile.



NOTE – By default UA Explorer uses discovery to retrieve the server url. If the Local Discovery server (LDS) is not available UA Explorer will throw an error and no server is listed. If the LDS is available, the MDB endpoints are listed under Local Discovery. This procedure assumes that no LDS is available.

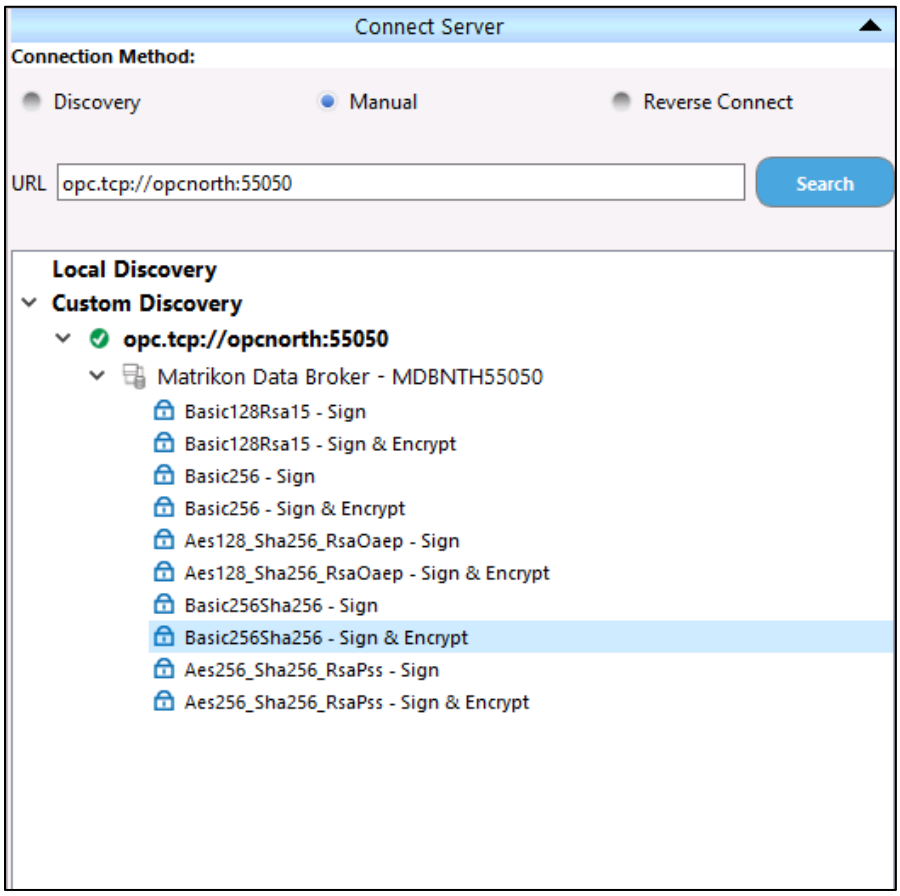
3. Select the Manual radio button in the Connect Server window and enter the url of the MDB instance using the syntax:

opc.tcp://<computername>:<portnumber>

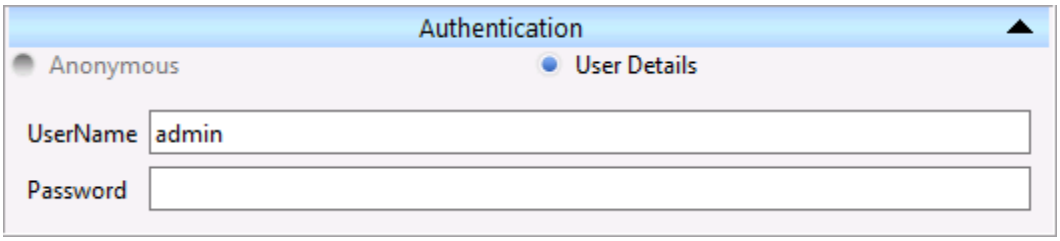


4. Click on the Search button to retrieve the endpoints for this server.

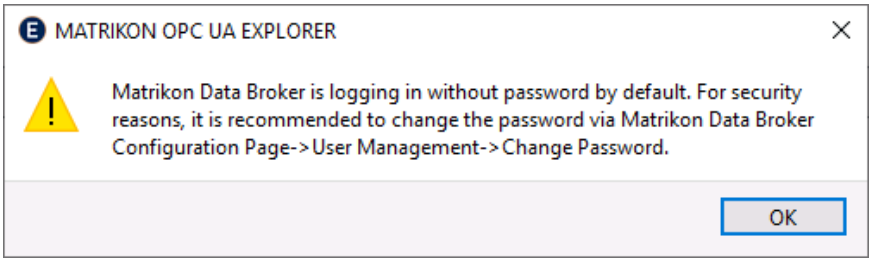
5. Expand the list of endpoints under the MDB instance and select the required connection endpoint.



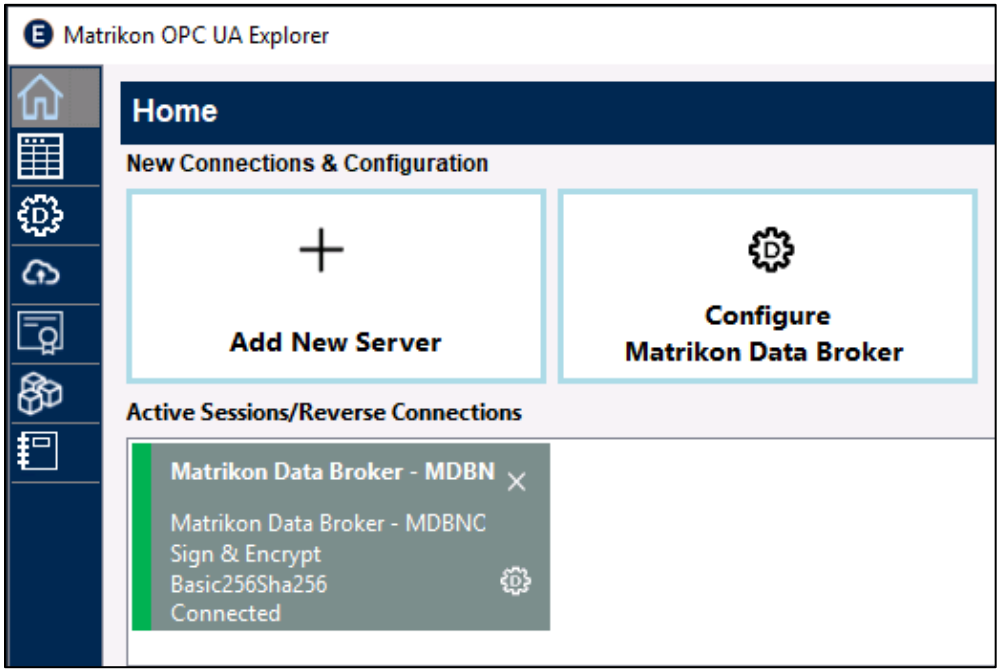
6. In the Authentication window, select the User Details radio button and fill in the required UserName and Password.



   a. For the first connection to a new installation of MDB, the default login is the *admin* account with a blank password. This will result in UA Explorer displaying a warning. For information on changing passwords in MDB, refer to the section **How do I change the user password in MDB?**



7. UA Explorer displays an active connection tile in the Active Sessions/Reverse Connections list.



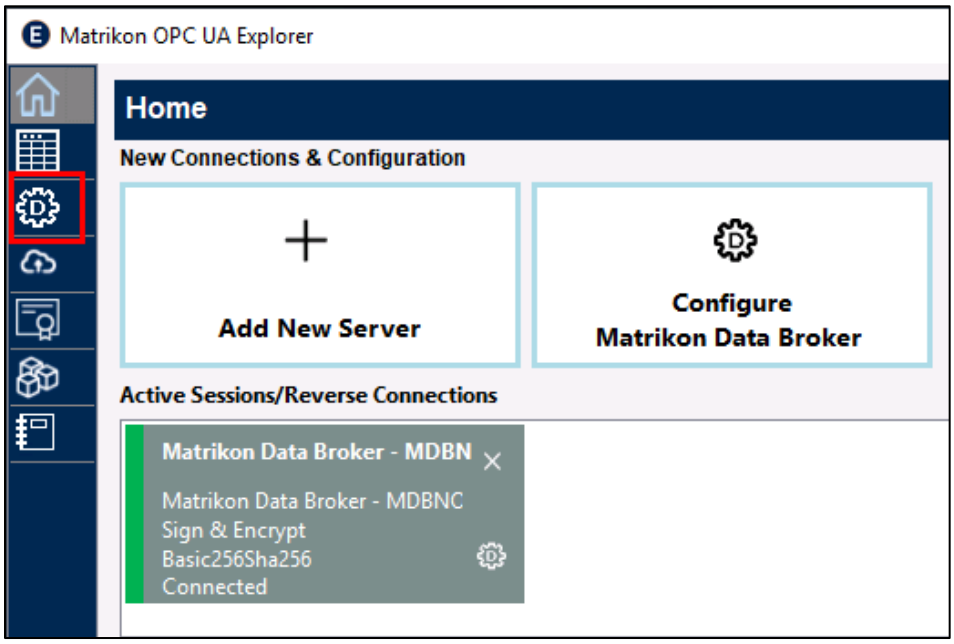
Changing the user password in MDB

Use Case

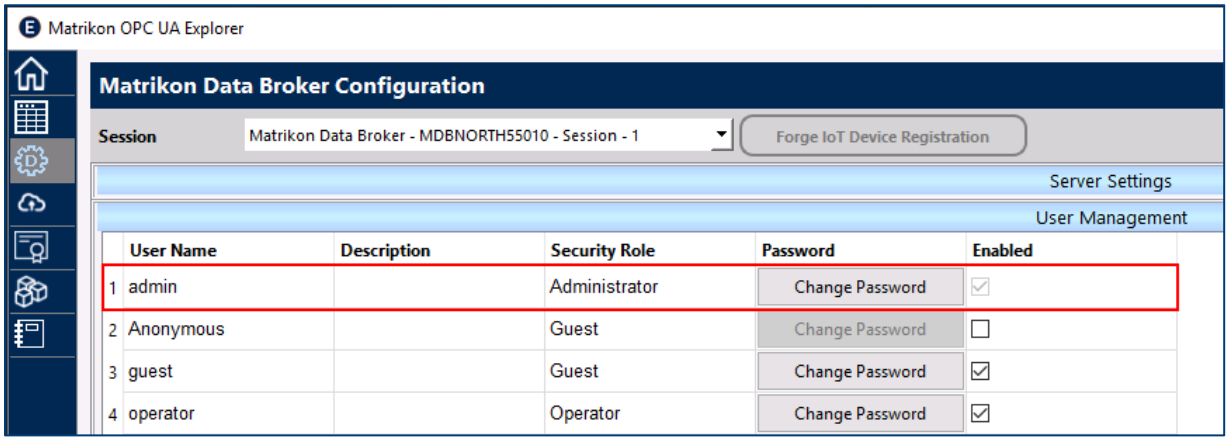
MDB comes preconfigured with no passwords for any of the users. When logging in for the first time, users receive a message recommending changing the password. To accomplish this task, it is necessary to access the User Management tab in the MDB configuration tab.

Procedure

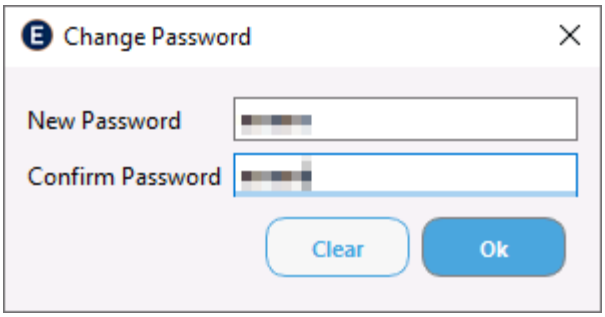
1. Connect UA Explorer to MDB.
2. Click on the Configure MDB icon on the left edge of the display.



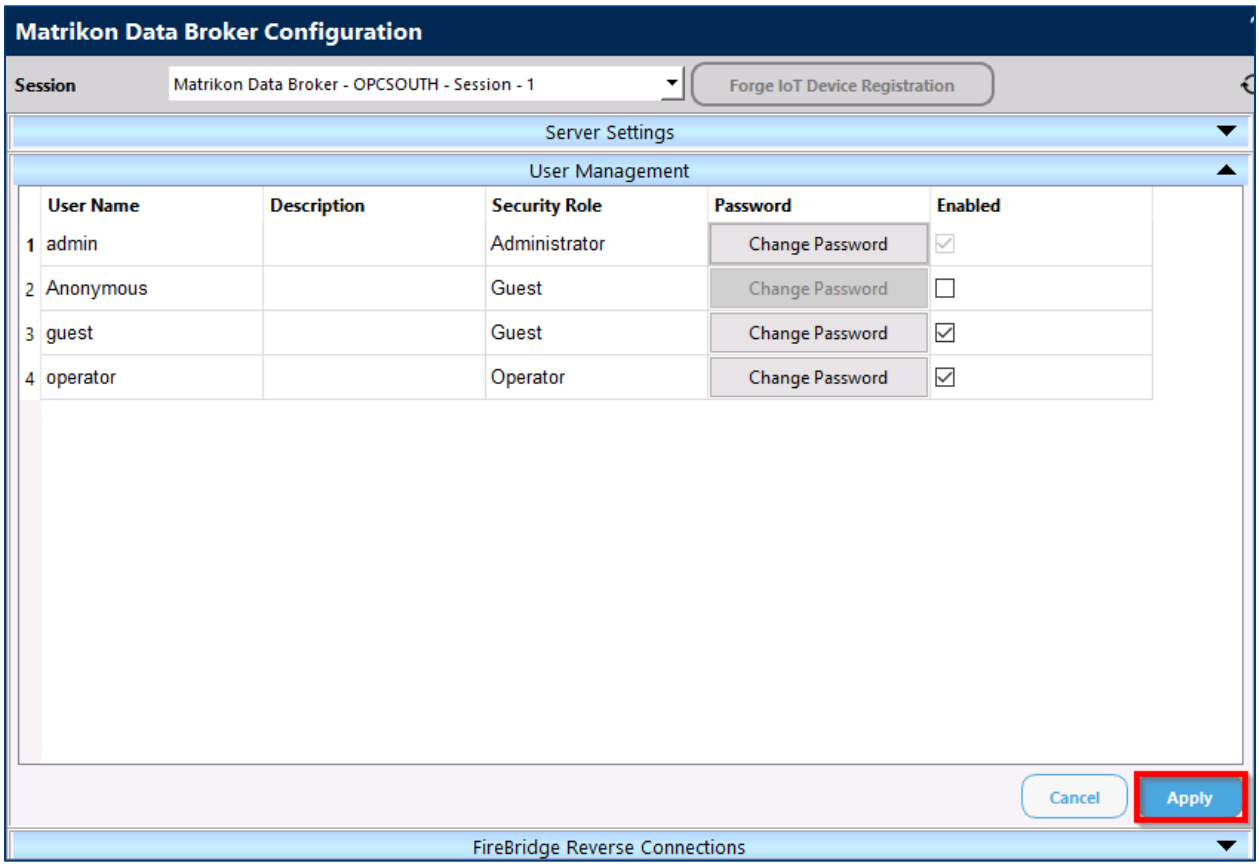
3. In the User Management tab, click on the Change Password button for the account you wish to modify. In this example we are changing the *admin* account password.



4. Enter and confirm the new password in the Change Password dialogue. Click on OK.



5. Click on the Apply button to execute the requested change.



6. In UA Explorer, disconnect from MDB, then reconnect using the new password.

Connecting third-party clients to MDB

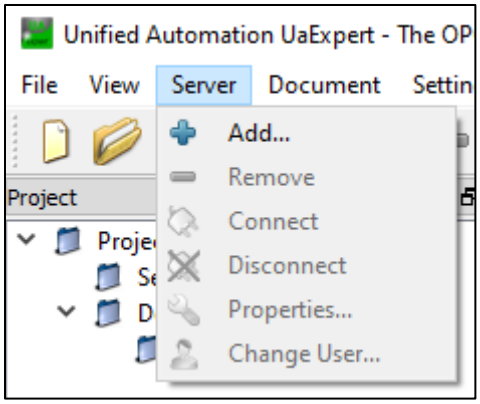
Use Case

Although Matrikon's UA Explorer is the client application that is installed with MDB and acts as the configuration tool, it is not intended as an industrial solution. The client application that normally connects to MDB is usually the application that consumes the data, such as a DCS or an archive application.

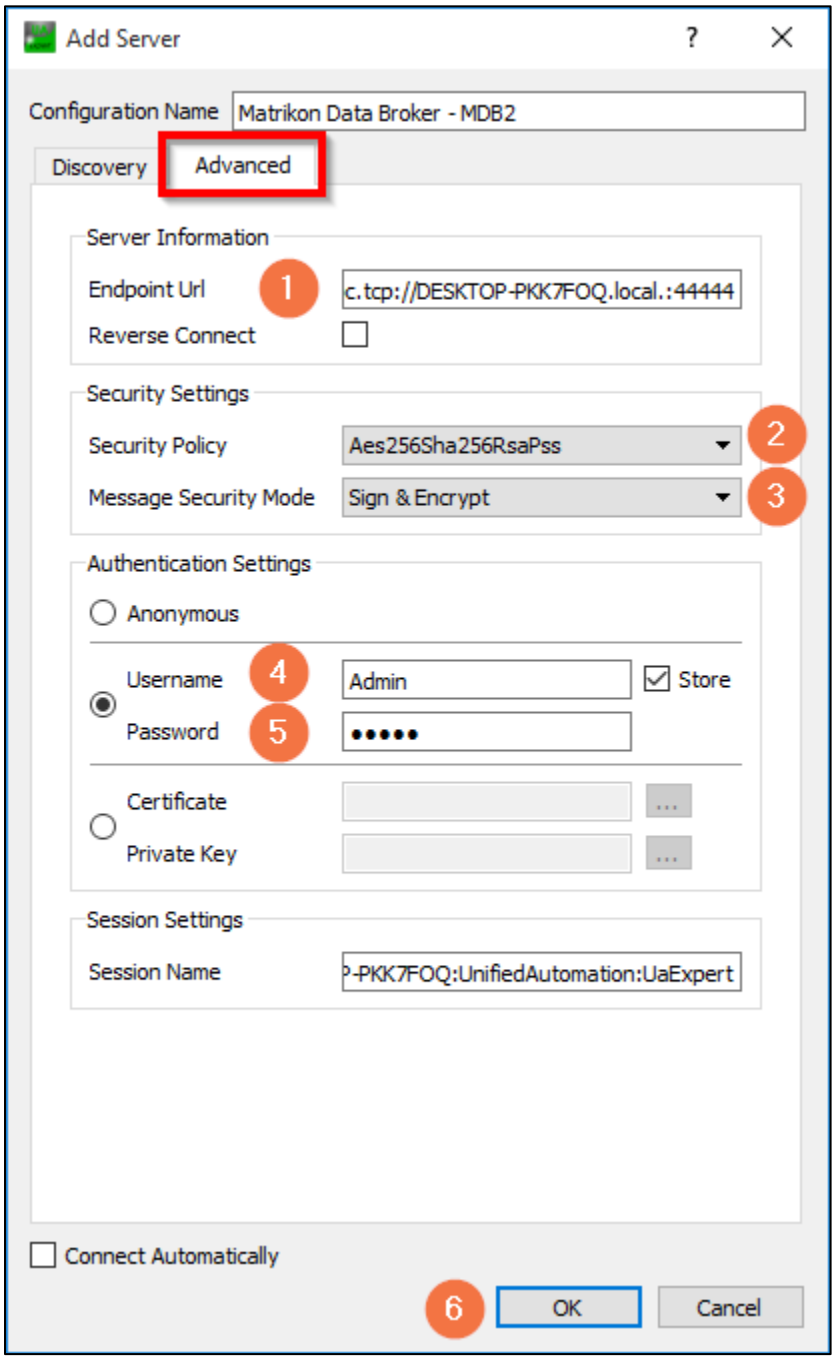
Procedure

Any client application connecting to MDB must execute the same process as UA Explorer, ensuring that each required field is correctly filled in. The specific actions and GUI items will vary from client to client, but the requirements remain the same. In this example we use the Unified Automation UA Expert as our client.

