https://www.halvorsen.blog

OPC UA in Python

Hans-Petter Halvorsen

Contents

Introduction

Table of Contents

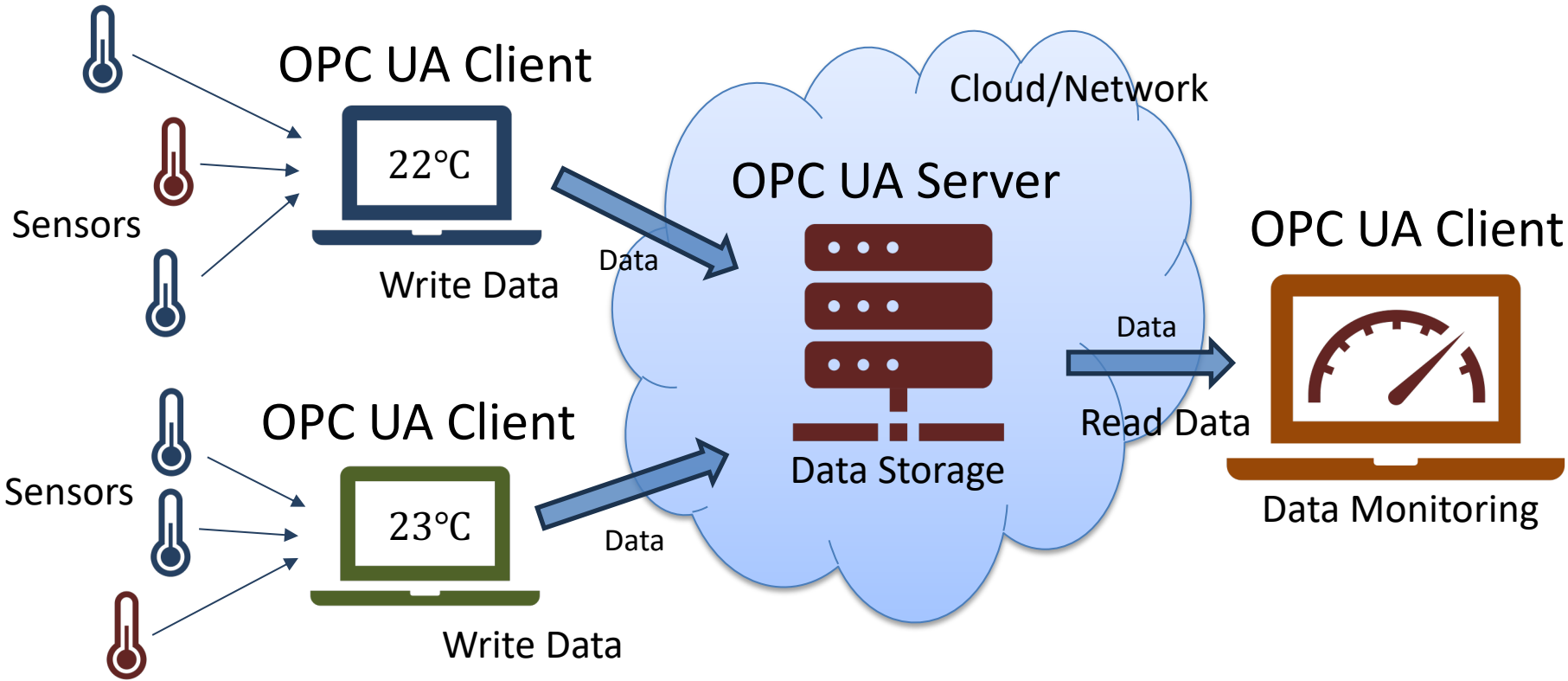
Hans-Petter Halvorsen

Introduction

- This Tutorial will use the OPC UA protocol to send data between different devices or different programs.
- OPC UA is a communication protocol.
- OPC UA is mainly used for industrial purposes (industrial automation), and it is one of the most used protocols in the industry today.
- OPC UA is maintained by: https://opcfoundation.org.
- Python is a widely used multi-purpose Programming language and one of the most used programming languages today.
- Python is maintained by: https://www.python.org.

