



Open Platform Communications United Architecture Ethernet Driver FS-8705-51

Chipkin - Enabling Integration



salesgroup1@chipkin.com

Tel: +1 866 383 1657

© 2021 CHIPKIN AUTOMATION SYSTEMS

Driver Version:
Document Revision: 3

TABLE OF CONTENTS

TABLE OF CONTENTS	2
1 OPC UA DESCRIPTION	4
2 CONNECTION DIAGRAM.....	5
3 OPC UA CONFIGURATION	6
3.1 OPC UA SERVER CONFIGURATION	6
3.2 CREATE CONNECTION.....	7
3.3 CREATE NODE.....	8
3.4 CREATE TASK.....	9
3.5 SAVING THE SERVER CONFIGURATION	10
3.6 RESETTING THE SERVER CONFIGURATION.....	10
3.7 OPC UA CLIENT CONFIGURATION.....	10
3.7.1 <i>Create Connection.</i>	10
3.7.2 <i>Create Node</i>	12
3.7.3 <i>Create Task.....</i>	13
3.7.4 <i>Saving the Client Configuration.....</i>	14
3.7.5 <i>Resetting the Client Configuration</i>	14
4 OPC UA TEST TOOLS.....	15
4.1 UAEXPERT	15
4.2 OPC REFERENCE CLIENT.....	15
5 LICENSE	16
5.1 HOW TO GENERATE A PRE-KEY	16
5.2 HOW TO ACTIVATE A PRODUCT KEY	16
6 FIRMWARE.....	18
6.1 UPLOADING THE OPC UA FIRMWARE	18
6.2 HOW TO DOWNGRADE A SECURE QUICKSERVER TO INSECURE QUICKSERVER	18
7 IMPORTING AND EXPORTING CONFIGURATIONS.....	19
7.1 HOW TO EXPORT THE CONFIGURATION.....	19
7.2 HOW TO IMPORT THE CONFIGURATION	19
7.3 HOW TO IMPORT A PE CONFIGURATION	20
8 REVISION HISTORY	21
APPENDIX A. OPC UA COMMUNICATION FUNCTIONALITY	22
APPENDIX A.1 SUPPORTED FUNCTIONALITY.....	22
APPENDIX A.2 SUPPORTED SERVER DATATYPES• NULL.....	22
APPENDIX A.3 SUPPORTED CLIENT ATTRIBUTES TO READ	22
APPENDIX A.4 DESCRIPTION OF NODEID.....	22
APPENDIX B. TROUBLESHOOTING	23
APPENDIX B.1 DEBUGGING AN OPC UA CONNECTION	23
APPENDIX B.2 USING UAEXPERT FOR TESTING AN OPC UA SERVER	23
APPENDIX B.3 TESTING FIELD SERVER AS AN OPC UA SERVER.....	28

APPENDIX C. EXAMPLE CONFIGURATIONS	31
APPENDIX C.1 EXAMPLE CLIENT CONFIGURATION.....	33
APPENDIX D. MARKETING	35
APPENDIX D.1 CASE STUDY	35
APPENDIX D.2 KEYWORDS	35
APPENDIX E. GLOSSARY OF TERMS	36

1 OPC UA Description

The OPC UA driver allows the FieldServer to transfer data to and from devices over Ethernet using the OPC UA protocol. The FieldServer can emulate either a Server or Client.

If configured as an OPC UA Client, the FieldServer will create socket connections to configured OPC UA Server endpoints. Upon successfully connecting, the driver will create client sessions and using the session, will poll for the requested data points.

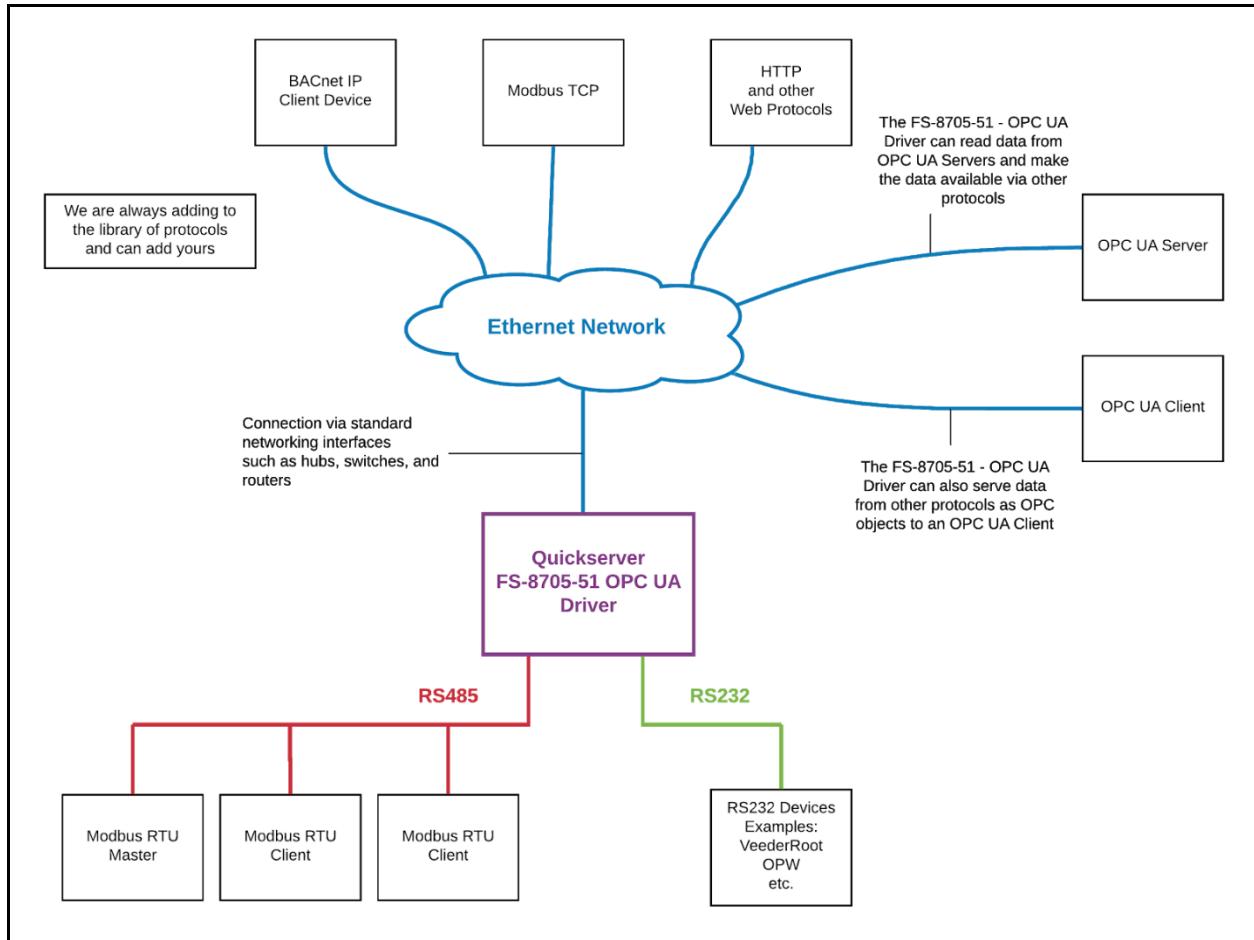
If configured as an OPC UA Server, the FieldServer creates one OPC UA endpoint that other OPC UA Clients can connect to. Nodes are added to OPC Server to store data points and tasks are configured to map data stored in the FieldServer (usually from other drivers) to the Value attribute of the nodes.