1. Open UaExpert.
2. Click on the Add item from the Server dropdown menu

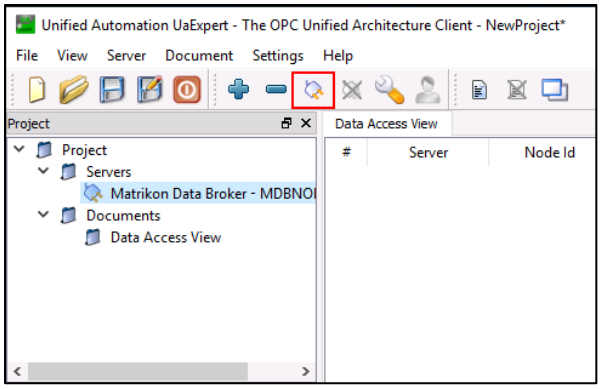


3. Fill out the connection details under the “Advanced” tab within the “Add Server” window:

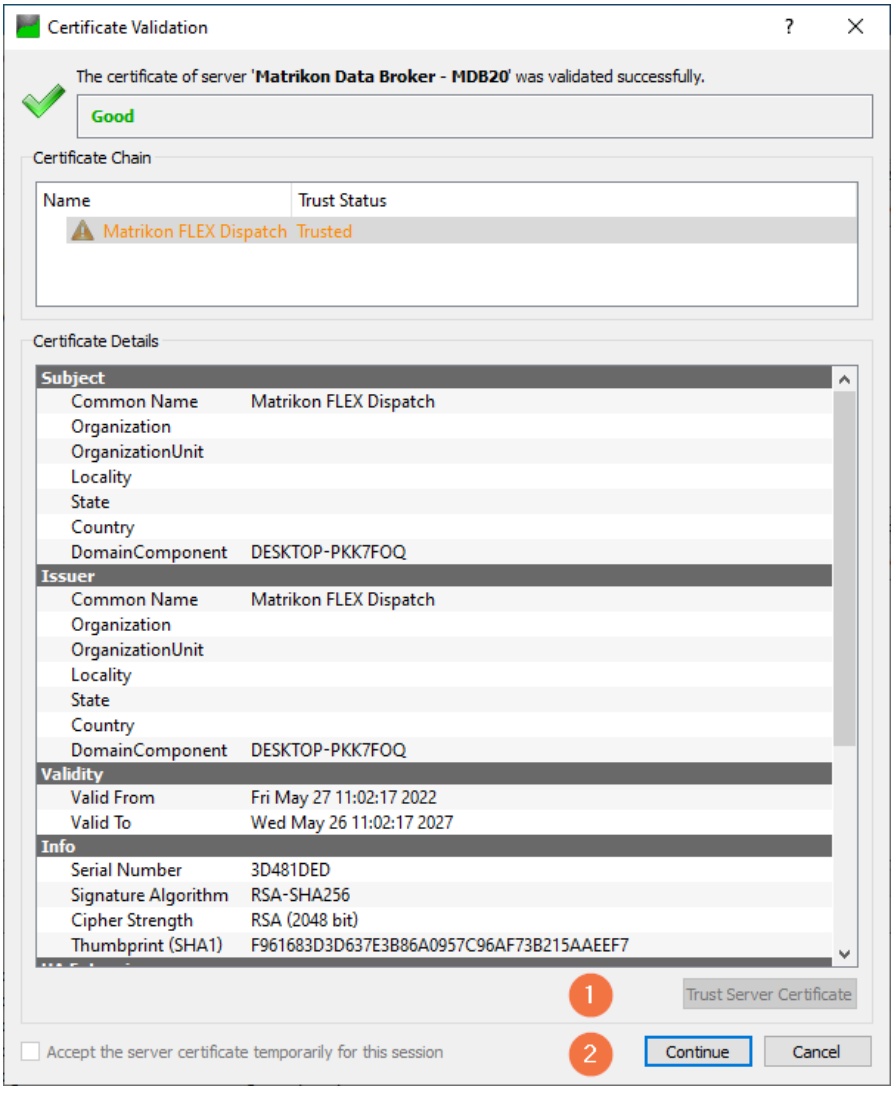


1) Endpoint URL
2) Security Policy
3) Message Security Mode
4) User Name
5) Password
6) Click on OK.

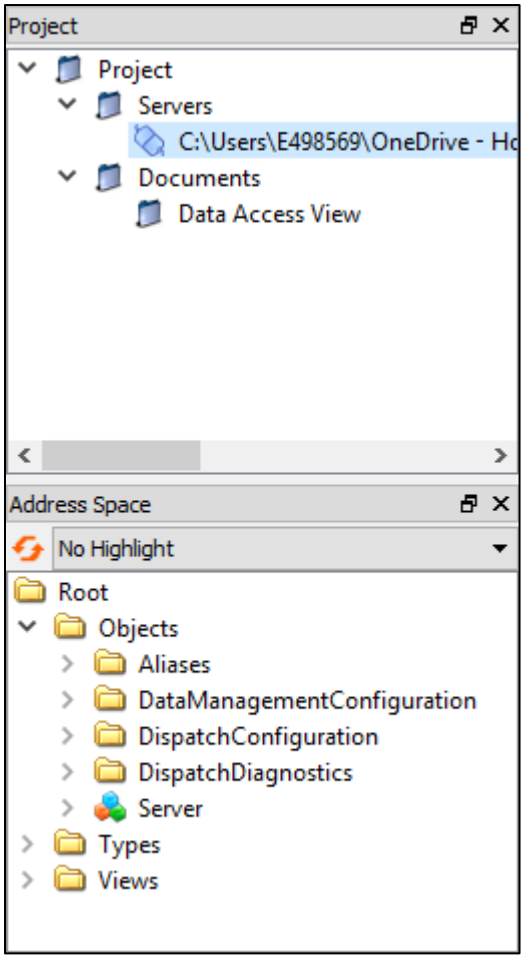
4. There are a number of commands for connecting to this new server, including right-clicking the server and selecting “Connect”, or as shown here, clicking on the “connect button in the button bar.



5. If this is the first attempt to connect to the MDB, a “Bad Security” error may occur due to untrusted certificates. If this happens, the Certificate Validation dialog opens automatically.



6. To accept the MDBs certificate with UA Expert
   1) Click Trust Server Certificate
   2) Click Continue
7. To accept the UA Expert certificate with MDB, refer to the section “How do I accept application instance certificates in MDB?”
8. The “Address Space” window now displays the MDB address space.



Connecting MDB to OPC UA servers

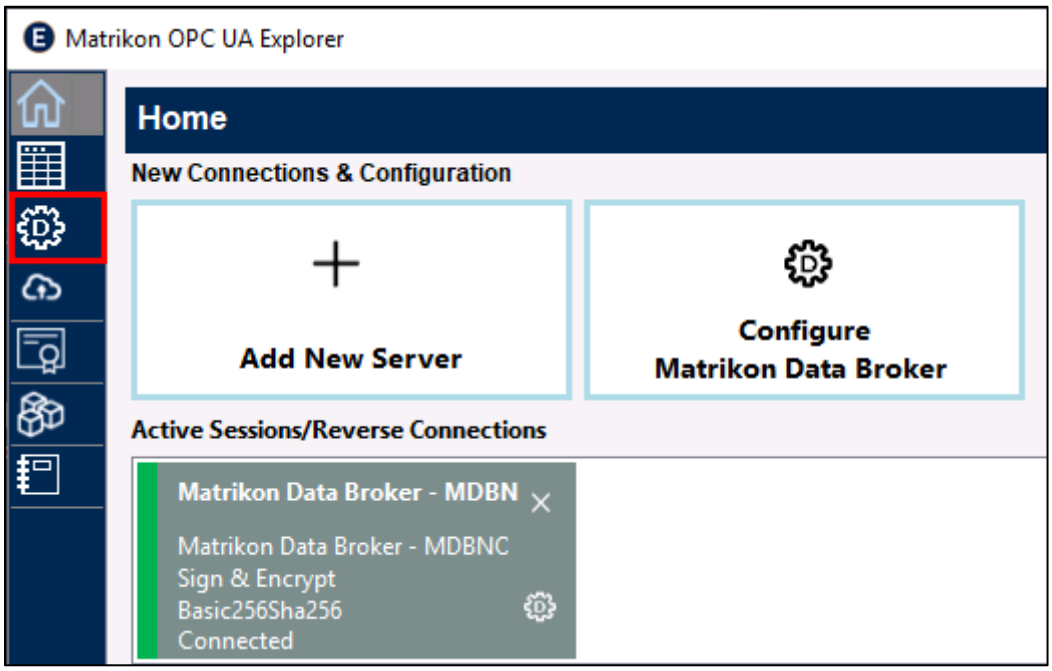
Use Case

OPC UA client and server functionalities are more easily integrated into a single application than was the case in OPC Classic. MDB is specifically designed and developed with this integration in mind. The Federator feature allows MDB to connect to other OPC sources such as OPC UA servers. MDB thus becomes the gathering point; a single application where all data from an extended system can be represented within a single address space.

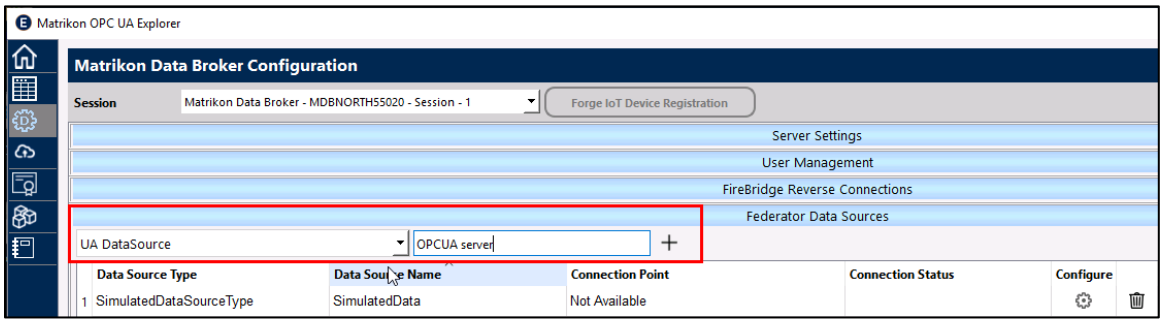
Procedure

NOTE - Errors may occur in this procedure if the target OPC UA server has not accepted the MDB application instance certificate. Before starting, ensure the MDB certificate has been added to the UA server's trusted certificate list, and is not in the rejected certificate list.

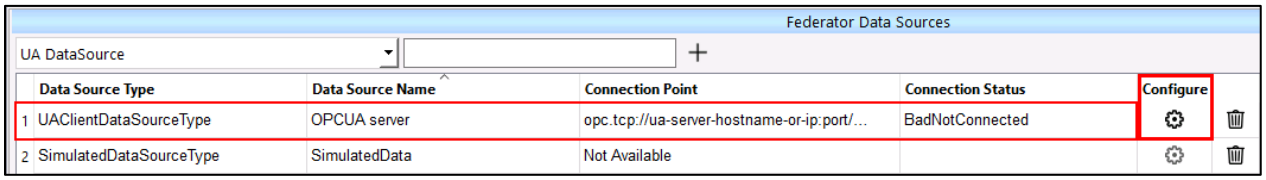
1. Connect UA Explorer to MDB using the *admin* account.
2. Click the Data Broker Configuration button.



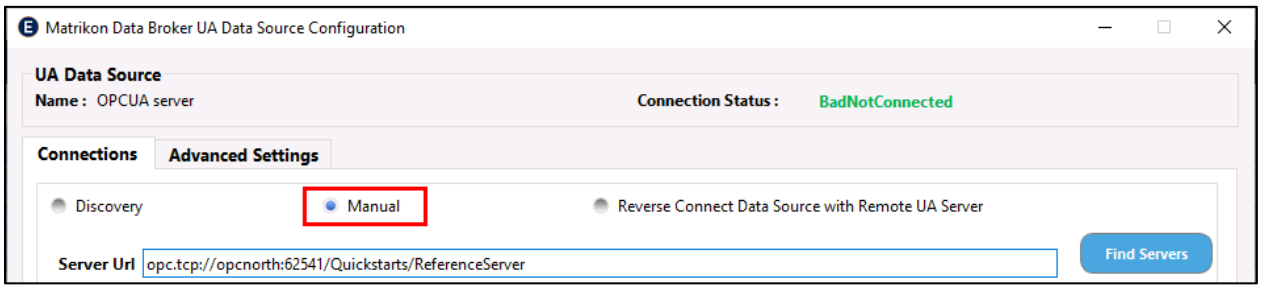
3. Expand the Federator Data Sources panel.



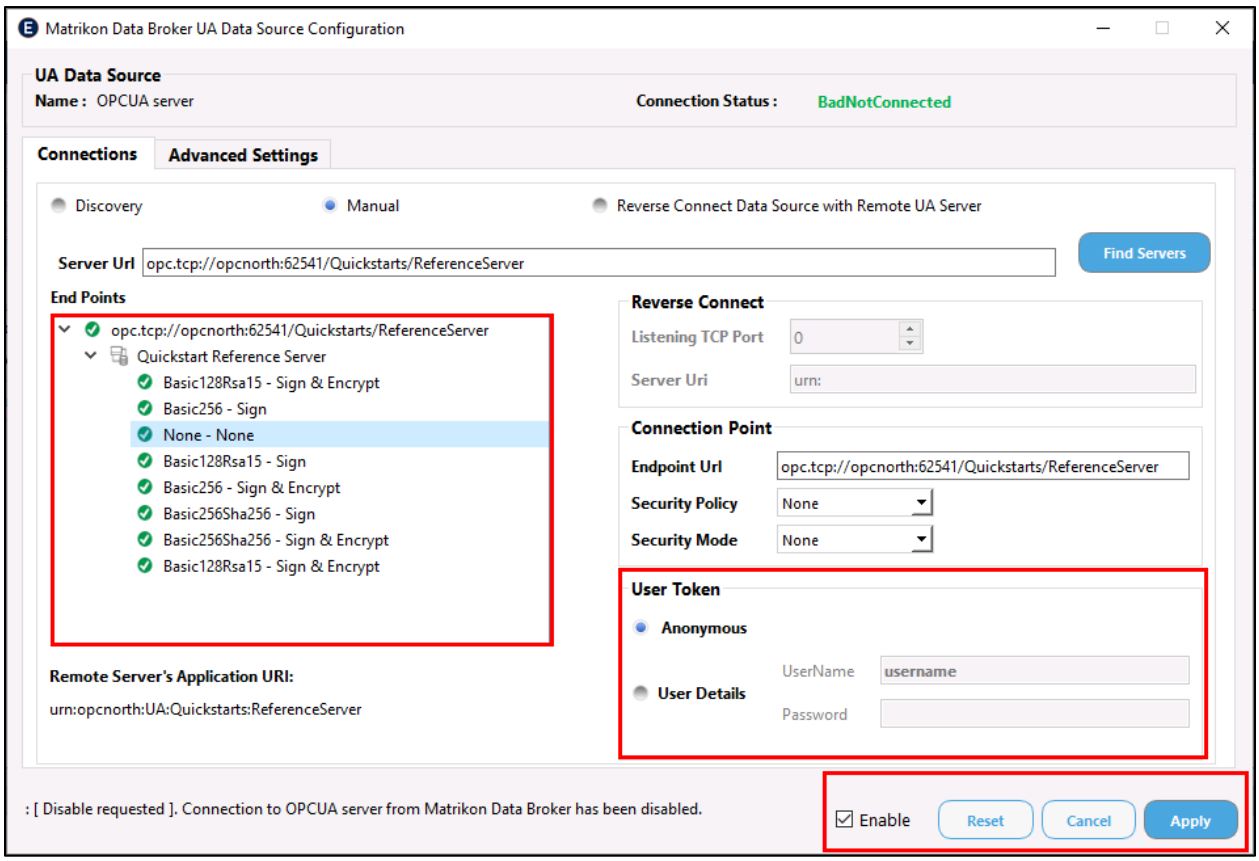
4. Select UA DataSource from the drop-down menu and enter a name for the connection.
5. Click the '**+**' to add the connection. The new connection appears in the list of Data Sources
6. Click on the Configure button for the new connection.



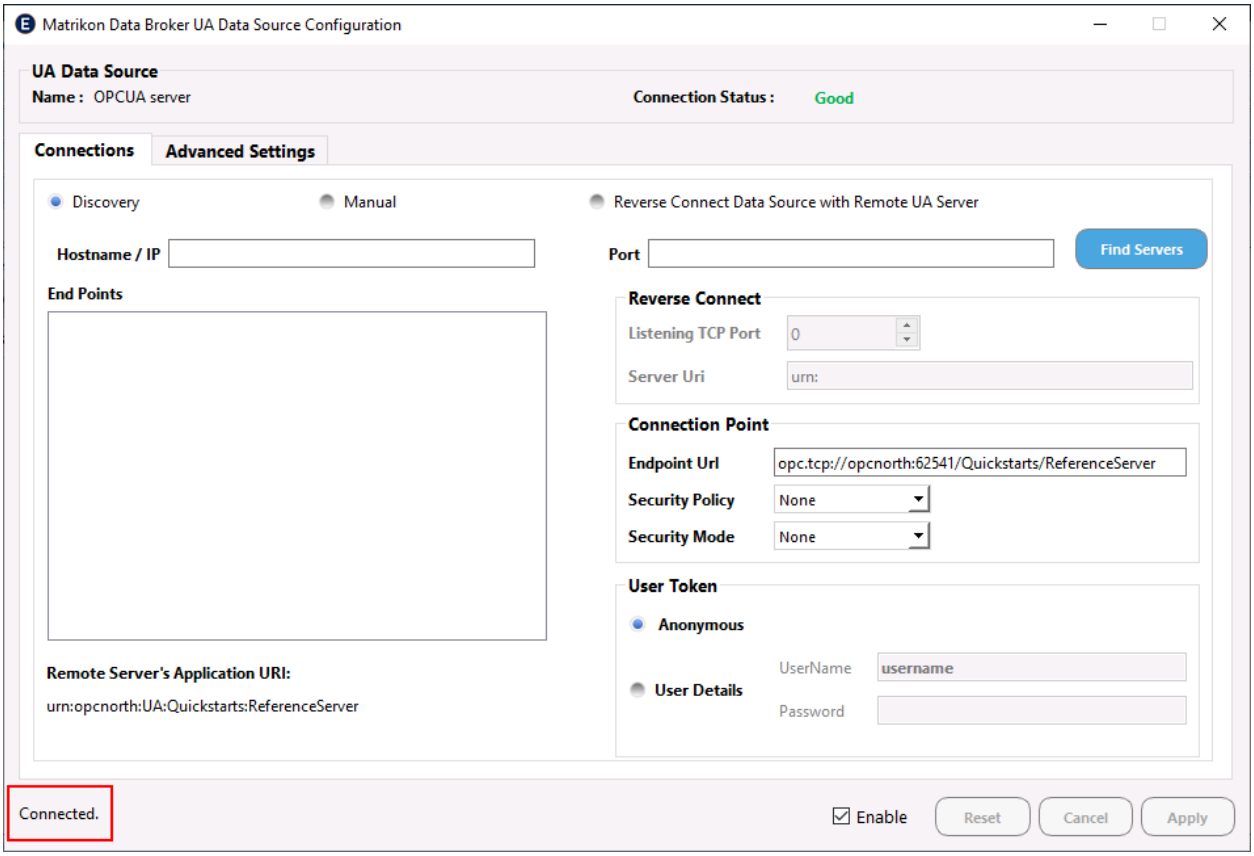
7. Select the Manual radio button to connect directly to the data source.