OPC UA Example

We will create different OPC UA Clients and an OPC UA Server in this Tutorial



Introduction

- There exists lots of different OPC packages and libraries for Python.
- This Tutorial will demonstrate OPC UA in Python with some basic Code Examples using the “**opcua**” Python package.
- This Tutorial will use the **Thonny** Python Editor and the **Spyder** Python Editor, but other Python Editors can of course also be used.
- This Tutorial will use the **OPC UA Server Simulator** from Integration Objects (free OPC UA server for Test and Demonstration purposes), but other OPC UA Servers can of course also be used.
- We will create both OPC UA Clients and an OPC UA Server using Python and the “**opcua**” Python package.

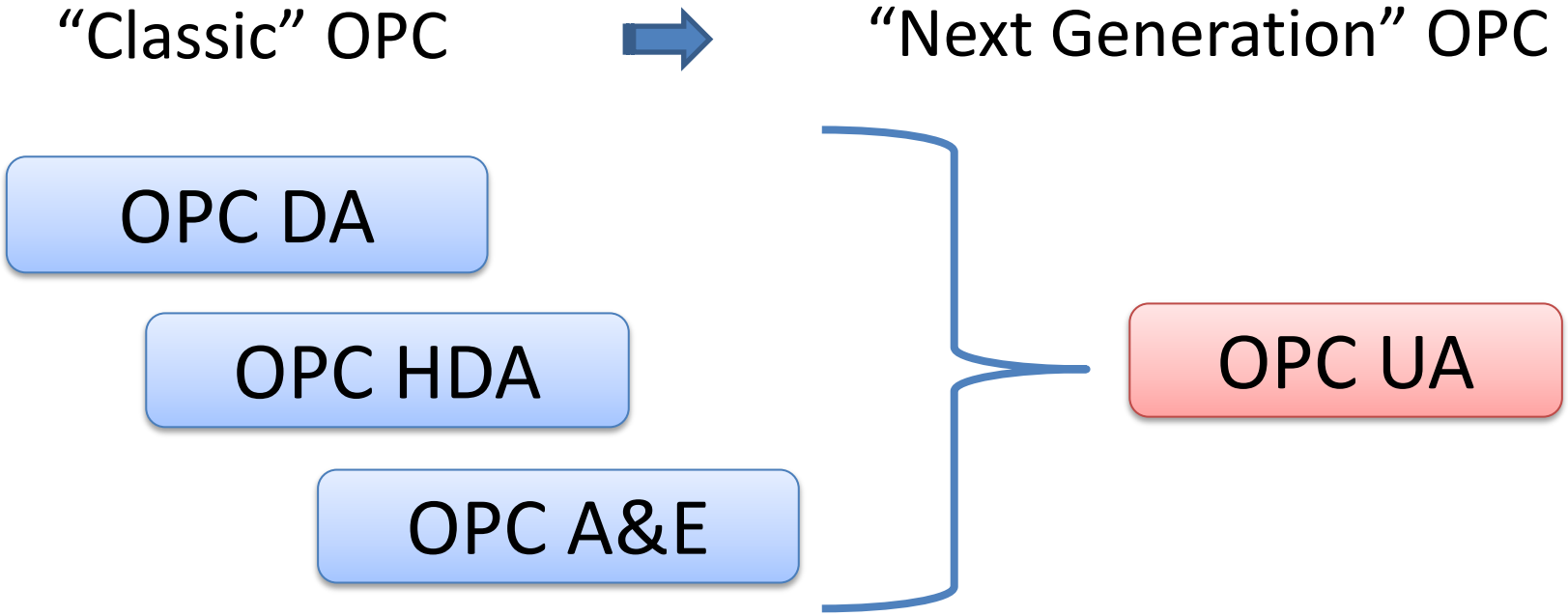
Software

The following software is used in this Tutorial:

- Thonny Python Editor
  https://thonny.org
- Spyder Python Editor. You can download it separately or part of Anaconda Python Distribution
  https://www.spyder-ide.org
  https://www.anaconda.com
- Python package/library opcua
  https://pypi.org/project/opcua/
- OPC UA Server Simulator:
  https://integrationobjects.com/sioth-opc/sioth-opc-unified-architecture/opc-ua-server-simulator/
- UaExpert OPC UA Client:
  https://www.unified-automation.com/products/development-tools/uaexpert.html

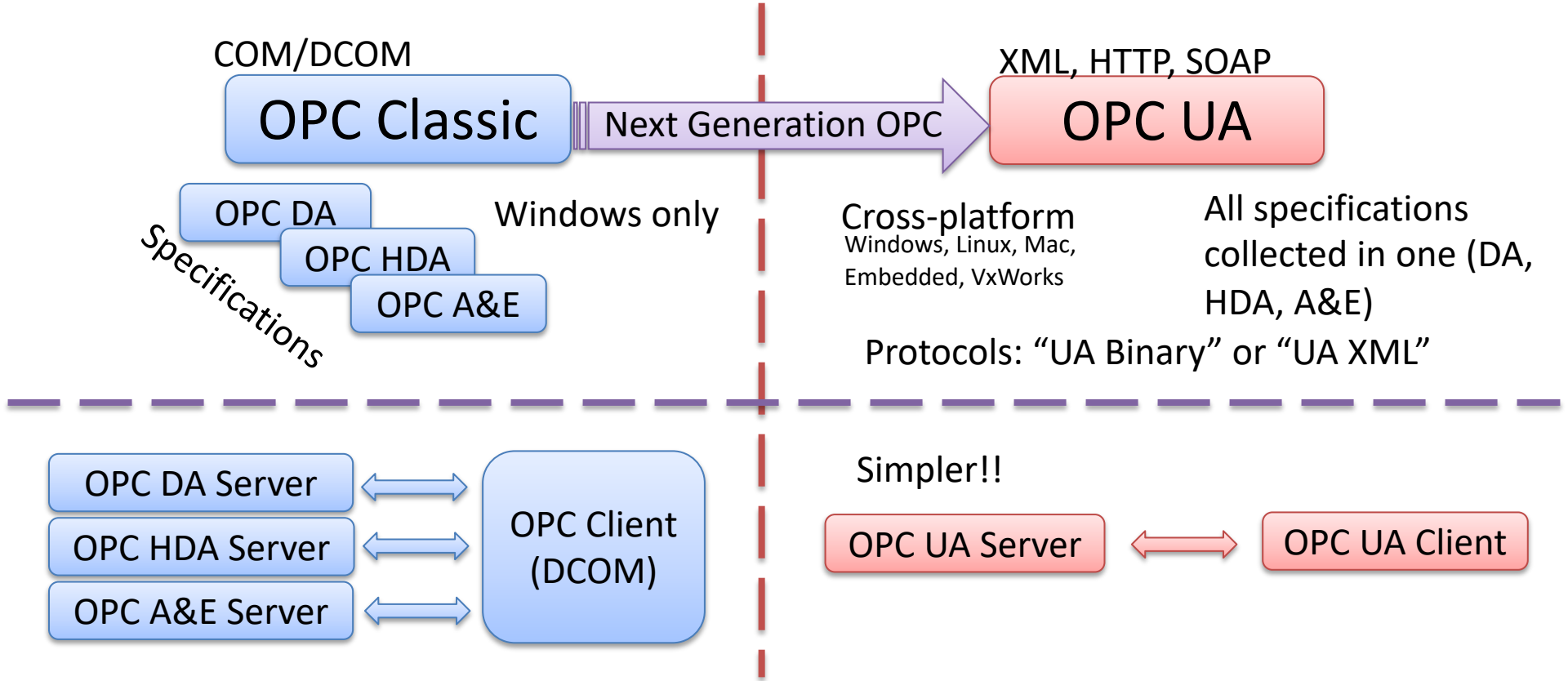
OPC Specifications

OPC is a standard that defines the communication of data between devices from different manufactures. OPC requires an **OPC Server** that communicates with one or more **OPC Clients**