In OPC UA, data is stored in nodes. Each node has a unique nodeId which is a combination of a namespace and identifier. See Appendix A.4 for more information on nodeIds. Each node has various attributes, the actual data value and sometimes references to other nodes. For a list of supported data types and attributes, please refer to Append A.2 and A.3 respectively.

The information that follows describes how to expand upon the factory defaults provided in the configuration files included with the FieldServer.

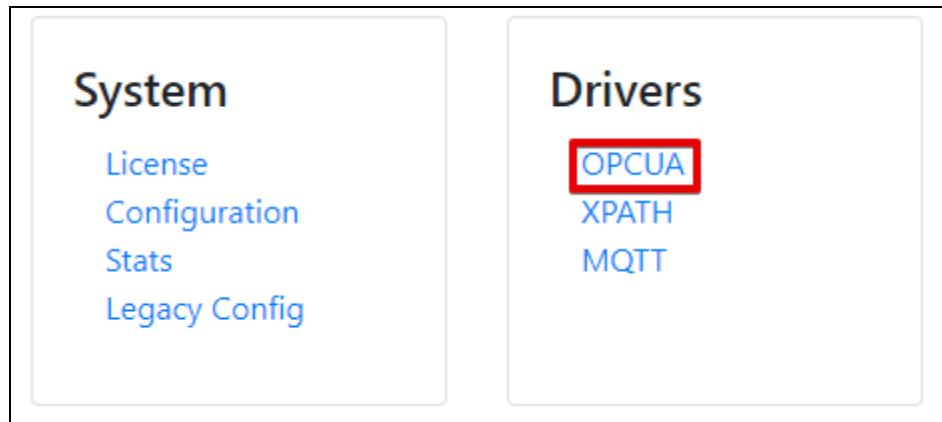
2 Connection Diagram

This block diagram lists common network connections that can monitor and/or serve OPC UA data using other protocols like Modbus® RTU/TCP, BACnet® and HTTP.

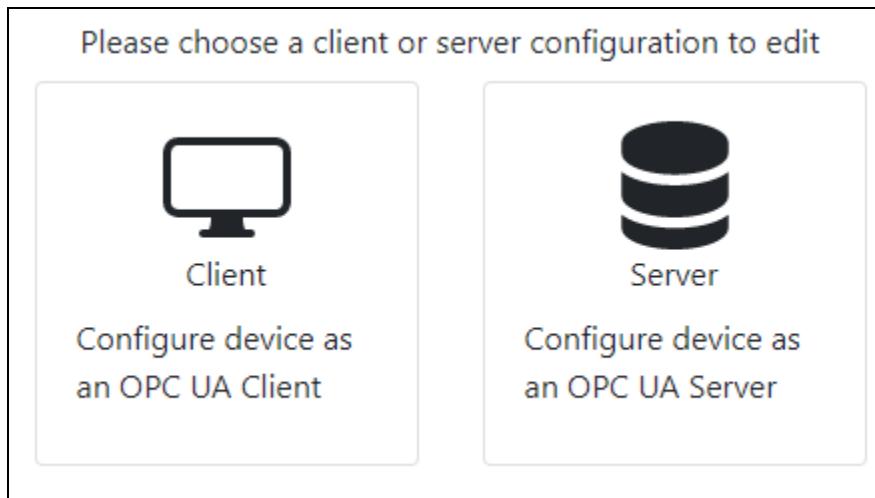


3 OPC UA Configuration

To configure the OPC UA driver, from the home page, click on the OPC UA Driver link or by visiting http://{IP_ADDRESS}/opcuaDriver/ui/



On the OPC Configuration screen, select to configure the Fieldserver as an OPC UA Client or Server.



3.1 OPC UA Server Configuration

This section documents and describes the parameters necessary for configuring the FieldServer to communicate with an OPC UA Client.

The configure the FieldServer as an OPC UA Server, follow the instructions below to add a Connection (contains information about the server), Nodes (UA Objects organized by folders), and finally Tasks (UA Variables that belong to the UA Objects).

3.2 Create Connection

To set up an OPC Server, first create a connection. The connection contains information about this device acting as an OPC Server and how OPC UA Clients will connect to it.

Connections							
Name	Type	Parameters	Port	Resource Path	Manufacturer Name	Product Name	Actions
Create Connection							

1. Click on the “Create Connection” button to open the Create Connection form.
2. Fill out the fields in the form. The fields are as follows:

Column Title	Function	Legal Values
Name	Name of the server, used internally as an identifier	Text, must be unique
Type	The type of connection this is	ethernet
Parameters: Port	The physical port on the FieldServer to use	n1
Port	The TCP listening port for the OPC UA Server	26543 , 4840, available TCP port
Hostname	The IP Address or domain host name of the device running the server.	192.168.1.72 , example-pc
ResourcePath	The URI endpoint for the server. Examples: /UA or /UA/MyDevice. If blank, the uri is simply the root endpoint: opc.tcp://{{IP Address}}:{port}	/UA, /UA/TestServer, any text that specifies a path format
ManufacturerName	The name of the manufacturer of the server	Text, Chipkin Automation Systems
ProductName	The name of the product	Text, CAS FS OPC UA Server

* Bolded values are defaults

3. Click the “Save” button to add the connection.

If successful, the new entry will be populated in the Connections table:

Connections							
Name	Type	Parameters	Port	Resource Path	Manufacturer Name	Product Name	Actions
Test Server	Ethernet		26543	/UA/TestServer	Example Company	Example Product-00XX	<button>Edit</button> <button>Delete</button>
<button>Create Connection</button>							

Note: Only one server connection can exist. If multiple connections are created, only the first one will be used.

3.3 Create Node

Follow the instructions below to add objects to the OPC UA Server.