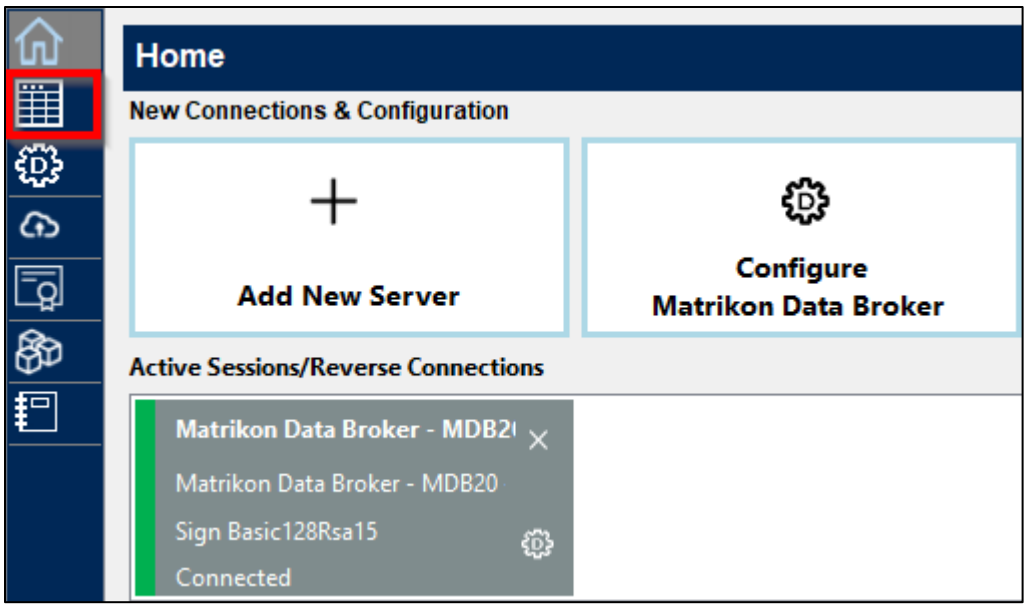
8. Enter the Server Url and click on the Find Servers button.
9. Expand the OPC Server and choose a connection method.



10. Select the User Token type based on the requirements of the server.
11. Click on the Enable checkbox to ensure it is checked and click Apply.
12. Before the data source can be accessed, MDB must complete all federation tasks. This is indicated in the configuration dialog when the message **Connected** is displayed.

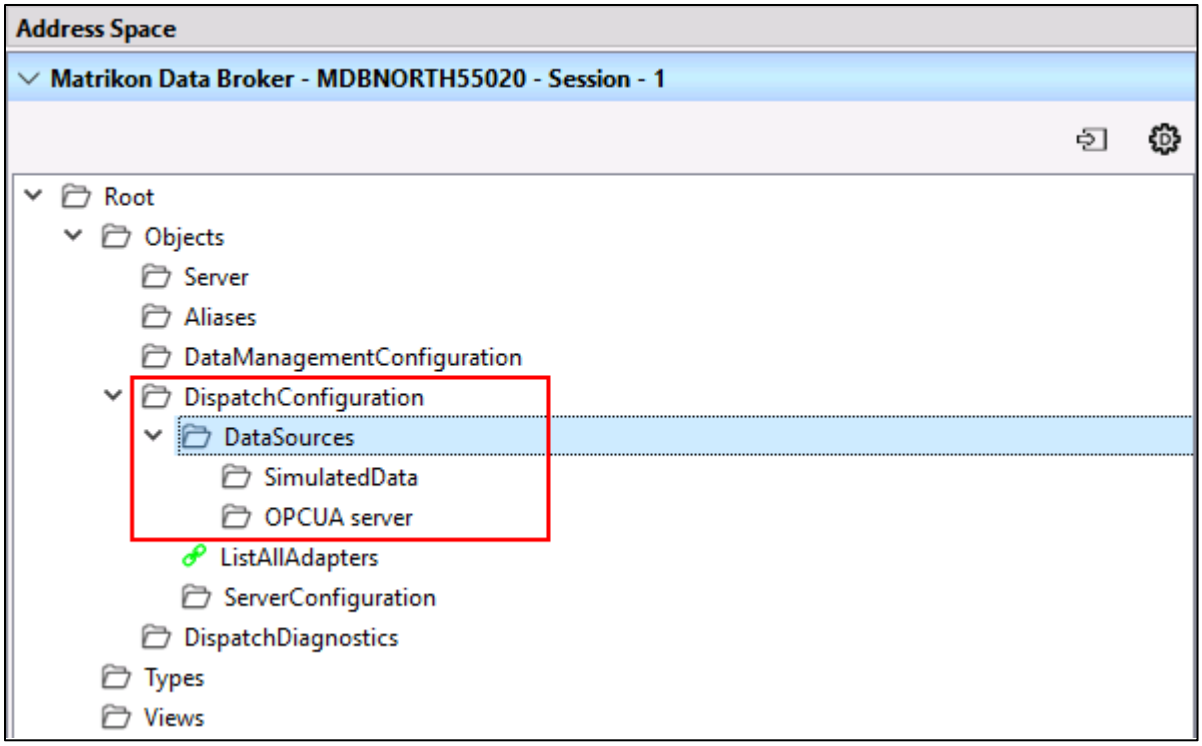


13. Collapse the Federator Data Sources tab, then expand it to refresh the Connection Status.
14. Browse the Federated data source in the Data View by clicking on the Data View icon.



15. The address space of the newly added data source appears in the **DispatchConfiguration** folder.

    Root > Objects > DispatchConfiguration > DataSources > *YourDataSource* **>**



16. The structure from this point differs from OPC server to OPC server. Expand the folders until the desired items are located.

17. Dragging the required tag(s) from the list in the Address Space to the Data View adds them to the MonitoredItemList for this server.

Accepting application instance certificates in MDB

Use Case

OPC UA security relies on the exchange of application instance certificates to authenticate connecting components. This is normally handled by the certificate management components in your infrastructure. Even where these components are not available, it remains a requirement to accept the certificates before a connection can be established. Although this can be accomplished manually, UA Explorer and MDB support this functionality.

Procedure

Manual certificate management

1. Navigate to certificate storage for rejected application instance certificates of your MDB instance.

C:\ProgramData\Matrikon\DataBroker\<*yourMDBinstance*>\pki\DefaultApplicationGroup\rejected\certs

2. Move the rejected certificate to the trusted application instance certificate storage.

C:\ProgramData\Matrikon\DataBroker\<*yourMDBinstance*>\pki\DefaultApplicationGroup\trusted\certs

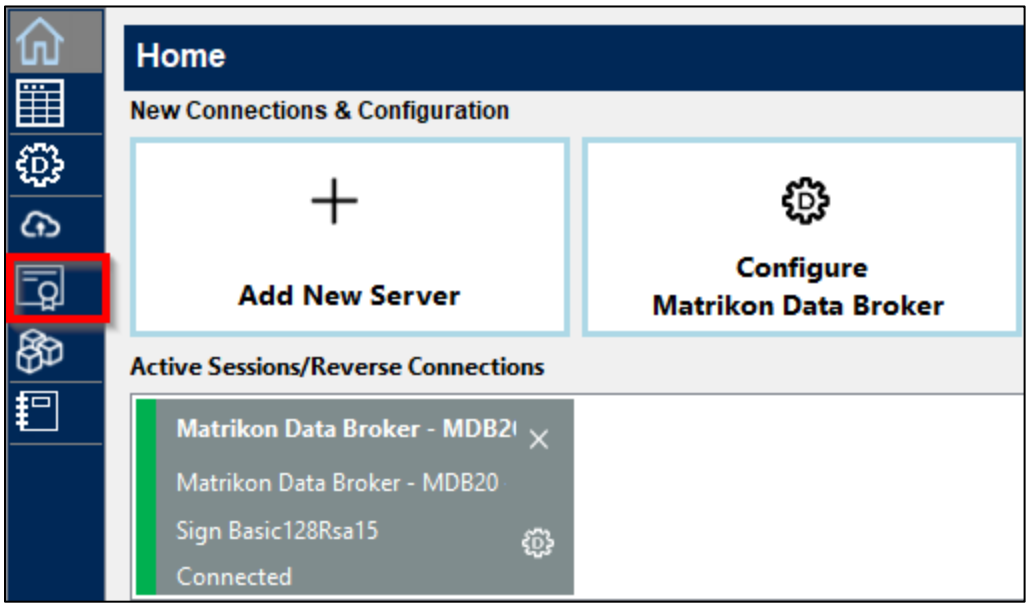
3. MDB now trusts this certificate and this application.

Via UA Explorer

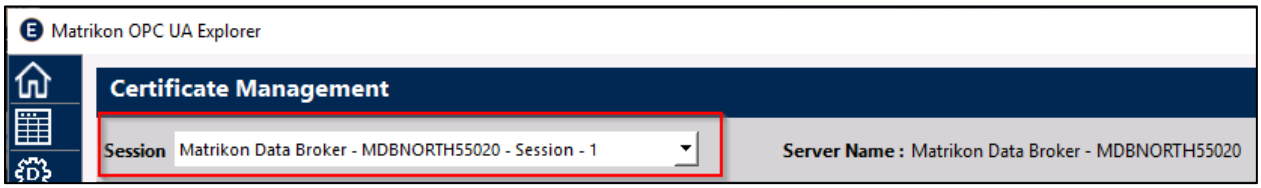
If the OPC UA server supports the certificate management methods defined in the UA specification, OPC Explorer can be used to manage the application instance certificates in the server's certificate store.

1. Open OPC UA Explorer and connect to MDB.

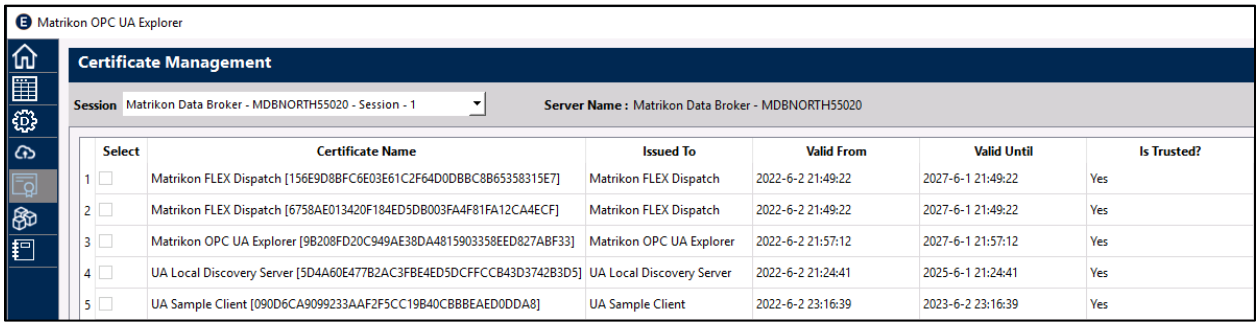
2. Click on the “Manage Certificate” icon on the left edge of the display.



3. If UA Explorer is connected to multiple servers that support certificate management methods, you must elect the server (session) for which you wish to manage the certificates. This includes multiple instances of MDB



4. Verify the status of the certificate by checking the Is Trusted? column in the Certificate Management panel. If a certificate is not trusted, select it by checking the checkbox in the left-most column.



	Select	Certificate Name	Issued To	Valid From	Valid Until	Is Trusted?
1	☐	Matrikon FLEX Dispatch [156E9D8BFC6E03E61C2F64D0DBBC8B65358315E7]	Matrikon FLEX Dispatch	2022-6-2 21:49:22	2027-6-1 21:49:22	Yes
2	☐	Matrikon FLEX Dispatch [6758AE013420F184ED5DB003FA4F81FA12CA4ECF]	Matrikon FLEX Dispatch	2022-6-2 21:49:22	2027-6-1 21:49:22	Yes
3	☐	Matrikon OPC UA Explorer [9B208FD20C949AE38DA4815903358EED827ABF33]	Matrikon OPC UA Explorer	2022-6-2 21:57:12	2027-6-1 21:57:12	Yes
4	☐	UA Local Discovery Server [5D4A60E477B2AC3FBE4ED5DCFFCCB43D3742B3D5]	UA Local Discovery Server	2022-6-2 21:24:41	2025-6-1 21:24:41	Yes
5	☐	UA Sample Client [090D6CA9099233AAF2F5CC19B40CBBBEAED0DDA8]	UA Sample Client	2022-6-2 23:16:39	2023-6-2 23:16:39	Yes

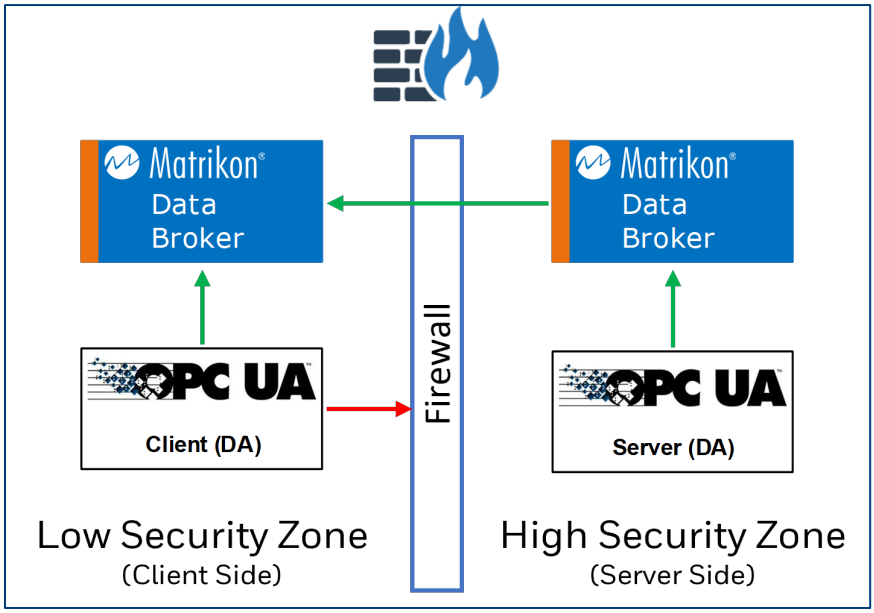
5. Click on the Accept button to move the certificate to the trusted certificate store.

6. MDB now trusts this certificate and this application.

Creating reverse connections across firewalls

Use Case

OPC servers are often installed on platforms within a high security zone, where unsolicited incoming messages are restricted by firewalls and other security mechanisms. For OPC Classic applications this required the complex configuration of multiple applications to allow communication between the client and server. The reverse connect functionality of OPC UA allows the OPC UA server in the high security zone to initiate the connection and communication. This outgoing message is not restricted by the firewall and the client application can respond without violating firewall protocols.



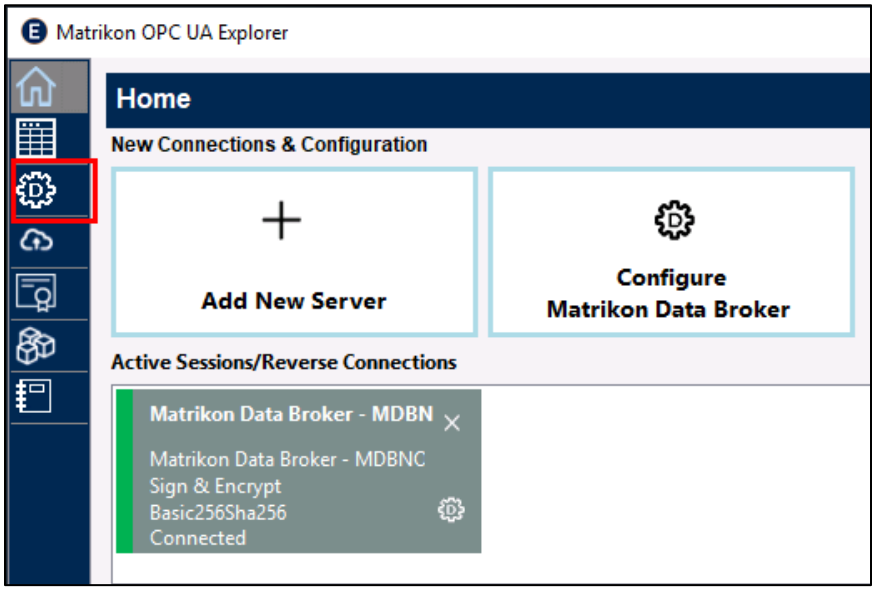
The FireBridge feature in MDB allows configuration of a reverse connection to an OPC UA client application. Ideally, the client application is enabled to invoke this service, and the server application supports the functionality. This instruction assumes that the original applications do not support this functionality, and that due to firewall restrictions the client cannot connect directly to the server. MDB acts as both client and server for this example, and UA Explorer is the configuration tool.

Procedure

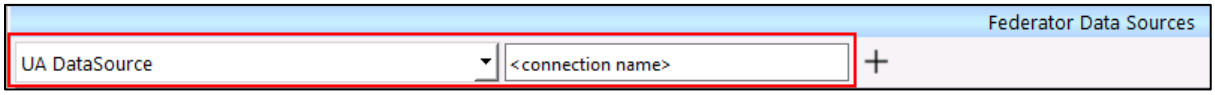
Configure the Client side by adding a data source to MDB

1. On the client side, connect Matrikon OPC UA Explorer to MDB as the admin user.

2. Click the Configure Matrikon Data Broker button.



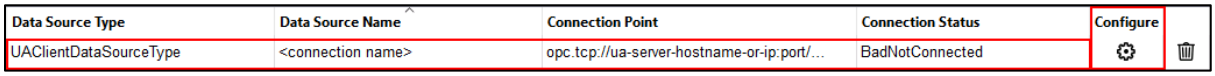
3. Expand the Federator Data Sources panel.