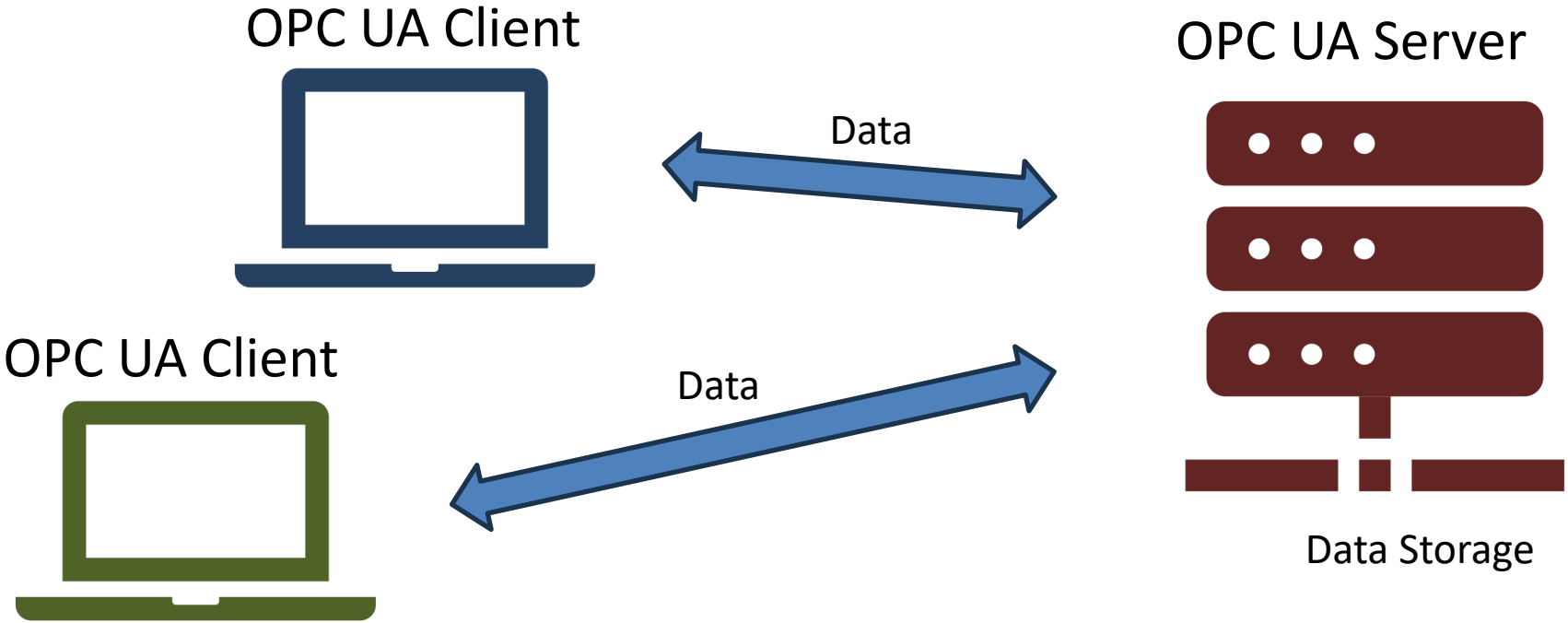
https://opcfoundation.org/about/opc-technologies/opc-classic/