Nodes				
Connection	Name	Node Id	Folder	Actions
<button>Create Node</button>				

1. Click on the “Create Node” button to open the Create Node form.
2. Fill out the fields in the form. The fields are as follows:

Column Title	Function	Legal Values
Name	The name of the object	Text, must be unique
Connection	The name of the server to add this object.	Text (Use the name of the Connection created in the previous section)
NodId	The nodId to assign to this object. If left blank, the driver will generate one. See Appendix A.4 for description of a nodId.	“ns=1;i=1001”, “ns=2;s=example_object”
Folder	The folder path where this object exists. This is primarily for organizing the objects. If left blank, the object will be placed in the root object folder of the OPC UA server.	Path Examples: “BuildingA/FloorB/RoomC” “MachineRoom1/CabinetX”

3. Click on the “Save” button to add the node.

If successful, the new entry will be populated in the Nodes table:

Nodes				Actions
Connection	Name	Node Id	Folder	
Test Server	Thermostat		BuildingA/FloorB/RoomC	Edit Delete
Create Node				

Repeat the above steps to add additional objects.

3.4 Create Task

Create tasks to add variables to configured objects.

Tasks						Actions
Name	Node	Data Type	Type	Data Broker		
Create Task						

1. Click on the “Create Task” button to open the Create Task form.
2. Fill out the fields in the form. The fields are as follows:

Column Title	Function	Legal Values
Name	The name of the variable to add.	Text, must be unique
Node	The node that this variable belongs to.	Text (Use the name of a node created in the previous section)
DataBroker: Name	The data array in the protocol engine to retrieve the value.	One of the Data Array names
DataBroker: Start	The starting offset in the array to retrieve the value	0 to (“Data_Array_length” - 1)
DataType	The type of data that this data point represents	Refer to Appendix A.2
Type	Select whether this data point is read, read/write, or write only.	Read, Read/Write, Write

Desc	A description of the data point	Text
------	---------------------------------	------

- Click the “Save” button to add the task.

If successful, the new entry will be populated in the Tasks table:

Tasks					
Name	Node	Data Type	Type	Data Broker	Actions
Temperature	Thermostat	Float	read	PE:DA_AI:10	Edit Delete
Set Point	Thermostat	Float	write, read	PE:DA_AI:20	Edit Delete
Create Task					

Repeat the above steps to add additional variables.

3.5 Saving the Server Configuration

When the configuration is complete, click on the “Save Configuration” button to save all of the updates and changes. For the configuration to take effect, reboot the system.



3.6 Resetting the Server Configuration

To clear the configuration and start over, click the “Reset Configuration” button. Then follow the instructions in the sections above to create new connections, nodes, and tasks.

3.7 OPC UA Client Configuration

To configure the FieldServer as an OPC UA Client, follow the instructions below to add a Connection (how to connect to OPC UA Servers), Nodes (OPC UA Client session information), and finally Tasks (data points on the OPC UA Server to read).

3.7.1 Create Connection

Add information on how to connect to one or more OPC UA Servers.

Connections							
Name	Type	Parameters	Endpoint	Initial Delay	Max Delay	Max Retry	Actions
Create Connection							

- Click on the “Create Connection” button to open the Create Connection form.
- Fill out the fields in the form. The fields are as follows:

Column Title	Function	Legal Values
Name	The name of the server to connect to, used internally as an identifier	Text, must be unique
Type	The type of connection	ethernet
Parameters: Port	The physical port on the FieldServer to use	n1
Endpoint	The OPC UA Server endpoint to connect to.	opc.tcp://{{hostname or ip address}}:{port}/{referencePath} Example: opc.tcp://192.168.1.72:26543/UA/TestServer
InitialDelay	Delay in milliseconds before the first attempt to reconnect is made after initial connection failure	0-65535, 1000
MaxDelay	Further reconnection attempts will have an increased delay. This parameter defines the maximum delay between two reconnections	0-65535, 20000
MaxRetry	The maximum number of unsuccessful consecutive retries before the connection fails	0-20, 1

* Bolded values are defaults

- Click the “Save” button to add the connection.

If successful, the new entry will be populated in the Connections table:

Connections							
Name	Type	Parameters	Endpoint	Initial Delay	Max Delay	Max Retry	Actions
ExampleServer	Ethernet		opc.tcp://192.168.1.72:26543/UA/TestServer	1000	20000	1	Edit Delete
Create Connection							

Repeat the above steps to add other servers to poll for data.

3.7.2 Create Node

For each connection, add a node that contains information on how to create an OPC UA Client Session.

Nodes				
Connection	Name	User Name	Password	Actions
Create Node				

1. Click on the “Create Node” button to open the Create Node form.
2. Fill out the fields in the form. The fields are as follows

Column Title	Function	Legal Values
Name	The name of the session, used internally as an identifier	Text, must be unique
Connection	The connection that this session uses	Text (Use the name of a Connection created in the previous section)
UserName	If the session requires a user login, this is the username to use when creating a session. Can be blank if the OPC UA Server endpoint supports anonymous connections	Text
Password	If the session requires a user login, this is the password to use when creating a session. Can be blank if the OPC UA Server endpoint supports anonymous connections	Text

3. Click the “Save” button to add the node.

If successful, the new entry will be populated in the Nodes table:

Nodes				
Connection	Name	User Name	Password	Actions
ExampleServer	MySession			Edit Delete
Create Node				

Repeat the above steps for each connection that was created.

3.7.3 Create Task

Add tasks to poll for specific data points.

Tasks						
Name	Node	Type	Node Id	Attribute	Data Broker	Actions
Create Task						

1. Click on the “Create Task” button to open the Create Task form.
2. Fill out the fields in the form. The fields are as follows:

Column Title	Function	Legal Values
Name	The name of the data point	Text, must be unique
Node	The node that represents the session to use to poll for this data point	Text (Use the name of a Node configured in the previous section)
DataBroker: Name	The data array in the protocol engine to store the value.	One of the Data Array names
DataBroker: Start	The starting offset in the array to store the value	0 to (“Data_Array_length” - 1)
Type	The type of task which defines how the data point is polled	read
ScanInterval	How often in seconds to poll for data	0 - (Max UINT32), 30
NodeId	The nodeId of the data point to poll	“ns=1;i=1001”, “ns=2;s=example_object” See Appendix A.4
Attribute	The attribute of the node to poll	Value , See Appendix A.3 for list of attributes

* Bolded values are defaults

1. Click the “Save” button to add the task.

If successful, the new entry will be populated in the Tasks table:

Tasks						
Name	Node	Type	Node Id	Attribute	Data Broker	Actions
Thermostat	ExampleServer	Read	ns=1;i=1004	Value	PE:DA_Ai:10	<button>Edit</button> <button>Delete</button>
<button>Create Task</button>						

Repeat the above steps to add additional data points to poll.

3.7.4 Saving the Client Configuration

When the configuration is complete, click on the “Save Configuration” button to save all of the updates and changes. For the configuration to take effect, reboot the system.



3.7.5 Resetting the Client Configuration

To clear the configuration and start over, click the “Reset Configuration” button. Then follow the instructions in the sections above to create new connections, nodes, and tasks.