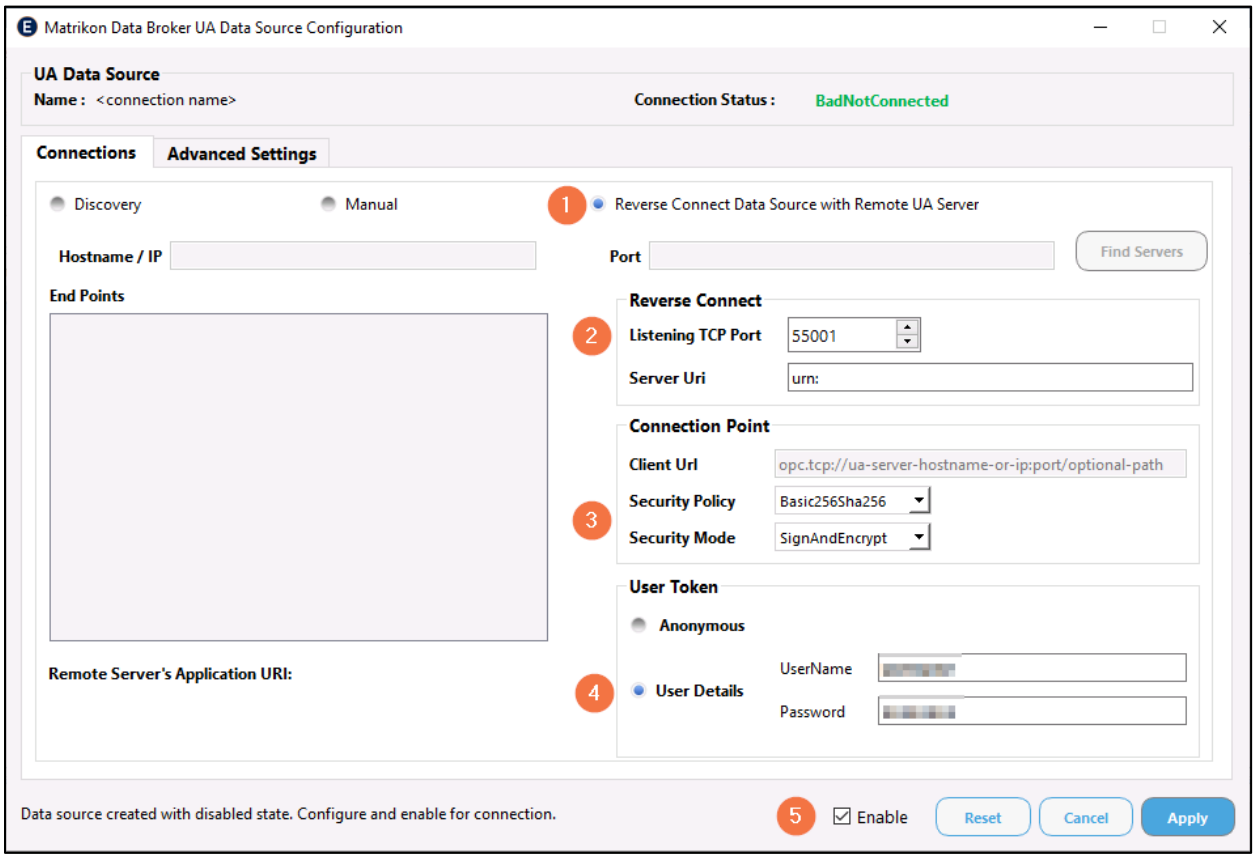
4. Ensure UA DataSource is selected from the dropdown menu and enter a name for this data source.

5. Click the '**+**' button to add this data source.

6. Click the Configure icon on the newly created data source to open the UA Data Source Configuration dialog.



7. In the MDB UA Data Source Configuration dialogue Connections tab:

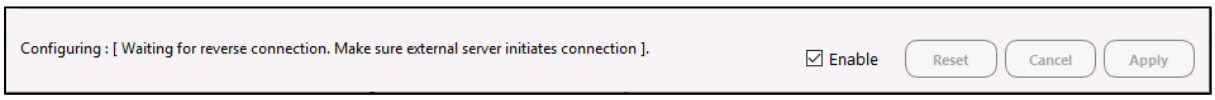


1) Select the Reverse Connect Data Source with Remote UA Server radio button option.
2) Enter the TCP port that this MDB node will use to listen for the reverse connection invitation from the server.

There is no need to enter the Server Uri, leave as-is.

3) Select the required Security Policy and Security Mode.
4) Set the User Token credentials as required by the server.
5) Ensure Enable is selected.

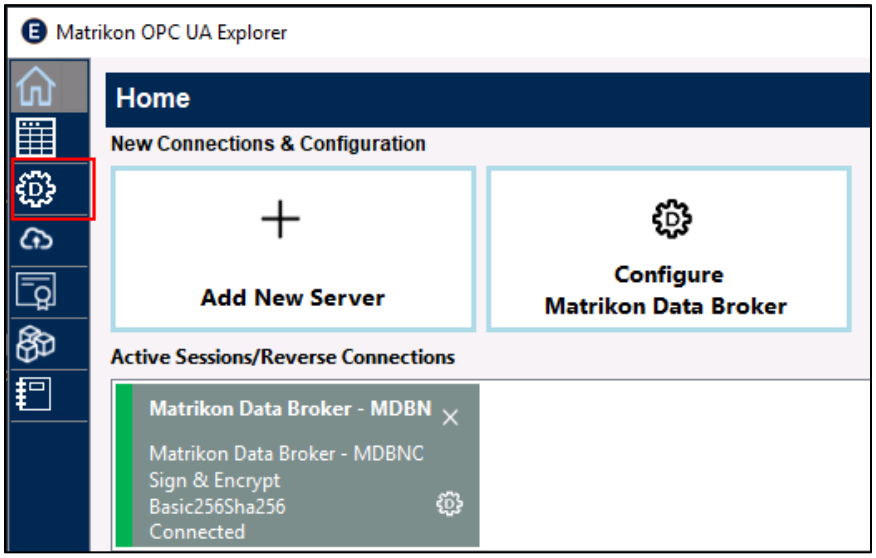
8. Click on the Apply button.
9. Note the message displayed at the bottom of the window "…Waiting for reverse connection…"



Configure the Server side

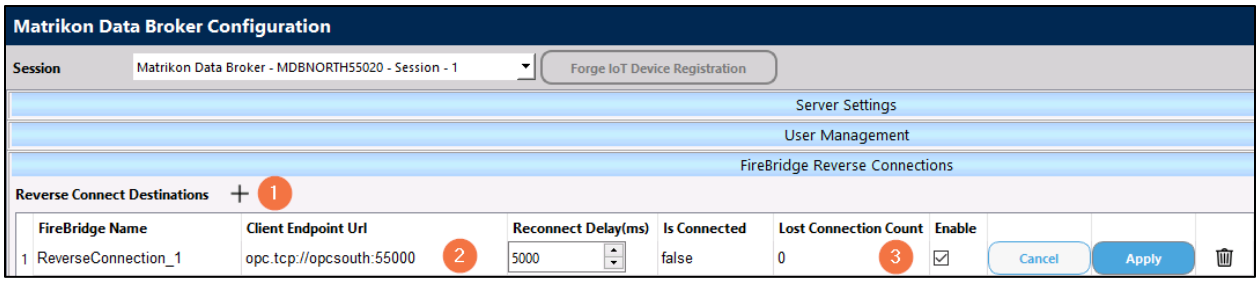
10. On the server side, connect UA Explorer to MDB as the *admin* user.

11. Click the Configure Matrikon Data Broker button



12. Expand the FireBridge Reverse Connections panel.

13. Configure the Reverse Connection Destination as follows:



1) Click the '**+**' button.

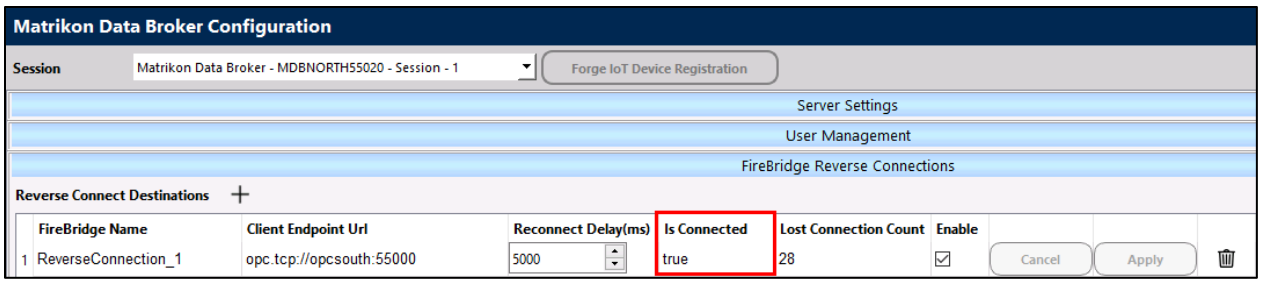
2) Double-click the existing Client Endpoint Url to edit. The client endpoint is the listening port previously configured

opc.tcp://<client-host name or IP>:<listening port>

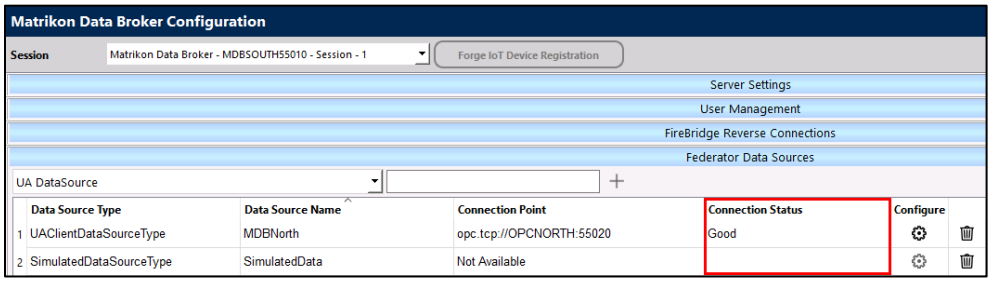
3) Ensure Enable is selected.

14. Click the Apply. Button.

15. On the server-side, verify that the MDB Reverse Connection status indicates that the Is Connected status is true.



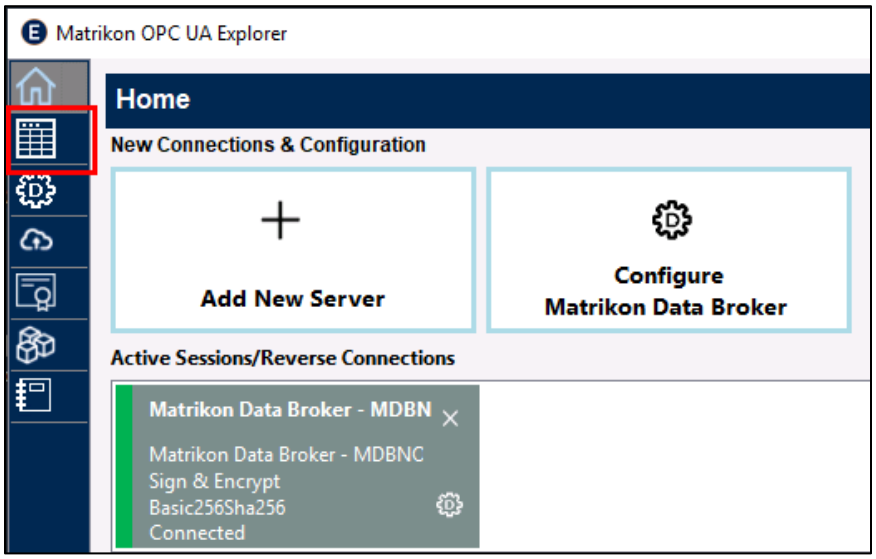
16. On the client-side, verify that the Connection Status of the Federated Data Source is Good.



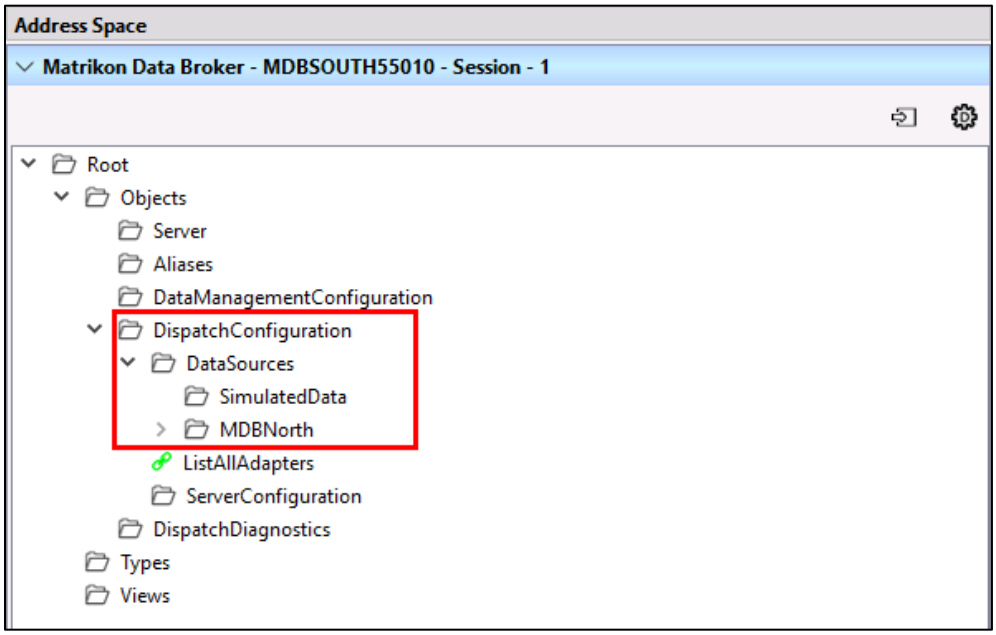
17. Connection issues at this point are likely caused by certificate errors. This can be verified by checking the log file for error messages. If certificate errors are indicated in the log, refer to the section on accepting application instance certificates.

Access the data from the OPC UA server

18. Click on the Data View icon to access this address space through MDB.



19. The client-side MDB now supports a server-initiated connection to the server-side MDB, which has been configured to connect to the remote OPC UA server. The address space for this remote server appears in the DispatchConfiguration folder in the client-side MDB address space.



20. The navigation of the remote server address space from this point can be treated as if the user were browsing it locally.

Importing information models into MDB

Use Case

Information models are used to provide a virtual definition of a real-world hardware, software, or business asset object. The information models used in OPC UA are based on the Companion Specifications developed by the OPC Foundation in cooperation with its members and partners. In addition to the definition of the model, the Companion Specifications include a *NodeSetFile* that includes the definitions of all types within the model.

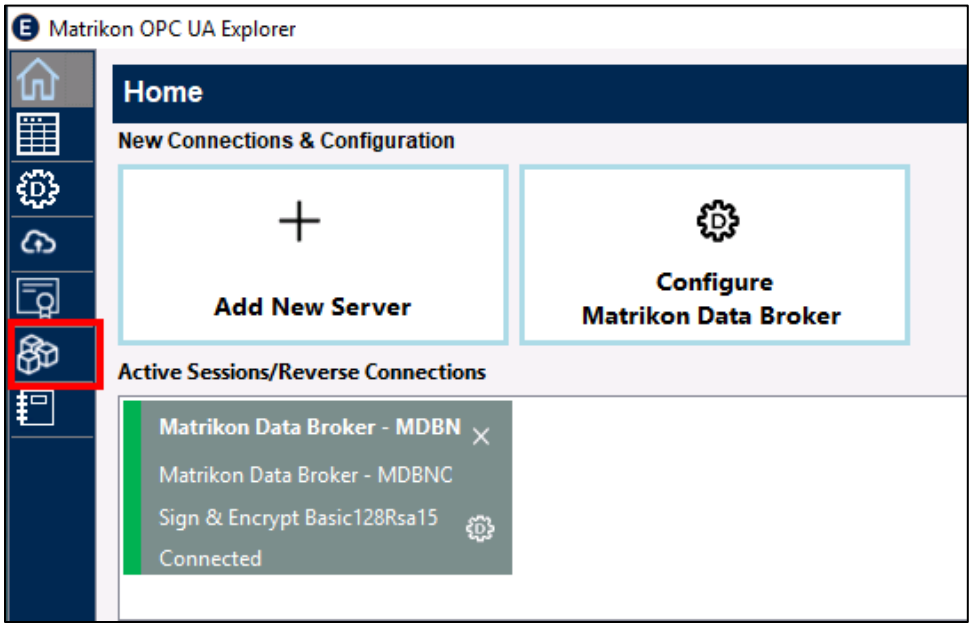
Importing the NodeSet file is the first step in configuring MDB to connect to a data source defined by the Companion Specification.

Procedure

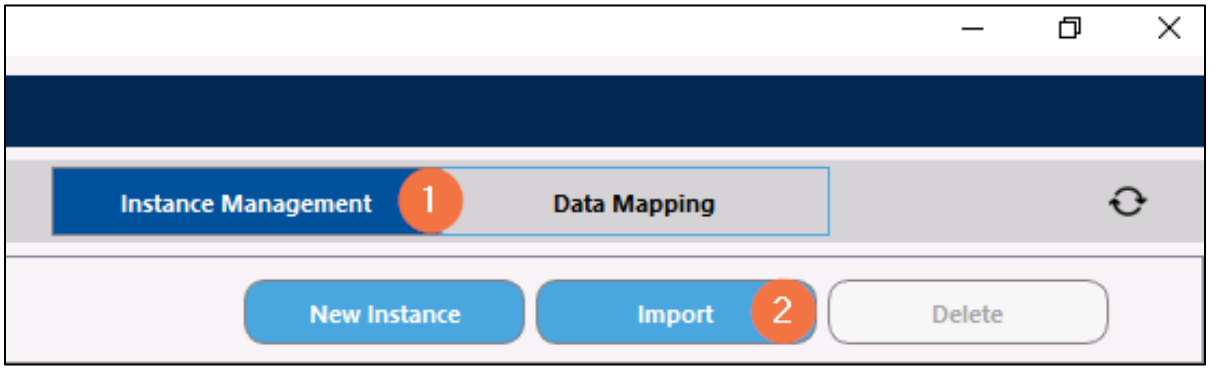
1. Ensure that all files required to represent the information model are accessible to the instance of MDB.

   NOTE - The *nodesetfile* must be in ASCII format for import into MDB. UTF8 files will cause MDB to throw an error.

2. Using OPC UA Explorer, connect to MDB and click on the Data Modeling icon in the left menu bar.

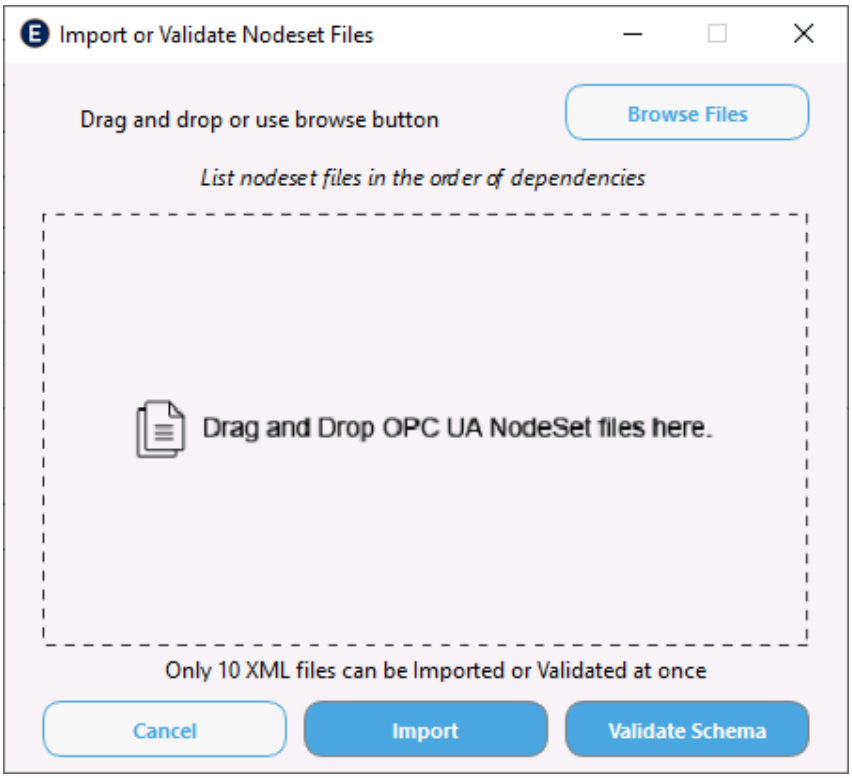


3. Click on the Instance Management tab (1), followed by the Import button (2). This opens the Import or Validate Nodeset Files dialog.