Next Generation OPC

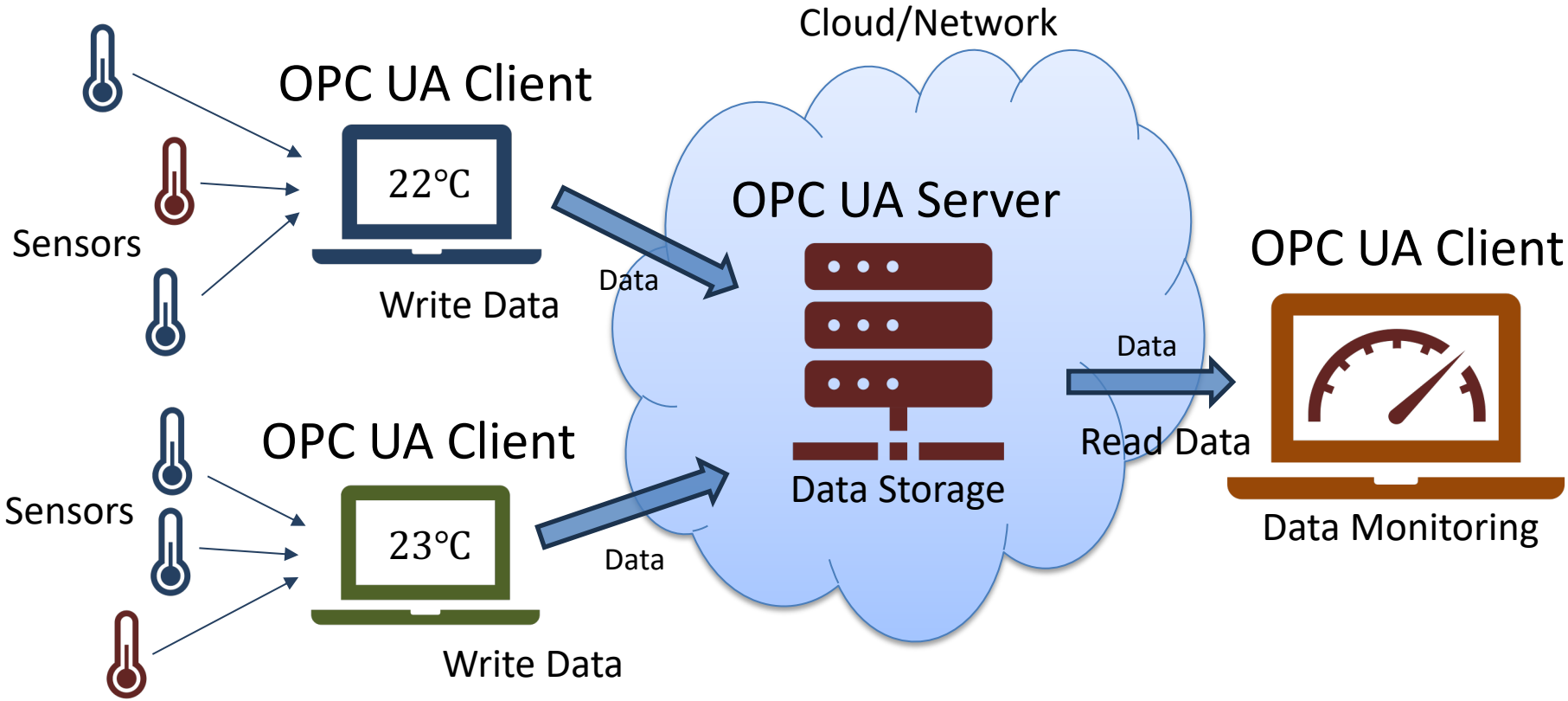


OPC UA – Server and Clients

OPC is a standard that defines the communication of data between devices from different manufactures. OPC requires an **OPC Server** that communicates with one or more **OPC Clients**.