4 OPC UA Test Tools

A list of OPC UA testing tools that you can use to test the functionality of your system.

4.1 UaExpert

A Full-Featured OPC UA Client

<https://www.unified-automation.com/products/development-tools/uaexpert.html>

Features

- OPC UA Data Access View
- OPC UA Alarms & Conditions View
- OPC UA Historical Trend View
- Server Diagnostics View
- Simple Data logger CSV Plugin
- OPC UA Performance Plugin
- GDS Push-Model Plugin

4.2 OPC Reference Client

Official reference client used by the OPC Foundation for certification.

<https://github.com/OPCFoundation/UA-.NETStandard>

<https://github.com/OPCFoundation/UA-.NETStandard/tree/master/Applications/ReferenceClient>

5 License

Some drivers such as OPC UA require a license product key. To generate this license product key a pre-key from the hardware is required.

5.1 How to generate a pre-key

1. Goto the license page http://{IP_ADDRESS}/chipkinLicenseDriver/ui/
2. Click the Get Pre-Key button.

The License Driver allows you to do the following

- Download a prekey to request a product key
- Upload a product key
- View the current enabled drivers

Click the Get Prekey button to download the prekey file

Get PreKey

A pre-key.txt file will be created and downloaded to your system. Send this pre-key.txt and your Job number (FSE1234) to Chipkin support.

5.2 How to activate a product key

Chipkin support can generate a license product key from the hardware pre-key. The product key will be sent as a text file via email.

1. Goto the license page http://{IP_ADDRESS}/chipkinLicenseDriver/ui/
2. Click “Browse” button and select the productkey-XXXXX.txt file provided to you by Chipkin Support.
3. Click the “Upload Product Key” button and wait for the product key to finish uploading.

Upload a product key. Select the product key to upload, then click the Upload Product Key button

Choose a productKey file or drop it here...

Upload Product Key

The list of enabled product codes can be viewed in the “Enabled Product Codes” list.

Enabled Product Codes

The list of product codes that have been enabled by uploaded product keys

No product keys have been uploaded.

6 Firmware

6.1 Uploading the OPC UA Firmware

The OPC UA firmware does not come in the DCC000 default firmware from MSA-Safety. A custom firmware image needs to be loaded to enable the OPC UA driver.

The Quickserver must be in insecure mode before it will accept the OPC UA firmware. Follow the instructions in “How to downgrade a secure quickserver to insecure quickserver” section to downgrade the secure quickserver to insecure quickserver.

Once the quickserver is in insecure mode you can update the firmware through the FSGui using the OPC UA firmware.

6.2 How to downgrade a secure quickserver to insecure quickserver

The OPC UA driver firmware can not be loaded onto a secure quickserver, only an insecure quickserver. To downgrade a secure quickserver to insecure quickserver follow these instructions.

1. Download secure-to-insecure quickserver firmware image https://s.chipkin.com/files/uploads/2020/nov/secure-to-insecure_Chipkin-B0017-1.0.0-beta-armv7.simg
2. Use the Fieldserver toolbox to discover your Quickserver <https://www.sierramonitor.com/content/fieldserver-toolbox-0>
3. Browse to the FSGui firmware update page http://{IPADDRESS}/htm/fsgui.htm#20_OID
Setup => File Transfer, then click the Firmware tab on the right.
4. Click the “Choose File” button and select the secure-to-insecure quickserver firmware image “secure-to-insecure_Chipkin-B0017-1.0.0-beta-armv7.simg” then click the “Submit” button
5. Wait for the firmware to be uploaded. Then click the “System Reboot” button.
Note: System Reboot, not System Restart.

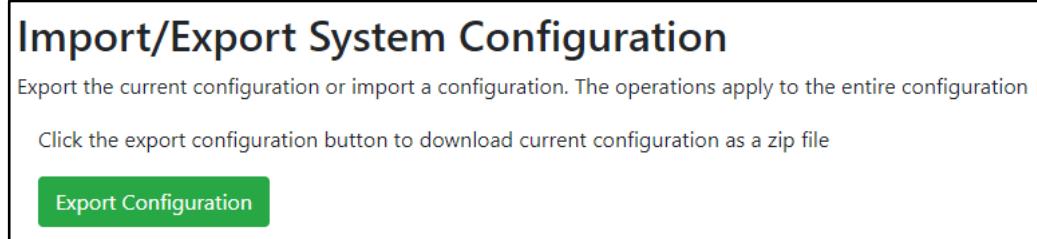
The Quickserver should now be in insecure mode.

7 Importing and Exporting Configurations

It is possible to export the current configuration to back it up or simply to make some edits. Users can also import either the entire configuration via a zip file or a PE (Protocol Engine) configuration.

7.1 How to Export the Configuration

1. Goto the system configuration page http://{IP_ADDRESS}/chipkinConfiguration/ui/
2. Click the Export Configuration button.



7.2 How to Import the Configuration

The file to import the configuration must be a zip file. The zip file should contain the following folders:

- ae - this folder contains any configuration files for the ae configuration
- documents - this folder contains any driver specific documents. For example, license product keys, etc.
- pe - this folder contains one config.csv file for the pe configuration.

To make sure the folder directory is correct, do an Export first, then extract the files, edit them, then zip them up again.

To import the configuration:

1. Goto the system configuration page http://{IP_ADDRESS}/chipkinConfiguration/ui/
2. Click the “Browse” button in the “Import/Export System Configuration” section and select the zip file containing the configuration to import.
3. Click the “Import Configuration” button and wait for the configuration to finish importing.
4. If successful, a success message will appear prompting a reboot of the Fieldserver for the changes to take effect.

Import/Export System Configuration

Export the current configuration or import a configuration. The operations apply to the entire configuration

Click the export configuration button to download current configuration as a zip file

Export Configuration

Import a configuration zip file. Select the file to import, then click the Import Configuration

Choose a configuration zip file or drop it here... **Browse**

Import Configuration

7.3 How to Import a PE Configuration

It is possible to import a PE (Protocol Engine) configuration separately. To import a PE configuration:

1. Goto the system configuration page http://{IP_ADDRESS}/chipkinConfiguration/ui/
2. Click the “Browse” button in the “Import Specific Configuration” section and select the csv file containing the pe configuration to import.
3. Click the “Import PE Configuration” button and wait for the configuration to finish importing.
4. If successful, a success message will appear prompting a reboot of the Fieldserver for the changes to take effect.

Import Specific Configuration

Use the following control to import a specific portion of the configuration, this includes:

- Import PE Configuration

Import a pe configuration csv file. Select the file to import, then click the Import PE Configuration

Choose a pe config.csv file or drop it here... **Browse**

Import PE Configuration

8 Revision History

This table summarizes the update history for this document. Please contact Chipkin for an updated version of this document if required.