4. In the Import or Validate Nodeset Files dialogue, either

- browse for the required files by clicking on the Browse Files button, or
- drag and drop the required files from a separate window.



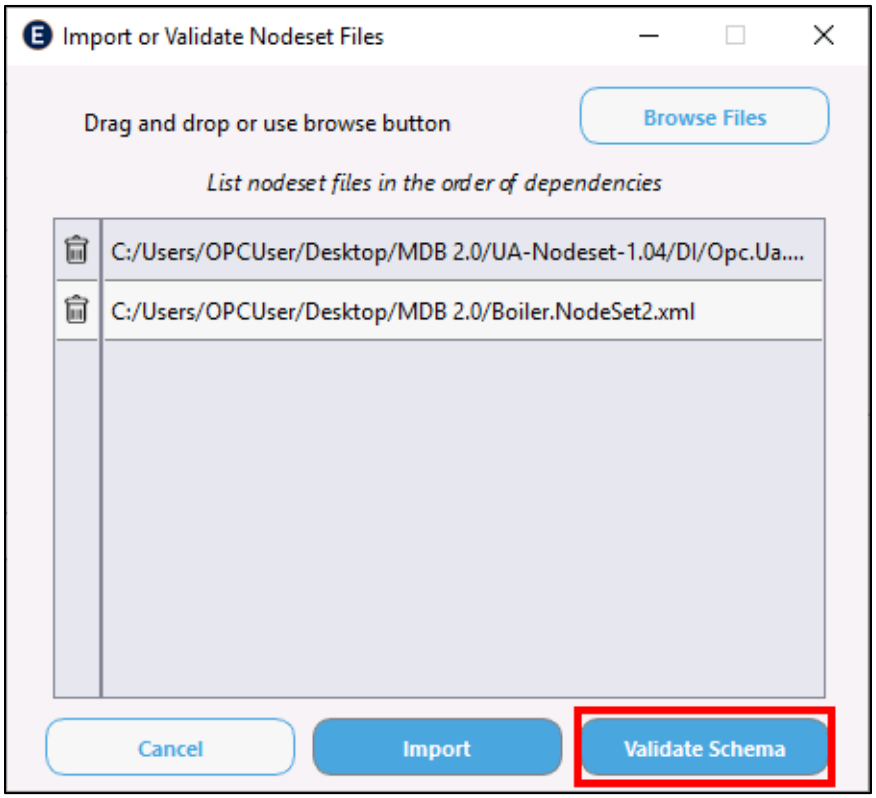
NOTE – A maximum of 10 .xml files can be imported in a single operation.

5. NodeSet files often contain dependencies, additional models that must also be imported to MDB. These dependent models are listed in the NodeSet file *(see example below)*. MDB requires that all dependent models be imported, in the order listed in the NodeSet file. The NodeSet file is the last to be listed.

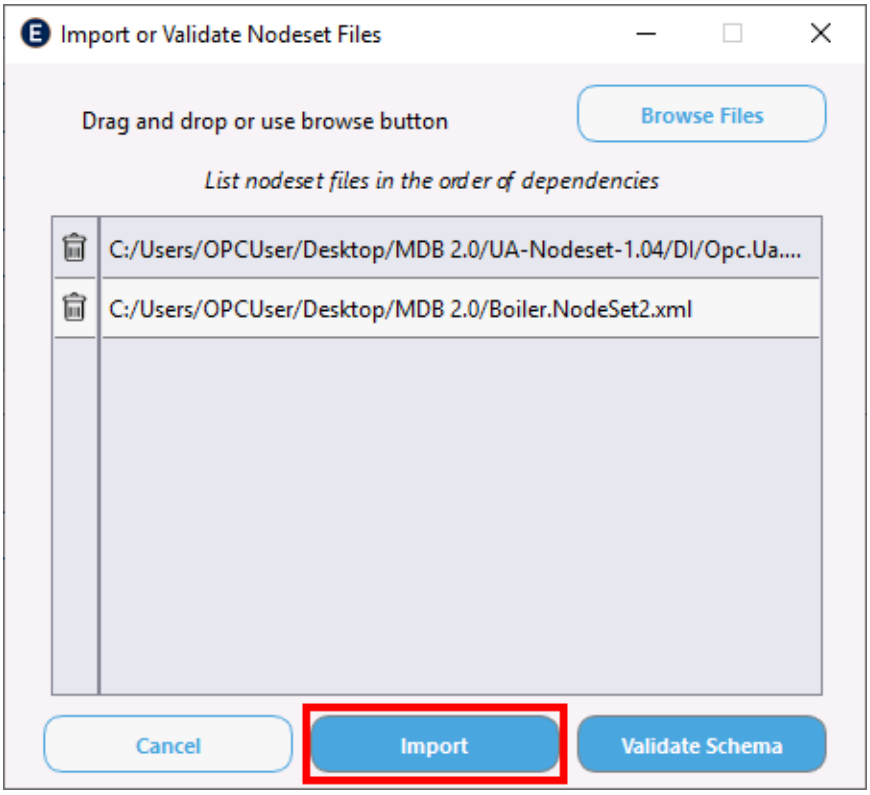
```
Models>
            <Model ModelUri="http://opcfoundation.org/UA/Machinery/"
Version="1.02.0" PublicationDate="2022-05-01T00:00:00Z">
                  <RequiredModel ModelUri="http://opcfoundation.org/UA/"
Version="1.04.7" PublicationDate="2020-07-15T00:00:00Z" />
                  <RequiredModel ModelUri="http://opcfoundation.org/UA/DI/"
Version="1.02.2" PublicationDate="2020-06-02T00:00:00Z" />
```

CAUTION – All dependent models must be included in the import list, and the xml files must be structurally and syntactically correct. Any deviation from the correct order or errors in the files will result in an error during file validation or import.

6. Once all required files have been added, click on the Validate Schema button to ensure the files conform to the UA xml schema.



7. Click on the Import button. During the import, the xml files are checked for structure and syntax. Errors will be reported on-screen and will result in an unsuccessful import.

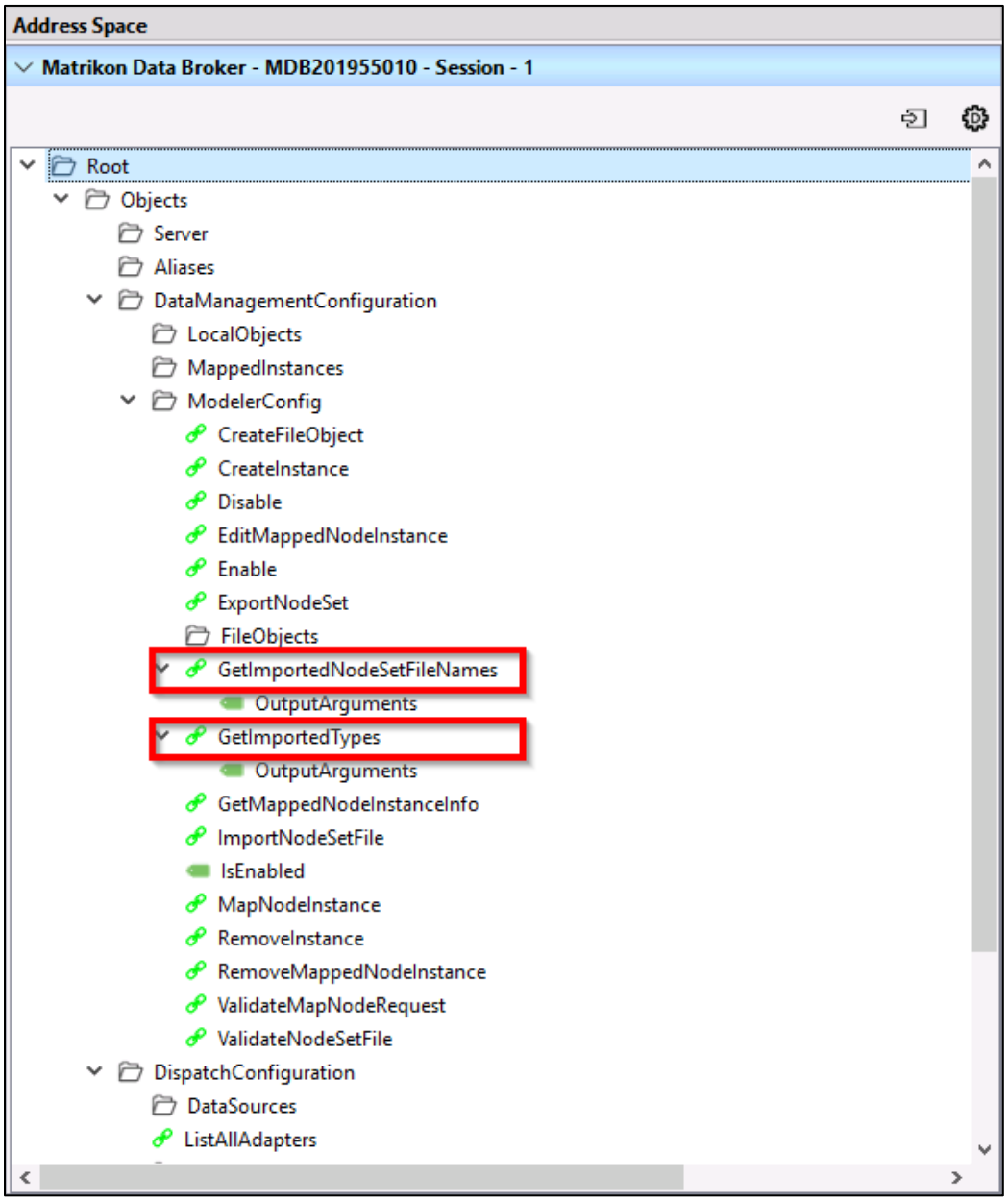


8. The successfully imported information model components can be viewed in the following locations.

   - Under Types in the Data Modeling page.
   - The imported NodeSet files are also listed in the *importednodesets* folder in the ProgramData directory.

     C:\ProgramData\Matrikon\DataBroker<*YourInstanceName*>**\importednodesets**

   - In the address space of the MDB server, navigate to the ModelerConfig directory.
     - Right-click on GetImportedNodeSetFileNames to view a list of the imported files, or
     - right-click on the GetImportedTypes to see a list of the imported types.



Creating instances of a type

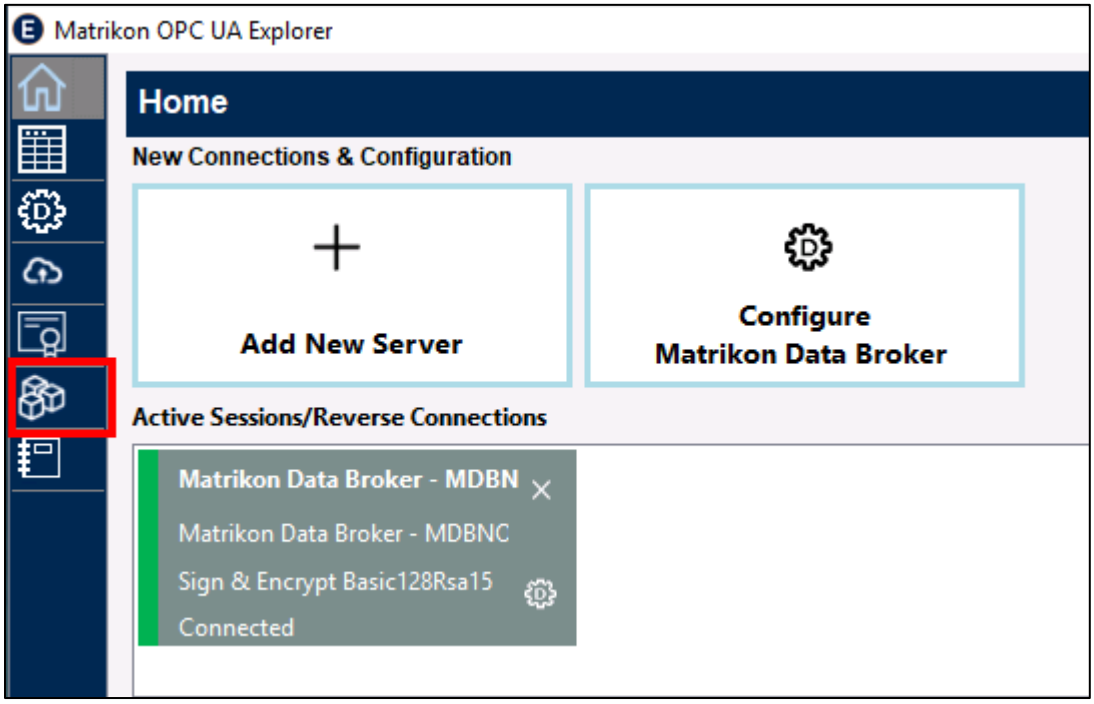
Use Case

Creation of an instance of a type in the OPC UA server address space is a part of the data or information modeling process. This functionality requires that the server implement these services and the client be enabled to invoke these services. This step is required to create computable objects within the address space of the server. There is no logical limit to the number of instances that can be created, either in total or of a single type.

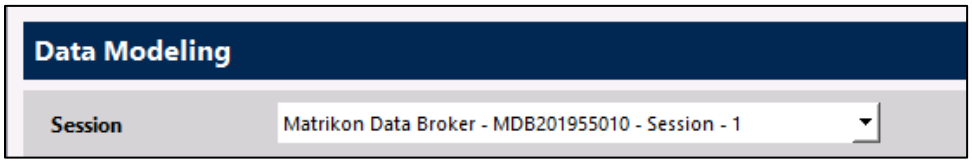
MDB supports this functionality, and this instruction describes the necessary steps to create an instance. This instruction assumes that the user is knowledgeable in OPC UA address space components and is aware of the required types and structure to be created.

Procedure

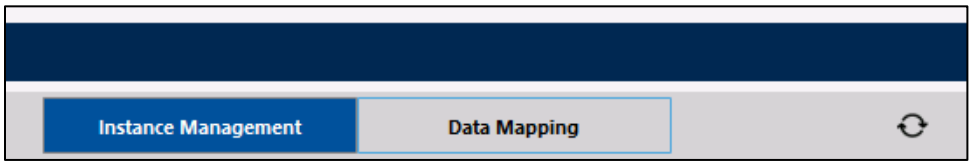
1. Connect UA Explorer to MDB and select Data Modeling from the menu items on the left edge of the UA Explorer window.



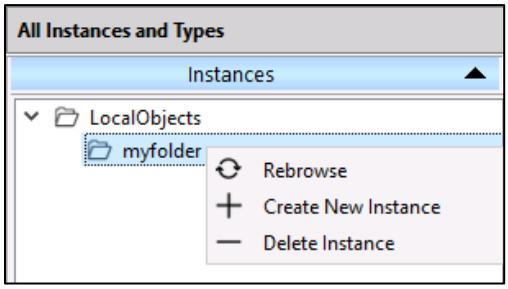
2. Select the OPC UA server whose address space you wish to manage from the drop-down in the upper-left corner of the UA Explorer display.



3. Click on the Instance Management tab in the upper-right corner of the UA Explorer display. The Instance Details options are available by default.

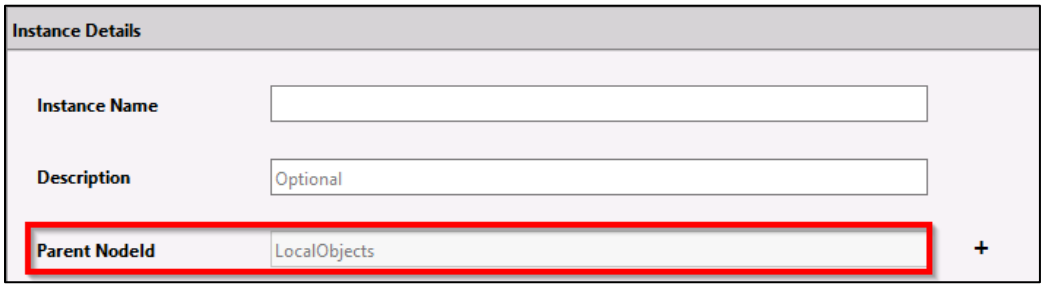


4. To create a new instance at a specific location in the address space, right-click on the node you wish to create the new instance in and select Create New Instance.

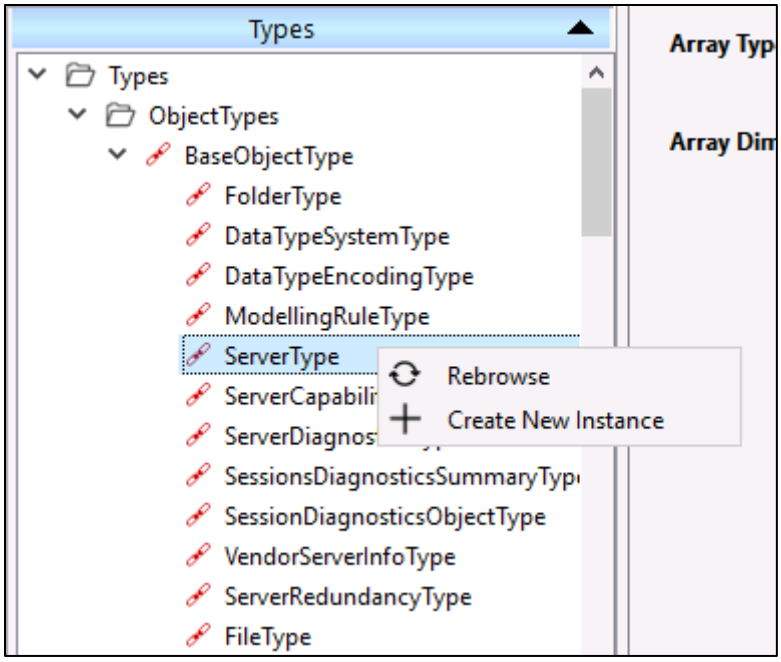


NOTE – At this time MDB supports creation of Instances within the local MDB only. This means that new instances of a type cannot be created within the address space of a Federated data sources.

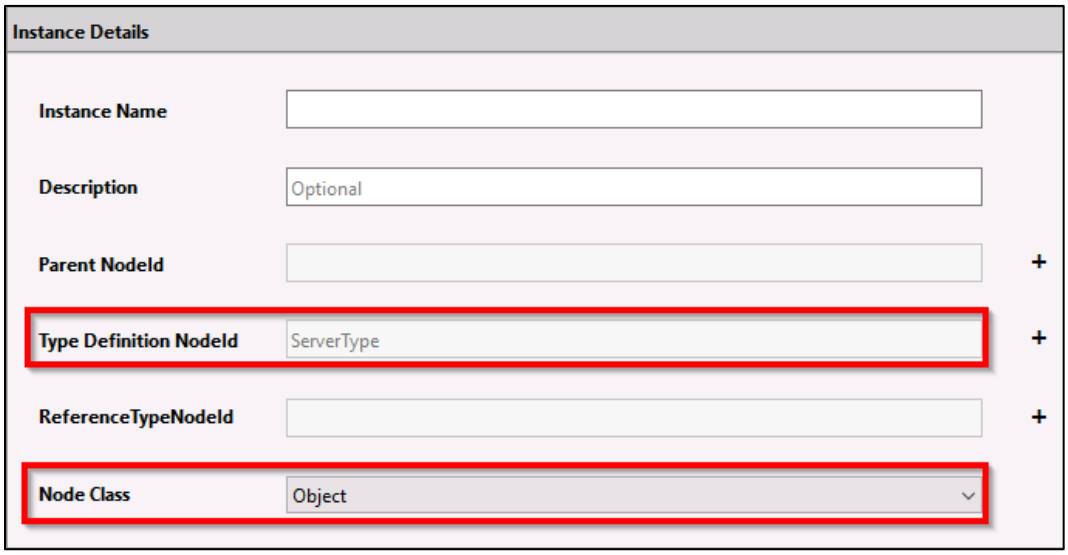
5. In the Instance Details window the Parent NodeId field is automatically populated with the name of the selected node.



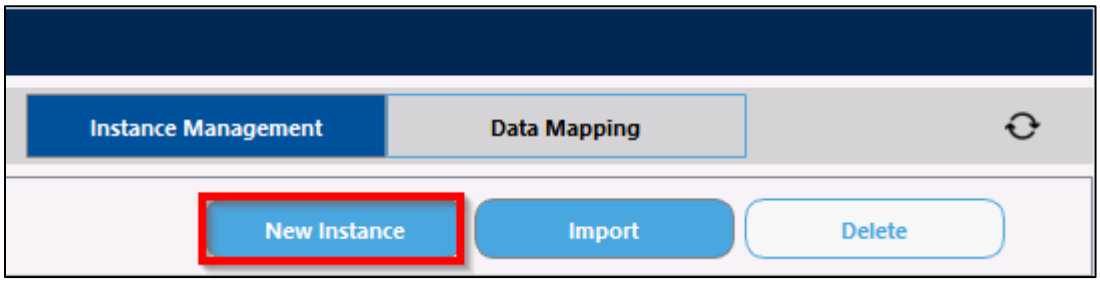
6. If you know the Type you wish to create an instance from, this can be done by right-clicking on the Type and selecting Create New Instance.



7. In the Instance Details window, the Type Definition NodeId and Node Class fields are automatically populated based on the selected Type.



8. At any time, clicking on the New Instance button clears all fields and allows manual entry of values for all fields. Clicking on the '**+**' beside a field presents a list of all possible items for that field.



9. Once all fields have been filled out as required, click on the Apply button to commit all changes.

Mapping data in MDB

Use Case

Data mapping allows a data item from a real-world data item to be connected directly to an instance of a node in the MDB address space. This provides context for the raw data from the device, based on the overall configuration of the server address space.

MDB supports the OPC UA functions for Information Modeling and Data Mapping. This includes extending the mapping function to transferring data from one real-world source item to a destination item in another real-world source.

There are four possible mappings available in MDB

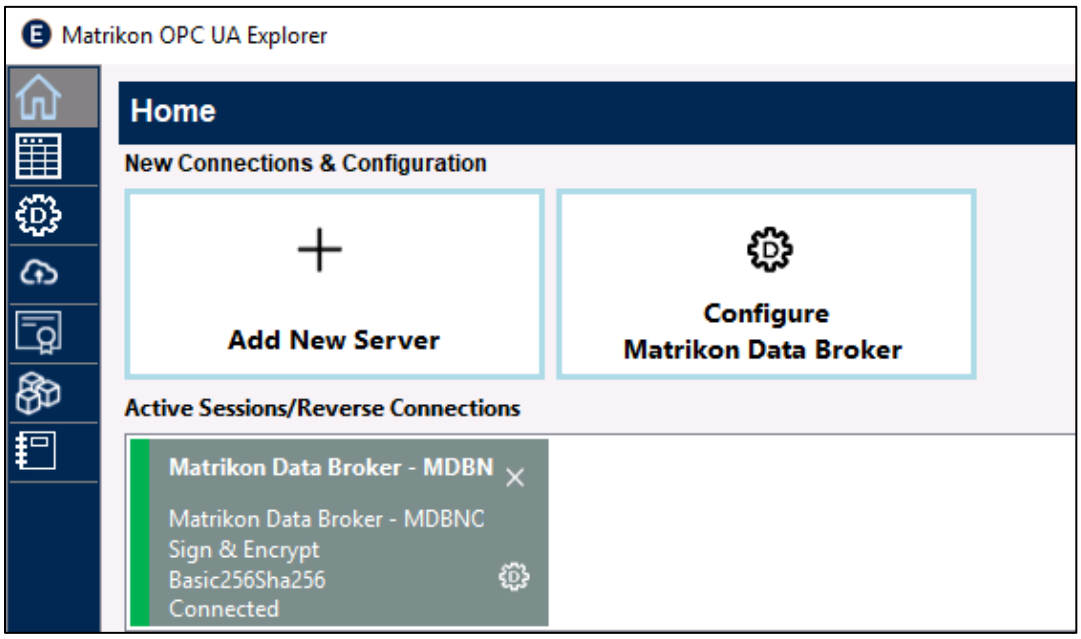
- Instance as source to instance as destination
- Instance as source to data item as destination
- Data item as source to data item as destination
- **Data item as source to instance as destination**

Of these, only the highlighted mapping of data item to instance is included in the basic license of MDB. The others require a Data Management license.

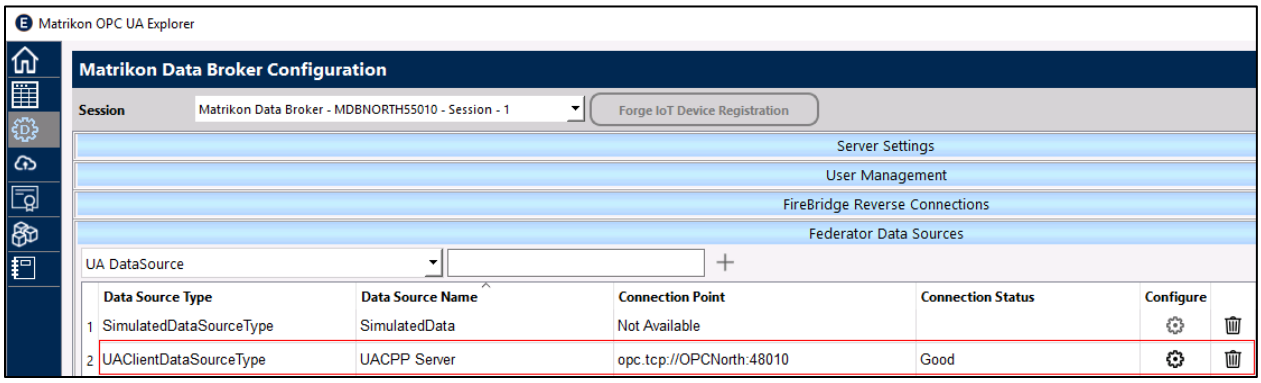
This instruction includes the data item to data item and data item to instance mappings as these are the most common uses for this feature. The procedure for both use cases is essentially the same, differing only in the selection of the source and destination items.

Procedure

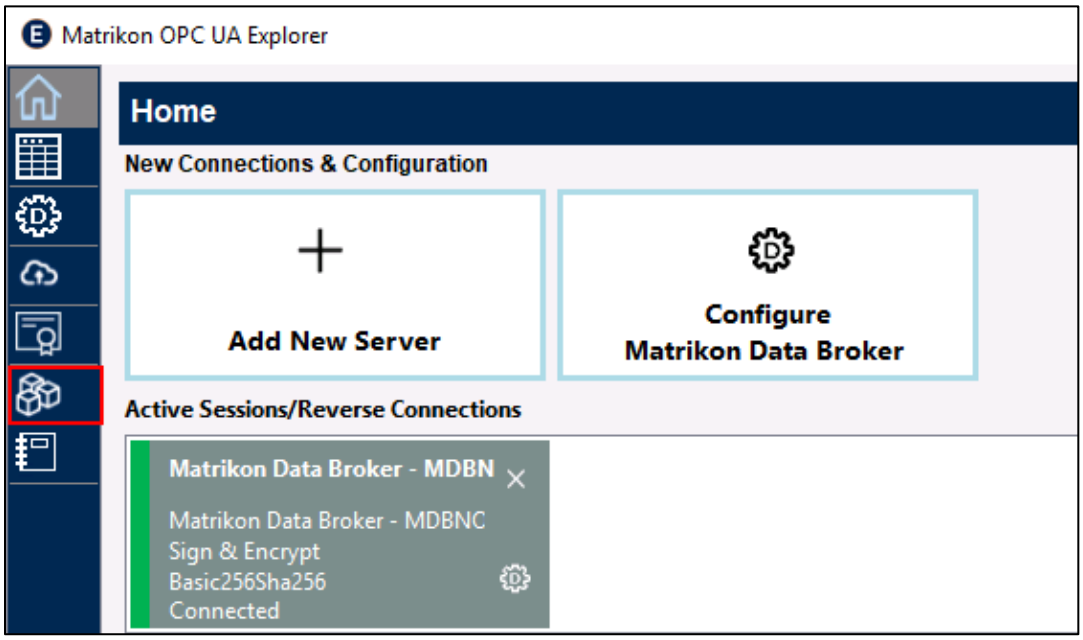
1. Connect OPC UA Explorer to MDB.



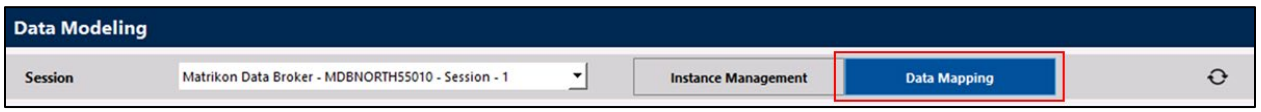
2. Connect MDB to the OPC UA Server for the real-world data item using the Federator feature of MDB.



3. Select Data Modeling by clicking on the icon in the menu bar on the left edge of the UA Explorer window.

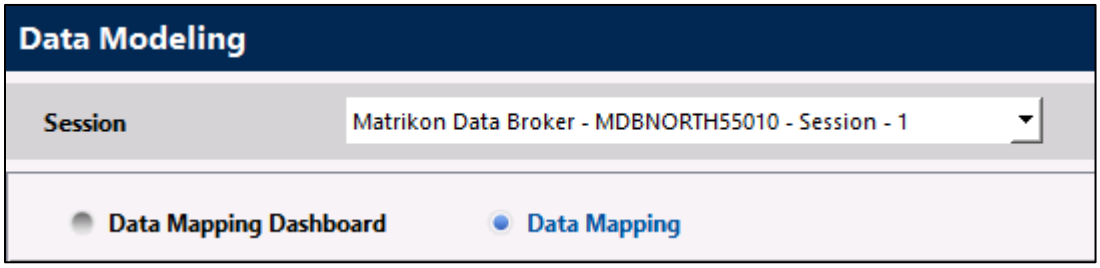


4. Select Data Mapping by clicking on the tile in the upper right-hand corner of the UA Explorer window.

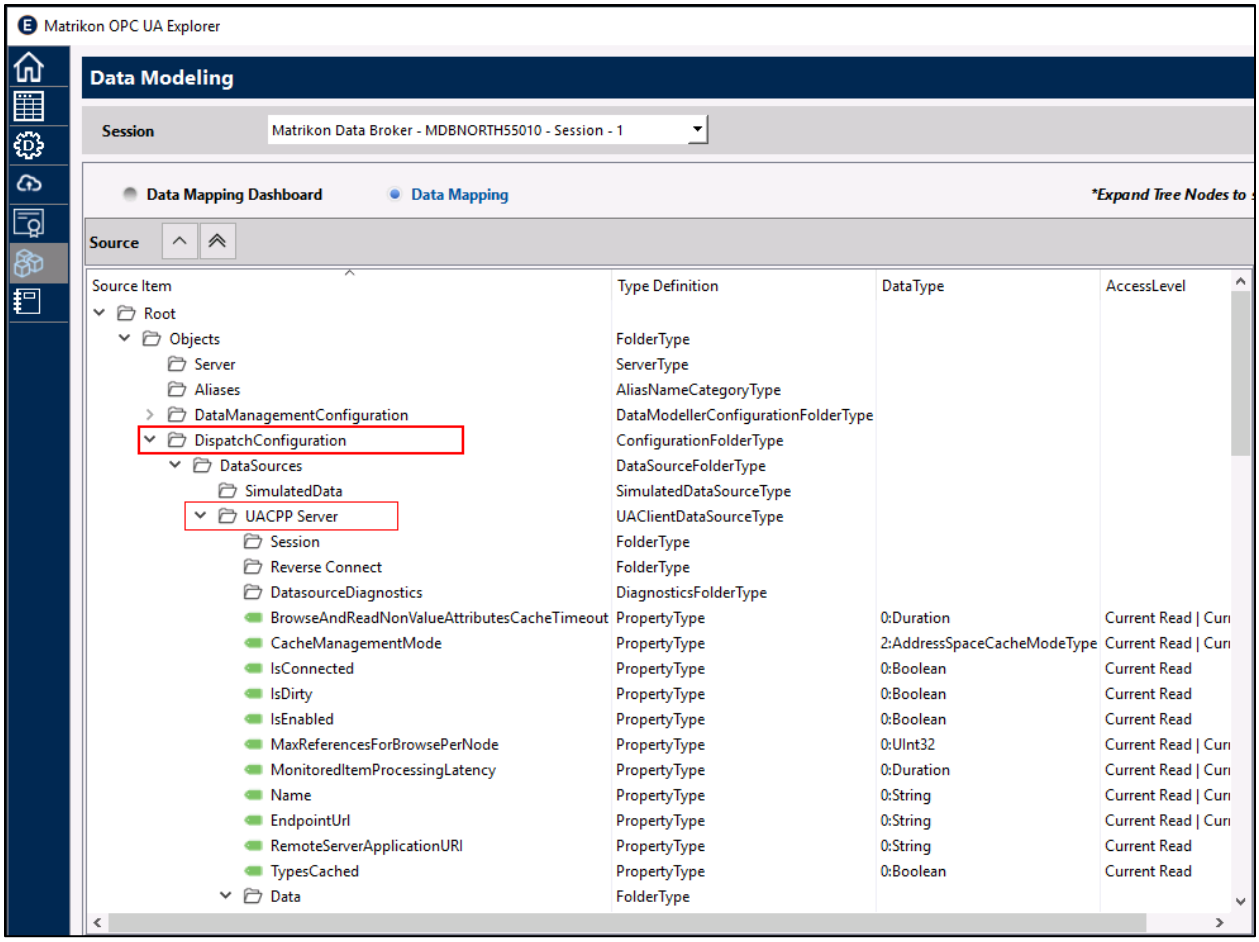


NOTE – UA Explorer supports multiple connections including multiple instances of MDB. Ensure the correct session is selected from the drop-down list on the left side of the display, under the Data Modeling label.

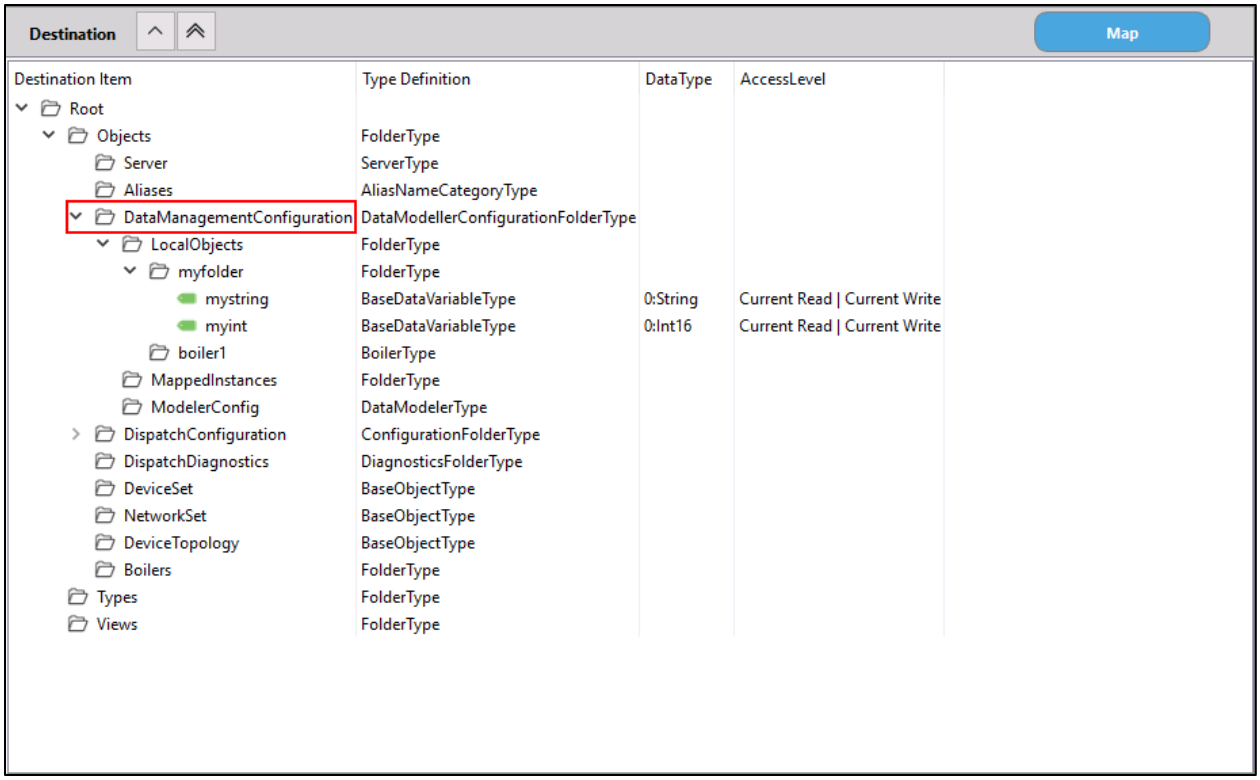
5. Select the Data Mapping radio button.