Date	Agent	Revision	Comments
17 Nov 2020	SWS	1	Created initial document
20 Nov 2020	AF	2	Added OPC UA Quickstart section.
20 May 2021	YC	3	Updated document format

Appendix A. OPC UA Communication Functionality

Appendix A.1 Supported Functionality

Client:

- Connect (using opc.tcp or https)
- CreateSession (with or without username and password)
- Read

Server:

- Read
- Write

Appendix A.2 Supported Server Datatypes

- Boolean
- SByte
- Byte
- Int16
- UInt16
- Int32
- UInt32
- Int64
- UInt64
- Float
- Double
- String
- DateTime
- Guid
- ByteString

Appendix A.3 Supported Client Attributes to Read

The following is a list of attributes that can be read from an OPC UA object. The most common one will be the 'Value' attribute, but the OPC UA driver supports reading from the other attributes as well.

- NodeId
- NodeClass
- BrowseName
- DisplayName
- Description
- WriteMask
- UserWriteMask
- IsAbstract
- Symmetric
- InverseName
- ContainsNoLoops
- EventNotifier
- Value
- DataType
- ValueRank
- ArrayDimensions
- AccessLevel
- UserAccessLevel
- MinimumSamplingInterval
- Historizing
- Executable

Appendix A.4 Description of NodeId

To Do

Appendix B. Troubleshooting

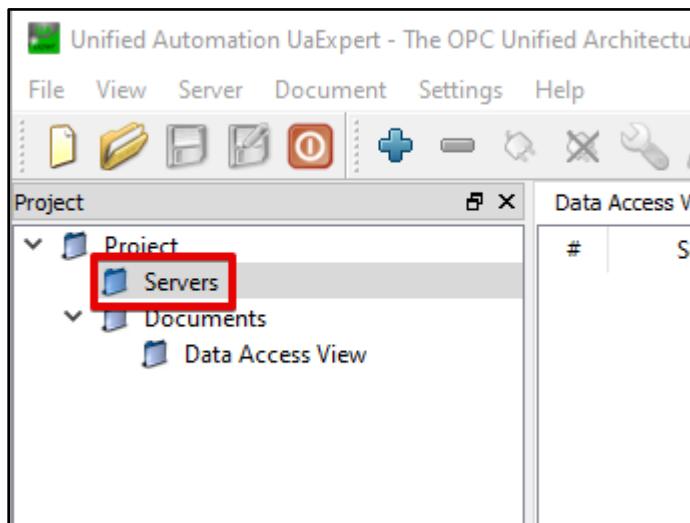
Appendix B.1 Debugging an OPC UA connection

- If connections are not working properly, confirm the OPC UA Server Endpoint. Depending on how it has been configured, try using both the IP Address and the HostName. For example:
 - IP Address format: opc.tcp://192.168.1.72:26543/UA/TestServer
 - HostName format: opc.tcp://example-pc:26543/UA/TestServer
- Verify that the url contains the opc.tcp://. If it is missing, the connection will fail.
- Verify the resource path part of the endpoint. These are very specific and case sensitive. Some servers do not specify a resource path. See the following examples:
 - No resource path: opc.tcp://192.168.1.72:26543/
 - With resource path: opc.tcp://example-pc:26543/UA/
 - With longer path: opc.tcp://192.168.1.72:26543/UA/TestServer
- When using the FieldServer as an OPC UA Server, ensure that the FieldServer is on the same subnet as the OPC UA Client that is attempting to poll for data.

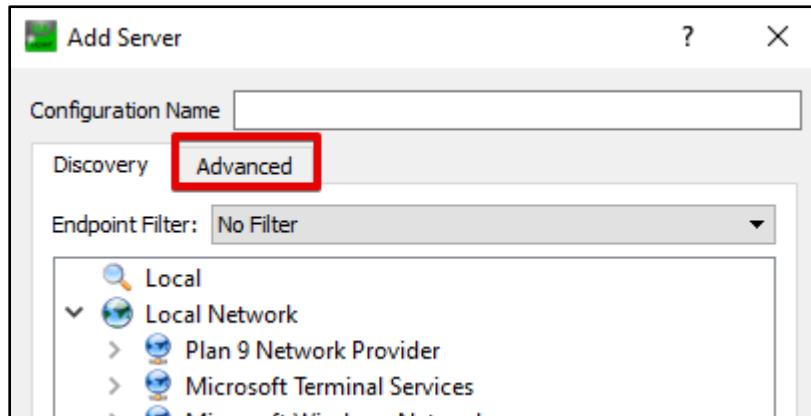
Appendix B.2 Using UaExpert for Testing an OPC UA Server

Follow the steps in this section to setup the UaExpert tool to test a connection to an OPC UA Server (either the FieldServer configured as an OPC UA Server or to test an existing server to see what data is on it).

1. Download and install the tool: <https://www.unified-automation.com/products/development-tools/uaexpert.html>
2. In the tool, right-click on the Servers folder and click the Add option.