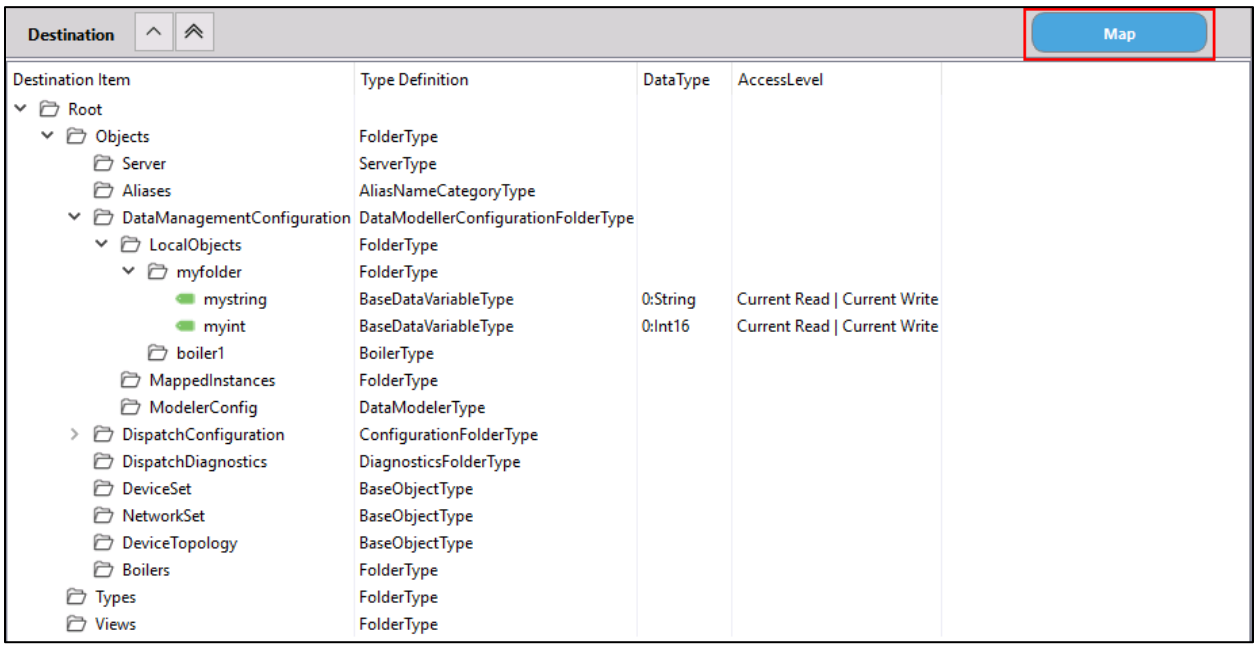
6. Select the source item. Federated data sources appear under DispatchConfiguration in the browse tree.



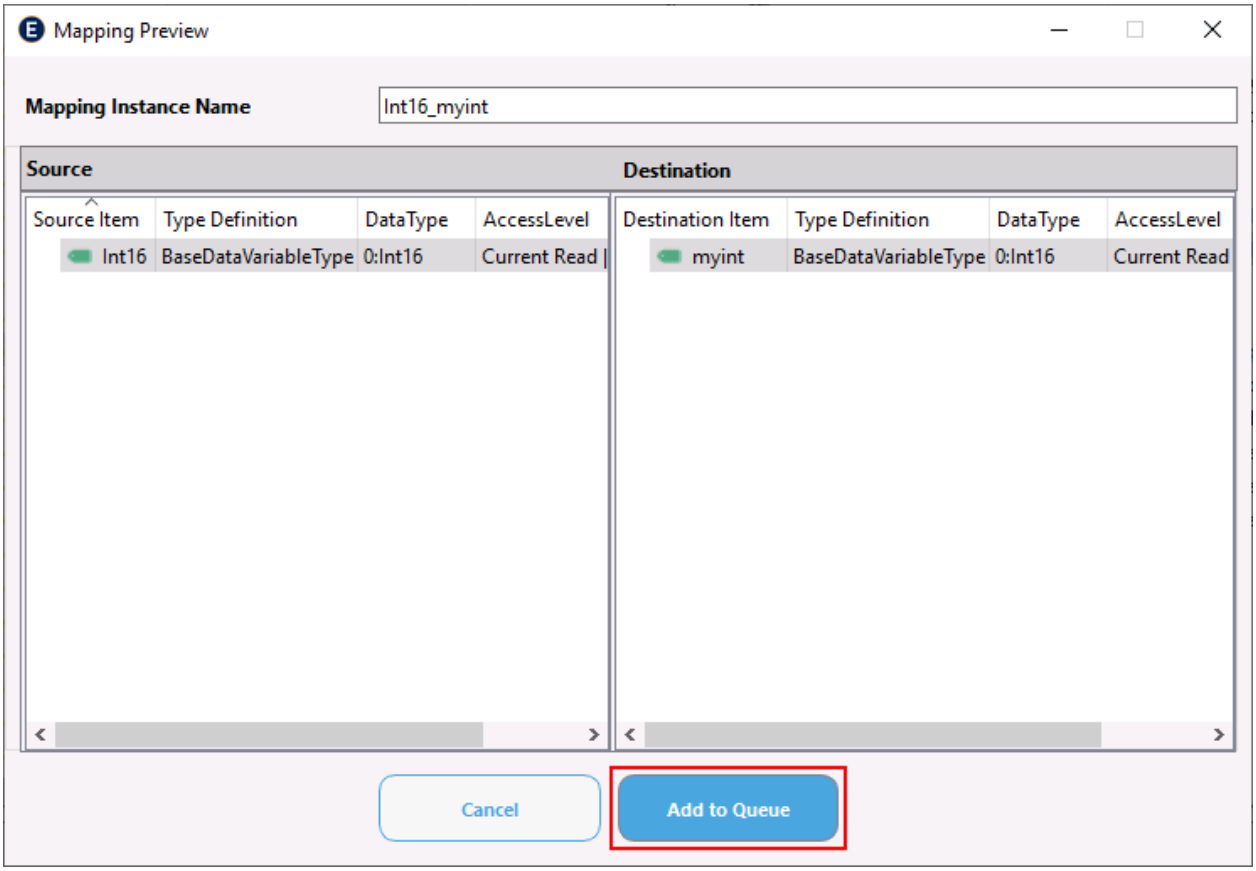
7. Select the destination item. When mapping to an instance of a type in the server address space, these appear under DataManagementConfiguration in the browse tree. When mapping to another Federated data source, these again appear under DispatchConfiguration in the destination browse tree.



8. Once you have selected the source and destination items, click on the Map button.



9. The Mapping Preview window appears. Verify the item selections, ensuring that
   - the correct source and destination items have been selected,
   - the Type Definition of the source and destination items match, and
   - the DataType of the source and destination items match.
10. Once all items are verified, click on the Add to Queue button.

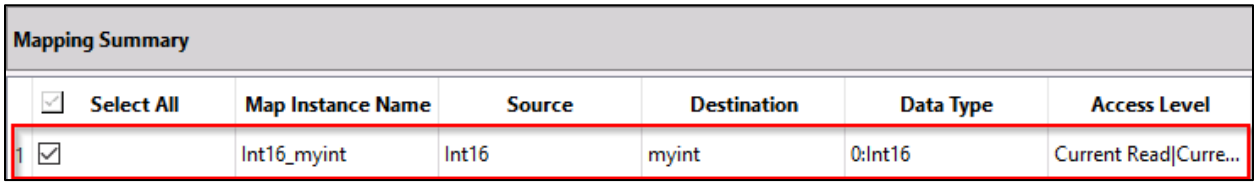


11. Ensure the newly mapped items are displayed in the Mapping Summary.

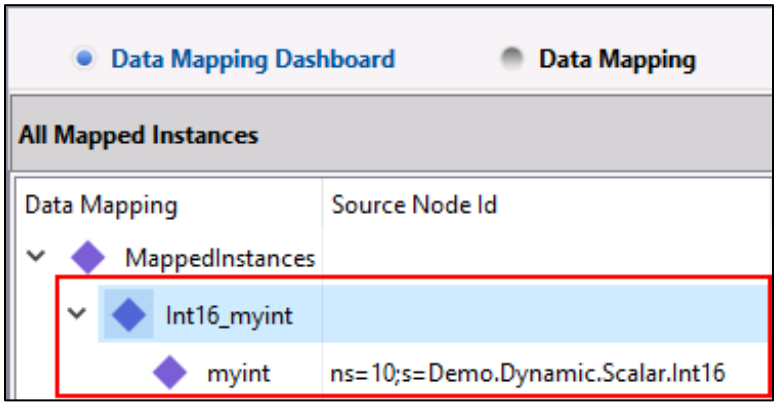
Mapping Summary

	Select All	Map Instance Name	Source	Destination	Data Type	Access Level
1	☐	Int16_myint	Int16	myint	0:Int16	Current Read|Curre...

12. Click on the checkbox to select the item(s), then click on the Commit button on the right side of the Mapping Summary panel.



13. Select the Data Mapping Dashboard to view all mapped items.



Installing an MDB container on a Linux platform

Use Case

Unlike OPC Classic, OPC UA applications can be developed to run on Linux platforms. Although this can be done natively, it is often easier to employ Docker containers to run the UA applications in a Windows environment on a Linux platform. This instruction provides the details for creating this environment in the **Ubuntu 20.04** Linux release.

Procedure

To prepare the system

1. Ensure the Linux installation has a connection to the internet.
2. Run the following commands prior to installation of the Matrikon Data Broker Docker container:

```
sudo apt-get update
sudo apt update && sudo apt upgrade -y
sudo apt install -y ca-certificates curl gnupg lsb-release
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
echo "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update
```

To install Docker

Following system preparation, run the following commands to install Docker:

```
sudo apt install docker-ce docker-ce-cli containerd.io -y
sudo usermod -aG docker sysadmin
newgrp docker
docker version
sudo systemctl status docker
```

To test the Docker installation

Run a simple built-in test to confirm Docker is working:

```
docker run hello-world
```

To install MDB

1. Copy the Matrikon Data Broker app image (.tgz file) to Linux.
2. Load the app image into Docker:

```
sudo docker load -i matrikondatabroker-rhelubi8-2.0.0.tgz
```

3. Create a temporary folder. Open New Terminal window then enter:

```
cd /tmp
mkdir MDB
```

4. Close window
5. Go back to original terminal window and issue this command to create the Docker Container:

```
sudo docker create --network host --name MDBContainer --volume
/tmp/MDB/:/mnt/dispatch -e DATABROKER_NAME=MDB -e DATABROKER_PORT=52000
-e DATABROKER_SECURE=1 -e DATABROKER_LOG_TO_VOLUME=1
matrikondatabroker-rhelubi8:2.0.0
```

NOTE - MDBContainer is what the Docker Container will be named. MDB is what the DataBroker instance will be named. TCP Port 52000 will be the port that MDB is listening on. All these variables can be customised by the user to suit the installation.

To configure the firewall

1. Start the newly created Container:

```
sudo docker start MDBContainer
```

2. Open the MDB TCP port in the firewall. In this example, this will be port 52000:

```
sudo ufw allow 52000
```

Note that no reboot is required unless enabling or disabling firewall.

To test MDB

1. Locate the Linux IP address. This can be done by issuing the `ip address` command.
2. Test the connection with a test OPC UA Client like Matrikon OPC UA Explorer. There are two options here:
   a. Test locally by installing Matrikon OPC UA Explorer on Linux
   b. Test remotely by installing Matrikon OPC UA Explorer on a Microsoft Windows Operating System that is on the same network as the Linux installation.

   The endpoint for MDB is opc.tcp://<IP where container is running>:<port>, in this example this is opc.tcp://10.0.0.1:52000
3. Connect with the username admin and a blank password. Expect a security error to be returned.

To accept the UA client certificate

1. Return to Linux and locate the installation location:

```
docker inspect <Container name> | grep volume
```

in this example:

```
docker inspect MDBContainer | grep volume
```

2. Copy the location returned.
3. Elevate to interactive shell:

```
sudo -i
```

4. Change directory to the copied location:

```
cd <copied location>/pki/DefaultApplicationGroup/rejected/certs
```

5. Check the contents of the folder, there should be a certificate from the OPC UA Client present:

```
ls
```

6. Move the file from this folder to the trusted folder:

```
mv <filename> <copied
location>/pki/DefaultApplicationGroup/trusted/certs
```

Note the space between filename and copied location in the above command

7. Exit from interactive mode **(most important!)**:

```
exit
```

To confirm MDB installation

Return to the OPC UA Client and attempt the connection again. As the certificate has been trusted, this action will now succeed.

NOTES

Every time the Linux installation is rebooted, the Docker container will need to be started after the reboot has completed:

```
sudo docker start <Container name>
```

Stop Docker container:

```
sudo docker stop <Container name>
```

Restart Docker container:

```
sudo docker restart <Container name>
```

Publishing data from MDB to the cloud

Use Case

Industrial Control Systems (ICSs) in the 21st Century increasingly make use of the latest technologies defined by and incorporated in IIoT and Industrie 4.0. This includes the use of cloud applications and storage capabilities. The MQTT Publisher adds this functionality to MDB, allowing any data collected by MDB to be pushed to the cloud using the MQTT protocol. The MQTT Publisher is a separate product package which must be co-installed with MDB

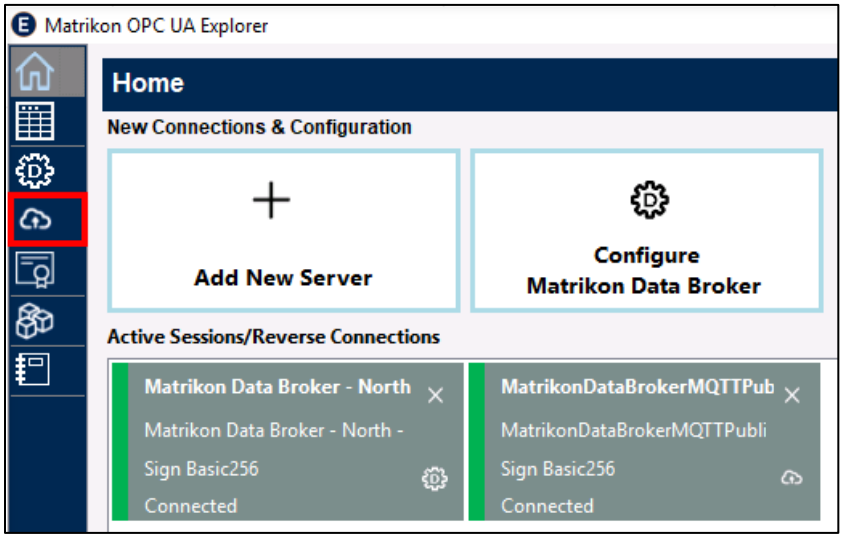
In this procedure, Microsoft Azure is acting as the MQTT broker. This requires that an Azure account be set up and that the credentials for this account are available to the user.

Procedure

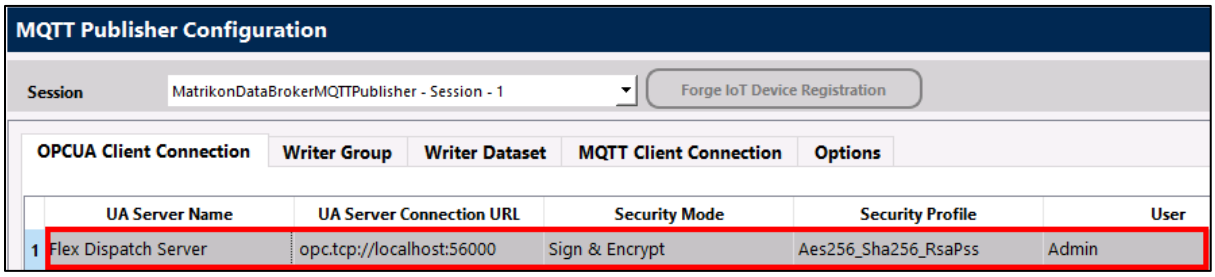
1. Connect OPC UA Explorer to both MDB and the MQTT Publisher.

Configure the connection to the OPC UA server

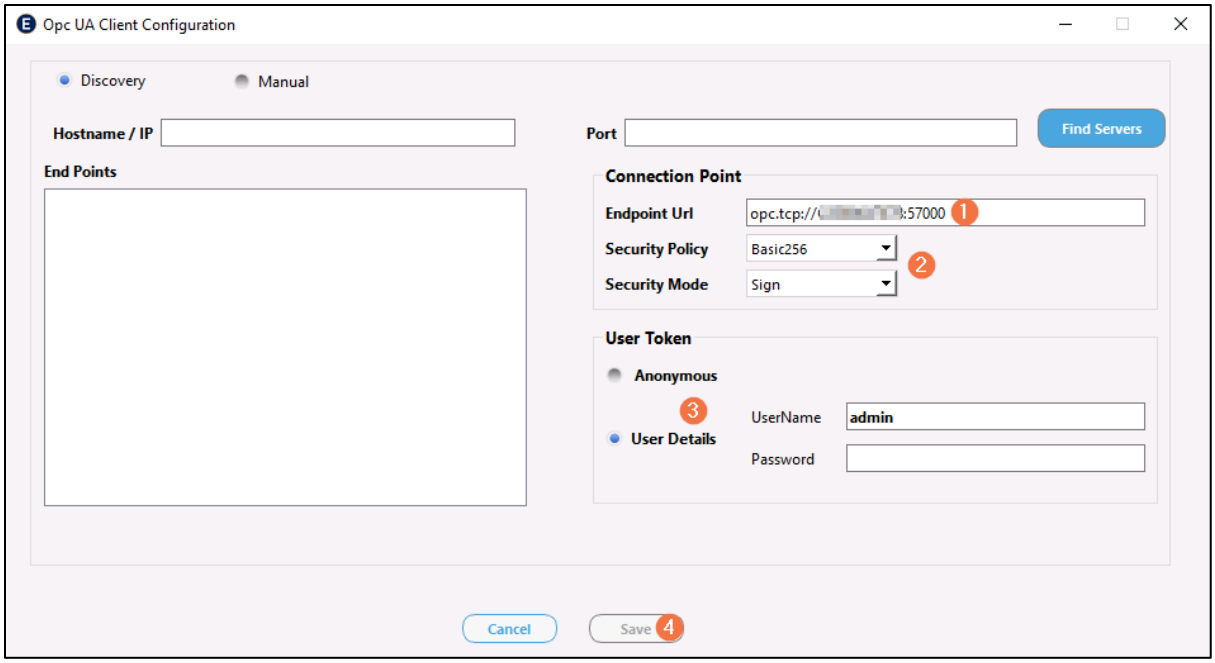
2. Click on the Manage MQTT Publisher Configuration icon in the menu on the left-hand edge of the UA Explorer window.



3. In the MQTT Publisher Configuration panel, select the OPCUA Client Connection tab and double click on the Flex Dispatch Connection.