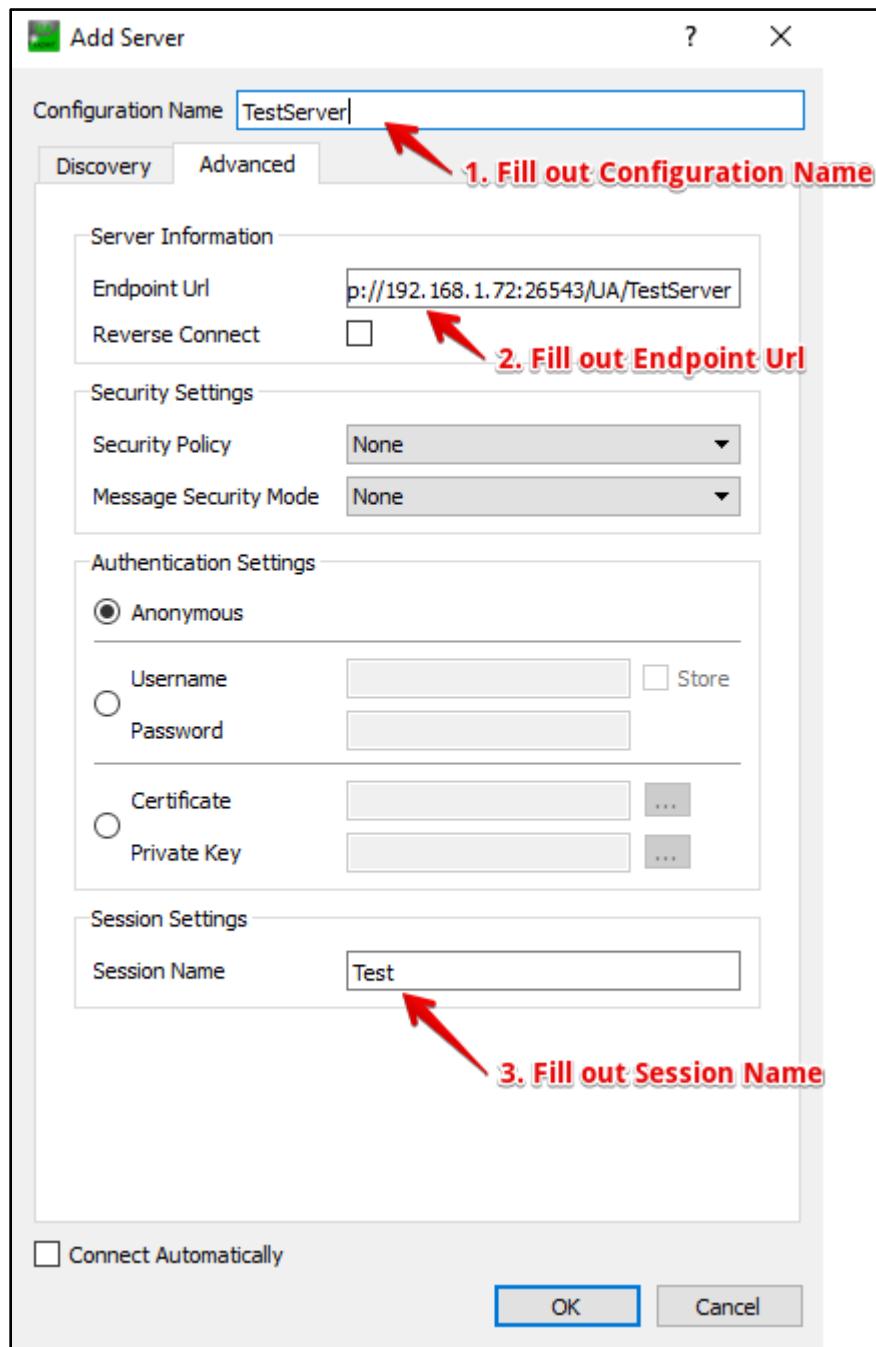


3. In the tool, right-click on the Servers folder and click the Add option.

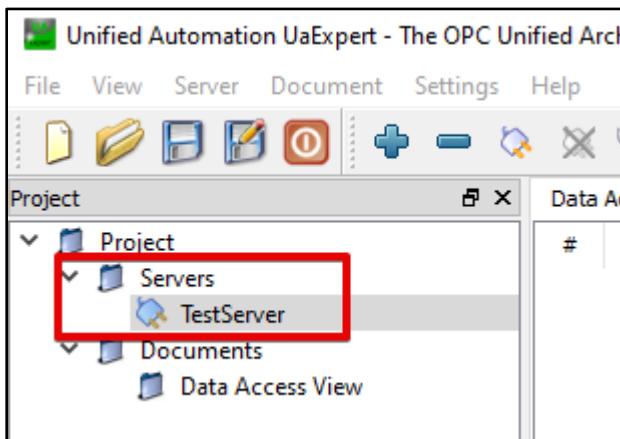


4. Fill out the following fields:

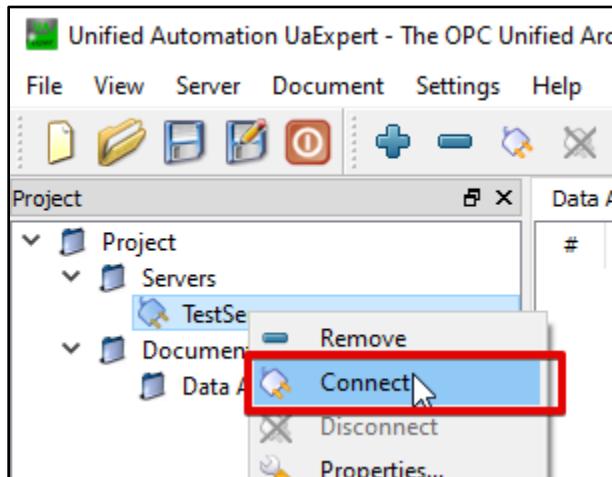
- Fill out the Configuration Name. This is a generic name to save this configuration
- Fill out the Endpoint Url. This should match the OPC UA Server endpoint.
- Fill out the Session Name. This is a generic name for the session.



5. Click the “Ok” button to add it to the Server list



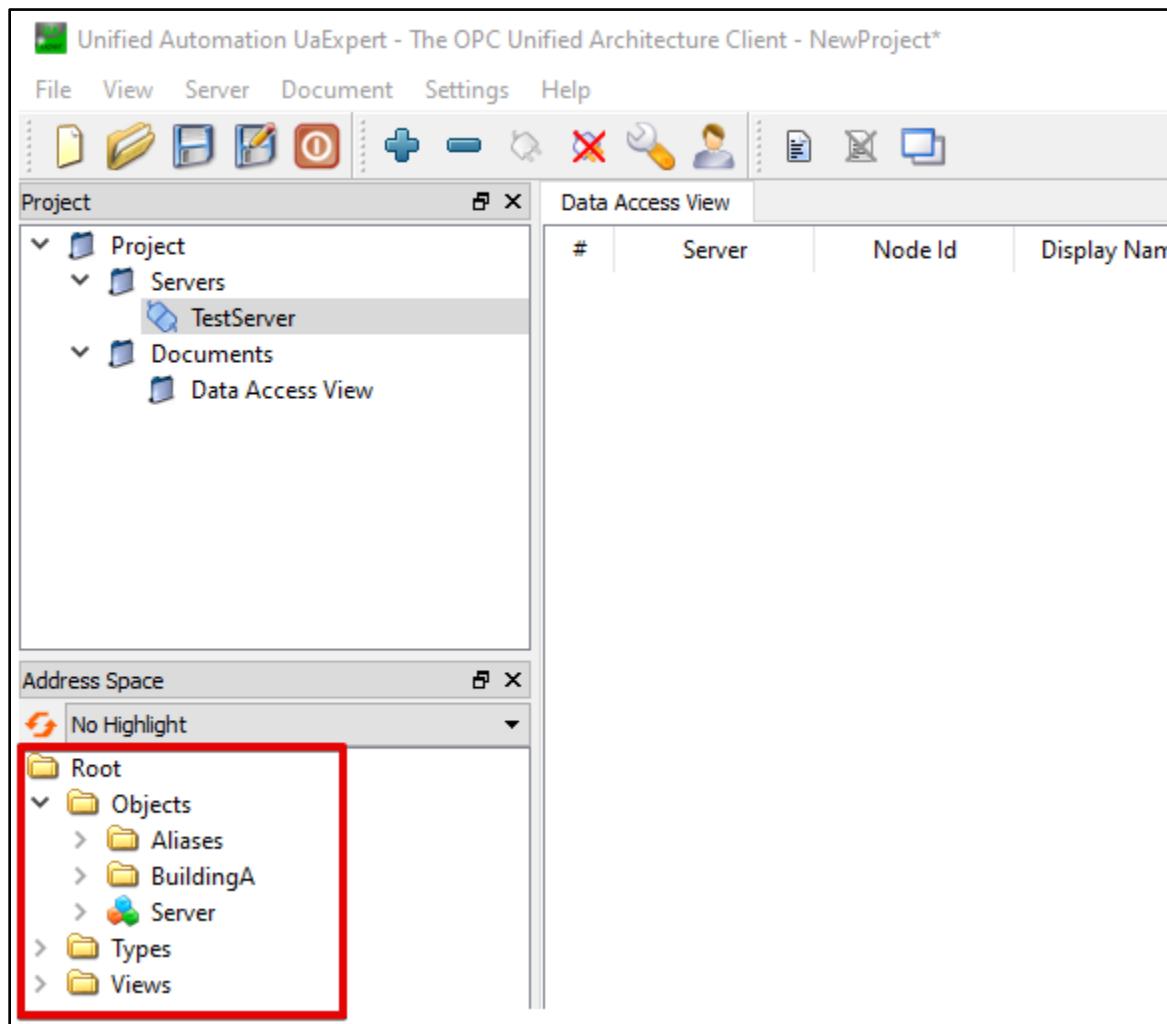
6. Right-click the newly added server and click connect