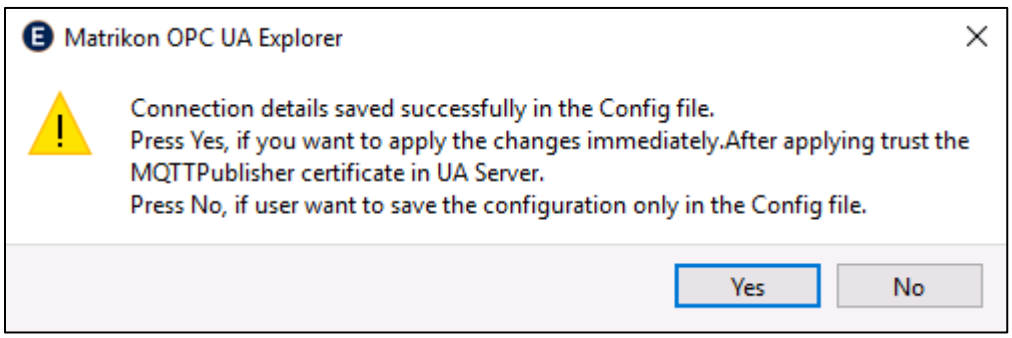


4. In the OPC UA Client Configuration window modify the following fields:



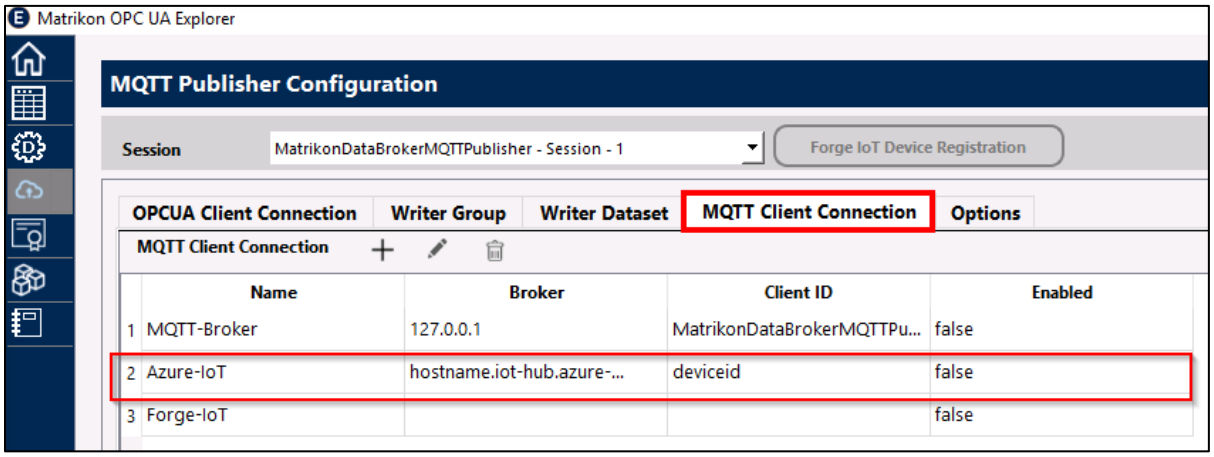
1) Change the endpoint to reflect Matrikon Data Broker's endpoint. In this example, Data Broker is listening on port 57000. Ensure to use the endpoint containing the hostname and not localhost.
2) Set the security policy and mode.
3) Set the User Token type and enter username and password. In this example this is admin with no password.
4) Click Save

5. A confirmation message appears. This message is being reworded for the next version of MDB and MQTT Publisher.
   a. Click Yes to restart the MQTT Publisher immediately and apply the configuration changes. Ensure MDB and the MQTT Publisher accept the new certificates *(see step 10)*.
   b. Click No to save changes to the configuration file only. Changes will then be applied at the next restart of MQTT Publisher.



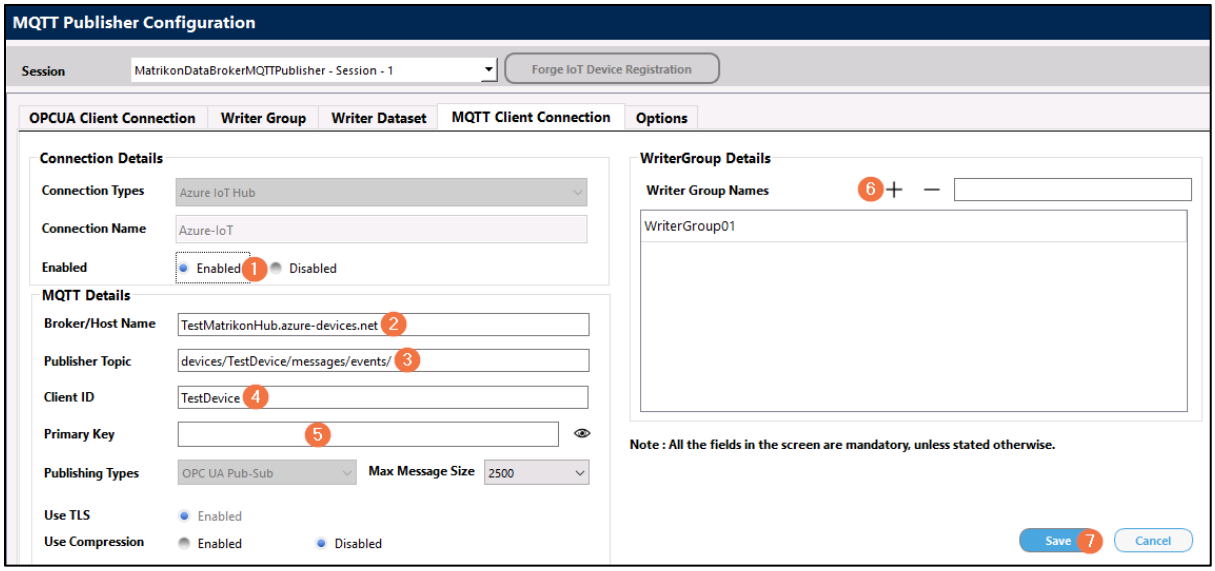
Configure the connection to the cloud platform

6. Select the MQTT Client Connection tab.



7. Double-click on the Azure-IoT item

8. Modify the fields as follows:



1) Set to Enabled
2) Copy the Hostname mentioned in the Azure IoT Hub Device configuration and paste it into Broker/Host Name.
3) Change the Publisher Topic to replace *deviceid* with the Device ID from Azure IoT Hub Device configuration, leaving the rest in-tact
4) Change the Client ID from *deviceid* to the Device ID from Azure IoT Hub Device configuration
5) Paste the Primary Key in from the Azure IoT Hub Device configuration
6) Enter the Writer Group name in the JSON file and click on the **'+'**.

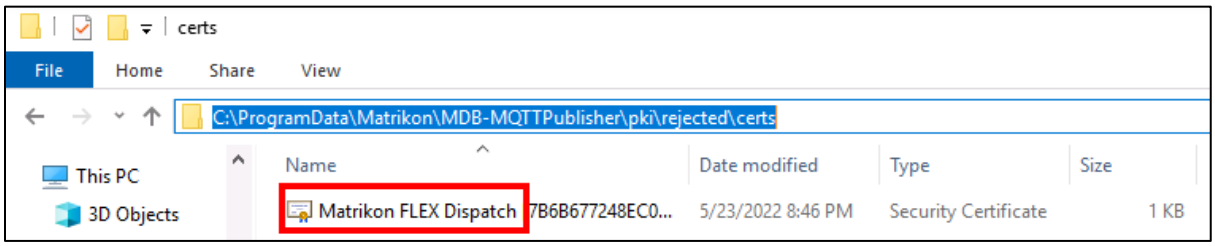
7) Click Save

9. Click Apply on the right side of the window. Close Matrikon OPC UA Explorer.

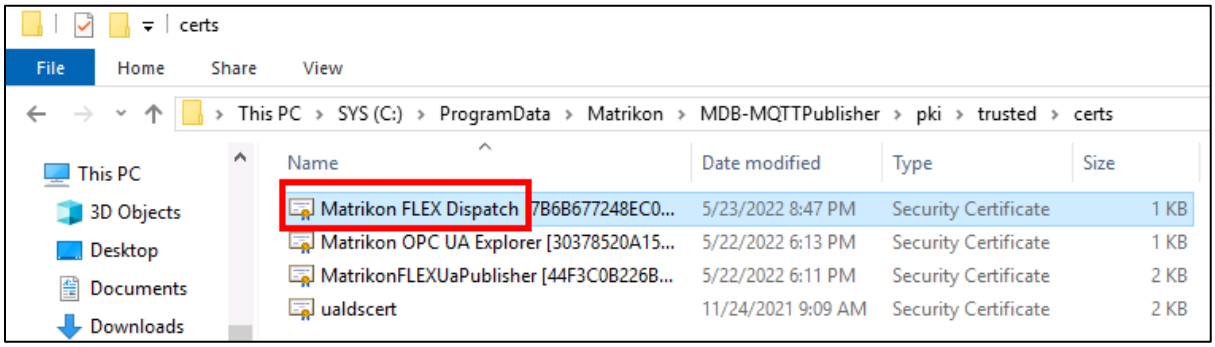
Confirm certificate acceptance by MDB and MQTT Publisher

10. Navigate to the MQTT Publisher certificate store

C:\ProgramData\Matrikon\MDB-MQTTPublisher\pki



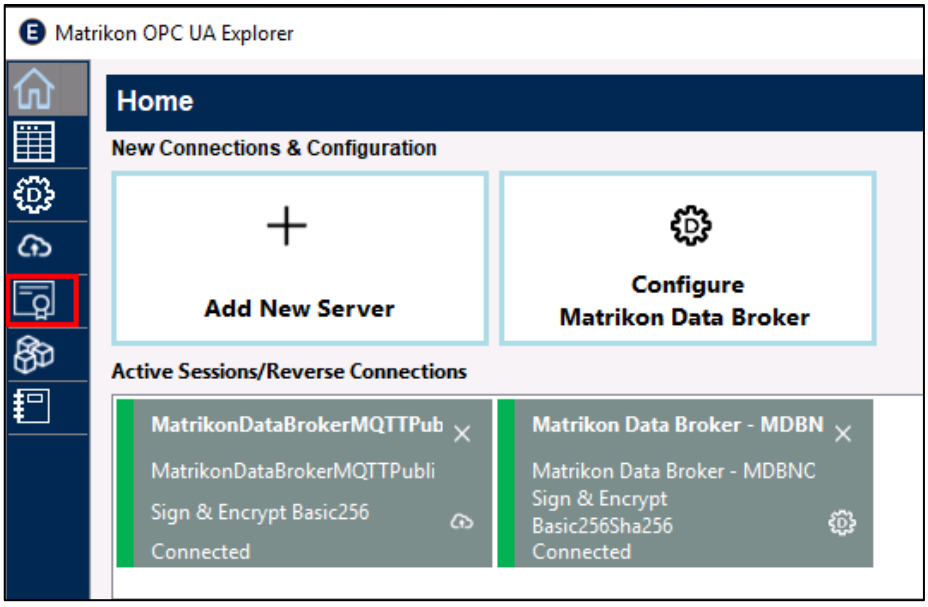
11. Locate the Matrikon FLEX Dispatch certificate in the **\rejected\certs** folder and move it to the **\trusted\certs** folder.



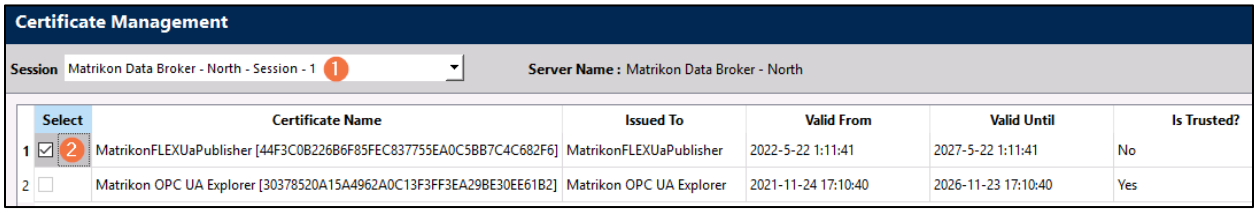
12. Open Windows Services and restart the following:

- Matrikon Data Broker - <instance name>
- Matrikon Data Broker – MQTT Publisher

13. In the UA Explorer **Home** window, confirm both MQTT Publisher and Data Broker are reconnected (green). Select Manage Certificate.

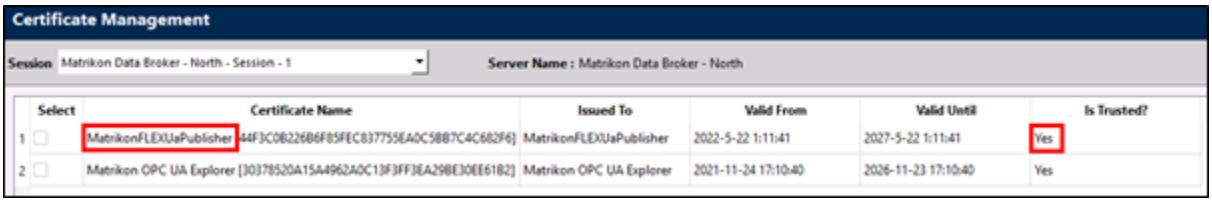


14. In the Certificate Management panel;



1) Select Matrikon Data Broker from the dropdown window
2) Select the certificate called MatrikonFLEXUaPublisher

15. Click on the Accept button. Note that the **Is Trusted?** status has changed to Yes. This indicates that MDB and the MQTT publisher are now communicating.



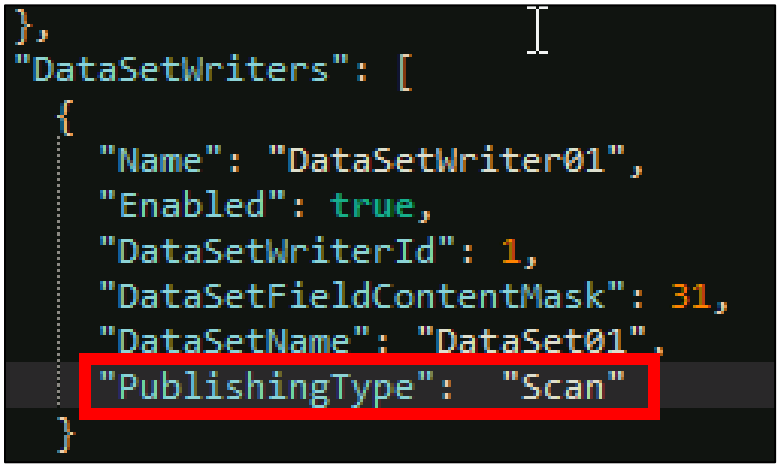
Set up the MQTT Publisher configuration

16. Navigate to **C:\ProgramData\Matrikon\MDB-MQTTPublisher\Config\writergroups**
17. Open **writergroups.json** in a text editor.
18. Locate WriterGroup01 (the name of the Writer Group added to the JSON file in step 8).

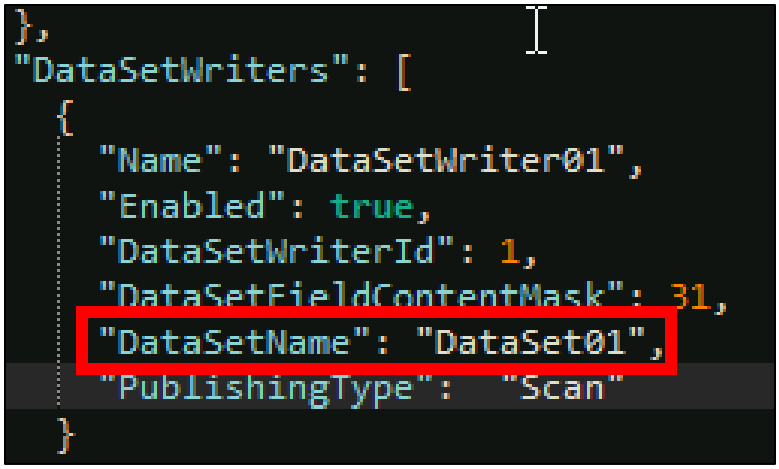
19. Set the *PublishingInterval* (in milli seconds) as required. This is the rate MQTT Publisher will send data to the cloud. For example, if *PublishingInterval* is set to 30000, MQTT Publisher will send data to the cloud every 30 seconds, if data is available to be sent.

20. Locate **DataSetWriter01**.

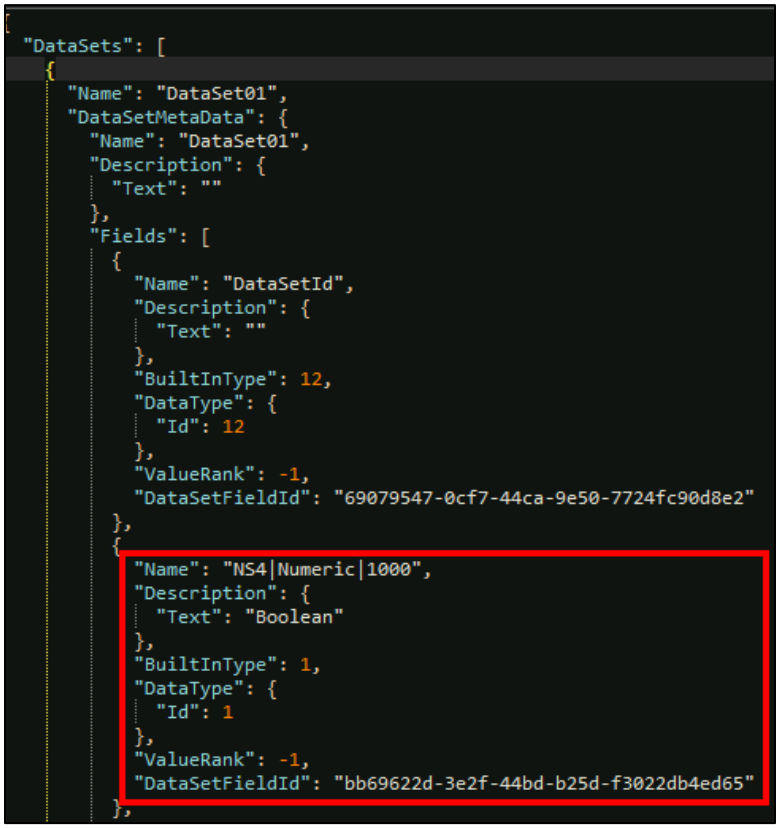
21. Set the *PublishingType* to either **Scan** or **Subscription**. Scan means poll the data source for data regardless of whether the data has changed. Subscription means data will only be received if the value has changed in the data source.



22. Notice the DataSetName **DataSet01**.

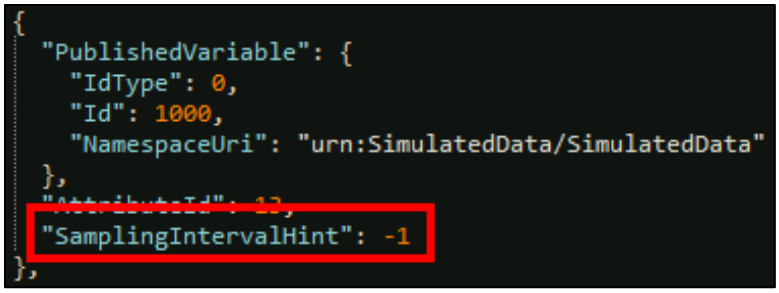


23. Save the file and close.
24. Navigate to **C:\Programdata\Matrikon\MDB-MQTTPublisher\Config\writerdatasets.**
25. Open datasets.json in a text editor.
26. The DataSetMetaData part of the file defines the friendly names of the tags to be published to the cloud. Notice the hook to **DataSet01** as noted in Step 22. Set the name, Description, Text and DataType fields as required.



27. Navigate to the **DataSetSource**. The source of the DataSet is defined here. For example, **NS4|Numeric|1000** as seen in step 26 is defined as having a source of id=1000 and NamespaceUri of urn:SimulatedData/SimulatedData:

28. Set the *SamplingInterval* in milliseconds. SamplingIntervalHint of -1 means the data source will be sampled at the same rate as defined in the PublishingInterval in Step 23.



29. Save and close the file once all required items have been configured.
30. Open Windows Services and restart Matrikon Data Broker – MQTT Publisher
31. Depending on the settings and data generation at the source, data should be published to the cloud shortly after service restart.

For more information on Matrikon Data Broker

Click here to visit the product website.

Contact Matrikon

For more information on Matrikon products and services, visit our website at www.matrikonopc.com

For information on contacting a sales team representative in your region, click here.