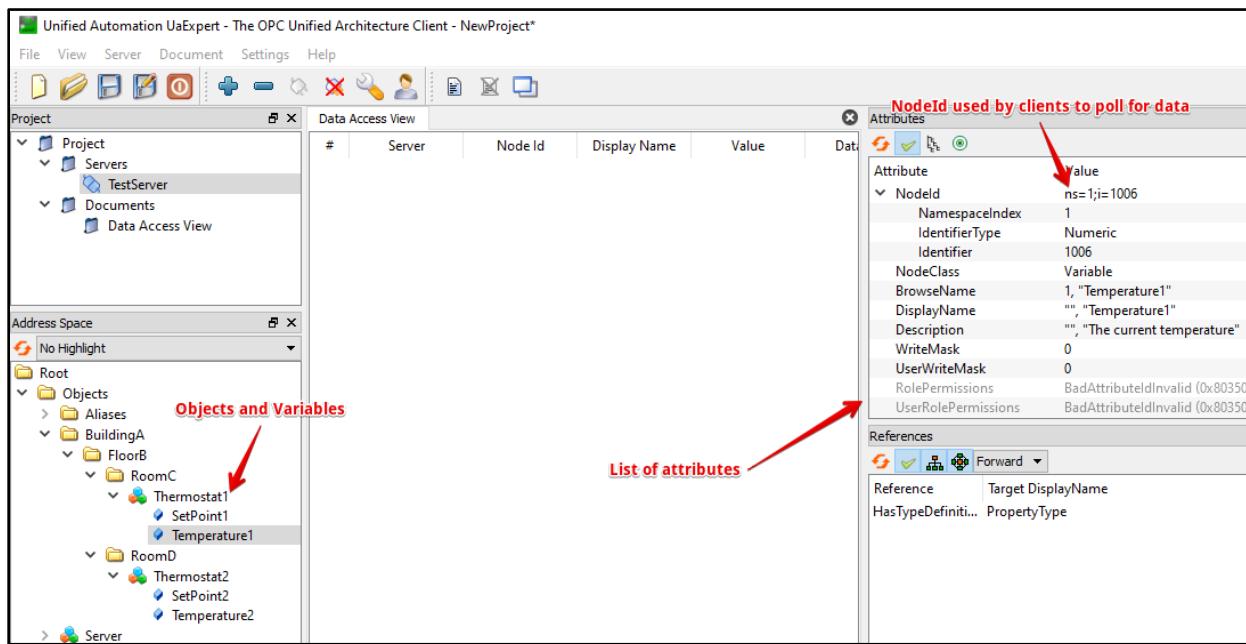
7. If there is an error, it will be displayed in the Log at the bottom of the screen. Here is an example of a BadCommunicationError:

2020-11-20 8:38:52.467 AM	General	[uastack] OpcUa_SecureConnection_OnNotify: Connect event: ERROR 0x80050000!
2020-11-20 8:38:52.467 AM	General	[uastack] OpcUa_SecureConnection_OnNotify: Connect event: Notifying owner! 0x80050000
2020-11-20 8:38:52.467 AM	Server Node	TestServer Could not connect to server: BadCommunicationError

8. If successful, the OPC UA Server object tree will be populated in the Address Space section. You may see a Certificate Validation dialog box appear. Review the information and Accept the server certificate temporarily for this session.



9. Open up any folders to see the objects that are available. You can see all the properties of the objects and attributes in the far right screens labelled Attributes and References.



10. Drag and drop variables to the Data Access View screen in the middle to check data values.

#	Server	Node Id	Display Name	Value	Datatype	Source Timestamp	Server Timestamp	Statuscode
1	TestServer	NS1 Numeric 1006	Temperature1	10	Float	8:42:16.570 AM	8:46:55.507 AM	Good
2	TestServer	NS1 Numeric 1007	SetPoint1	11	Float	8:42:18.100 AM	8:46:56.821 AM	Good
3	TestServer	NS1 Numeric 1008	Temperature2	20	Float	8:42:19.956 AM	8:46:57.972 AM	Good
4	TestServer	NS1 Numeric 1009	SetPoint2	21	Float	8:42:21.635 AM	8:46:58.843 AM	Good

Appendix B.3 Testing Fieldserver as an OPC UA Server

The following instructions are how to confirm that a FieldServer that has been configured as an OPC UA Server is working correctly.

1. Follow the instructions in Appendix B.2 to use UaExpert to connect to the FieldServer configured as an OPC UA Server.
2. Access the Data Array page on the FieldServer interface

The screenshot shows the UaExpert interface. On the left, the navigation tree is expanded to show 'View' > 'Data Arrays' > 'DA_AE_LOADER' > 'DA_AI'. This section is highlighted with a red box. The main panel on the right is titled 'DA_AI' and contains two tabs: 'Data Array' and 'Data Array Attrib'. The 'Data Array' tab displays a table of data items:

Offset	0	1	2	3	4
0	0.000000	0.000000	0.000000	0.000000	0.000000
10	10.000000	11.000000	0.000000	0.000000	0.000000
20	20.000000	21.000000	0.000000	0.000000	0.000000
30	0.000000	0.000000	0.000000	0.000000	0.000000

The 'Data Array Attrib' tab shows the following properties:

Name	Value
Data Array Name	DA
Data Format	Fl
Length in Items	20
Bytes per Item	4
Data Age	0.1

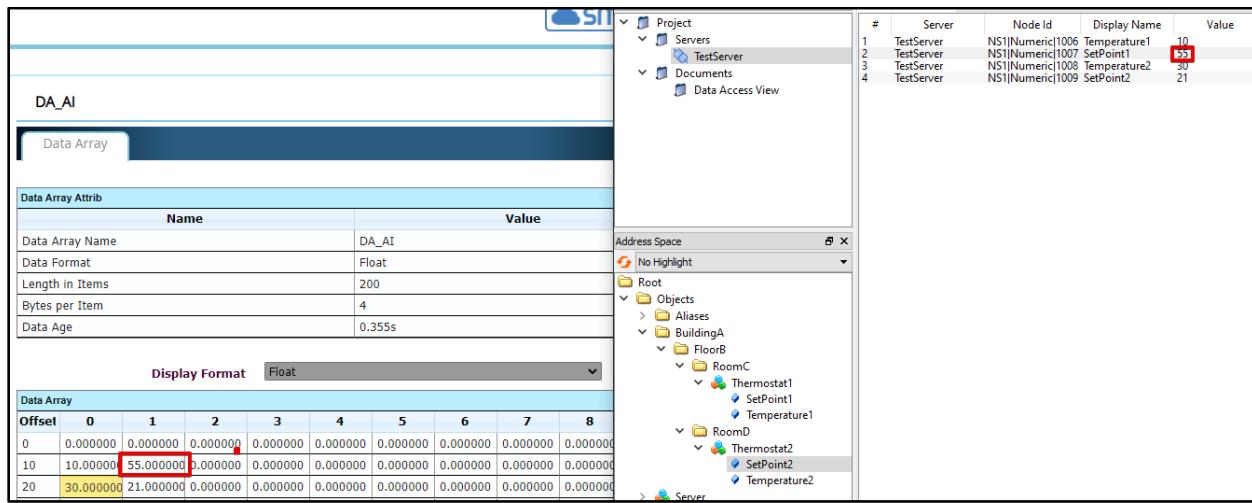
3. Change Values in the Data array and see the values updated in the UaExpert

The screenshot shows the UaExpert interface after modifying the Data Array. The 'Data Array' tab now displays the following data:

Offset	0	1	2	3	4	5	6	7	8
0	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
10	10.000000	11.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
20	30.000000	21.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000

The 'Address Space' tree on the right shows the hierarchical structure of objects, including 'Root' > 'Objects' > 'BuildingA' > 'FloorB' > 'RoomC' > 'Thermostat1' and 'RoomD' > 'Thermostat2', with their respective 'SetPoint1', 'Temperature1', 'SetPoint2', and 'Temperature2' nodes.

4. Change a value for a read/write or write only point in the UaExpert and see the value updated in the Data Array interface



Appendix C. Example Configurations

```
{  
  "ae": {  
    "OPCUAServer": {  
      "connections": [  
        {  
          "type": "ethernet",  
          "name": "TestServer",  
          "port": 26543,  
          "hostname": "192.168.2.168",  
          "resourcePath": "/UA/TestServer",  
          "manufacturerName": "Example Company",  
          "productName": "Example Product"  
        }  
      ],  
      "nodes": [  
        {  
          "name": "Thermostat1",  
          "connection": "TestServer",  
          "folder": "BuildingA/FloorB/RoomC"  
        },  
        {  
          "name": "Thermostat2",  
          "connection": "TestServer",  
          "folder": "BuildingA/FloorB/RoomD"  
        }  
      ],  
      "tasks": [  
        {  
          "dataType": "Float",  
          "path": "BuildingA/FloorB/RoomC/Thermostat1/CurrentTemperature",  
          "interval": 10000  
        }  
      ]  
    }  
  }  
}
```

```
"type": ["read"],  
"name": "Temperature1",  
"node": "Thermostat1",  
"dataBroker": { "pe": { "Name": "DA_AI", "Start": "10" } },  
"desc": "The current temperature"  
,  
{  
    "dataType": "Float",  
    "type": ["read", "write"],  
    "name": "SetPoint1",  
    "node": "Thermostat1",  
    "dataBroker": { "pe": { "Name": "DA_AI", "Start": "11" } },  
    "desc": "Setpoint for temperature"  
,  
{  
    "dataType": "Float",  
    "type": ["read"],  
    "name": "Temperature2",  
    "node": "Thermostat2",  
    "dataBroker": { "pe": { "Name": "DA_AI", "Start": "20" } },  
    "desc": "The current temperature"  
,  
{  
    "dataType": "Float",  
    "type": ["read", "write"],  
    "name": "SetPoint2",  
    "node": "Thermostat2",  
    "dataBroker": { "pe": { "Name": "DA_AI", "Start": "21" } },  
    "desc": "Setpoint for temperature"  
}
```

```
    ]
}

}

}
```

Appendix C.1 Example Client Configuration

```
{
  "ae": {
    "OPCUAClient": {
      "connections": [
        {
          "type": "ethernet",
          "name": "QuickReferenceServer",
          "endpoint": "opc.tcp://localhost:62541/Quickstarts/ReferenceServer",
          "initialDelay": 1000,
          "maxDelay": 2000,
          "maxRetry": 1
        }
      ],
      "nodes": [{ "name": "MySession", "connection": "QuickReferenceServer" }],
      "tasks": [
        {
          "type": "read",
          "scanInterval": 30,
          "attribute": "Value",
          "name": "ScalarStaticFloat",
          "node": "MySession",
          "dataBroker": { "pe": { "Name": "DA_AI", "Start": "10" } },
          "nodeId": "ns=2;s=Scalar_Static_Float"
        },
      ],
    }
  }
}
```

```
{  
    "type": "read",  
    "scanInterval": 30,  
    "attribute": "Value",  
    "name": "ScalarStaticInt16",  
    "node": "MySession",  
    "dataBroker": { "pe": { "Name": "DA_AI", "Start": "11" } },  
    "nodeId": "ns=2;s=Scalar_Static_Int16"  
}  
]  
}  
}  
}
```

Appendix D. Marketing

Appendix D.1 Case Study

A series of case studies for OPC can be found here

<https://opcfoundation.org/resources/case-studies/>

Appendix D.2 Keywords

OPC, OPC UA, OPC-UA Server, OPC UA Client, OPC Unified Architecture, OPC technologies, Open Platform Communications, Industry 4.0

Appendix E. Glossary of Terms

- **SOA** - Service-oriented architecture
- **OPC** - Open Platform Communications. The acronym OPC was borne from OLE (object linking and embedding) for Process Control.
- **OPC-UA** - Unified Architecture
- **OPC-Classic** - The OPC standard that was restricted to the Windows operating system.
- **PubSub** - Publish - Subscribe. A mechanism for data and event notification
- **UACTT** - OPC UA Compliance Test Tool <https://opcfoundation.org/developer-tools/certification-test-tools/opc-ua-compliance-test-tool-uactt/>
- **ERP** - Enterprise Resource Planning
- **MES** - Manufacturing Execution Systems