OPC UA Scenario



OPC UA Scenario

Programmable Automation Controller

PLC

Programmable Logic Controller

PLC

PAC

Programmable Automation Controller

OPC Server

HMI

DCS

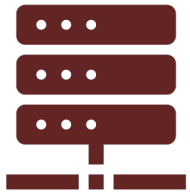
Distributed Control System

SCADA

Supervisory Control And Data Acquisition

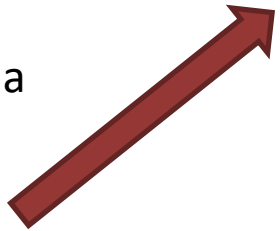
OPC UA Examples in this Tutorial

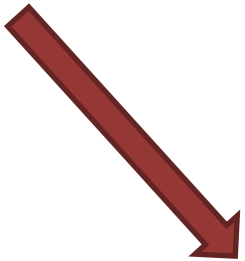
OPC UA Server



In this Tutorial we will create both OPC Clients and and an OPC Server where we use the “opcua” Python package

Normally, the OPC Server and the different OPC Clients runs on different computers distributed in a network

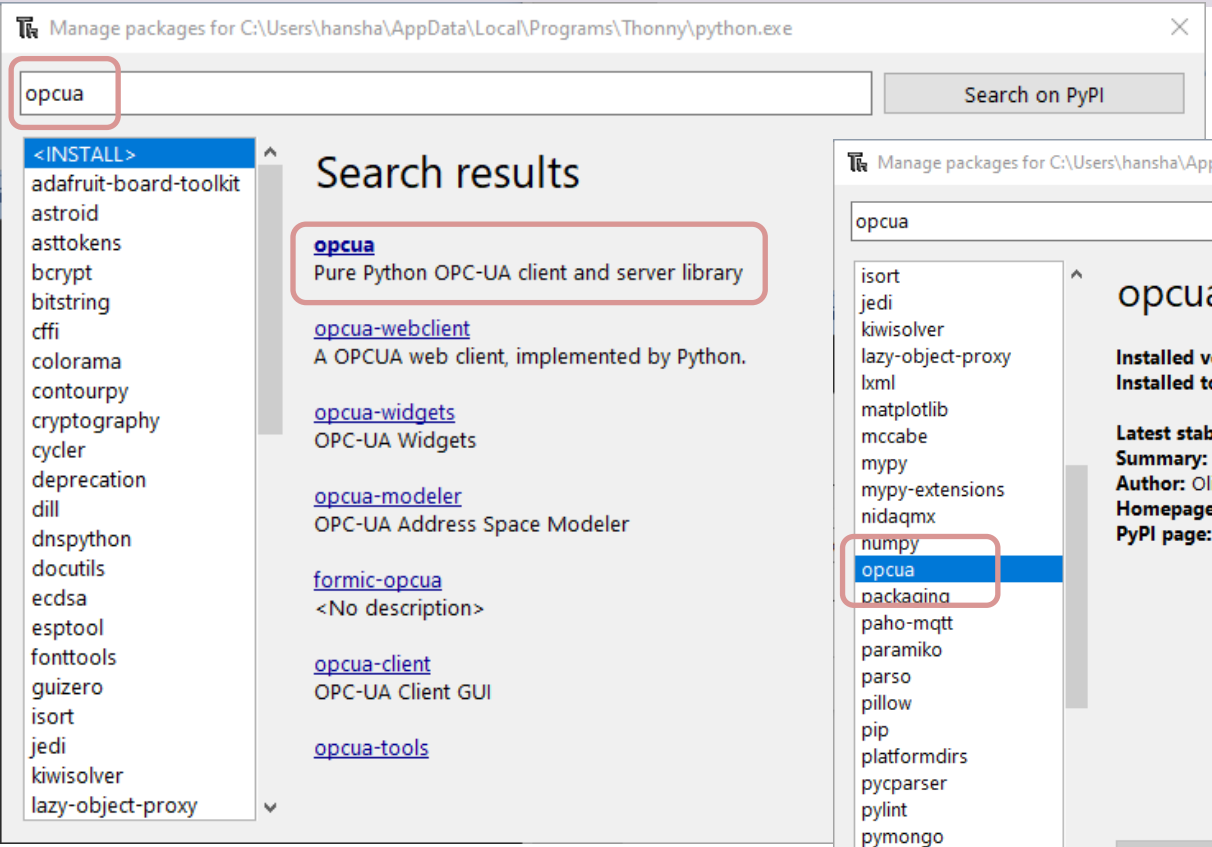


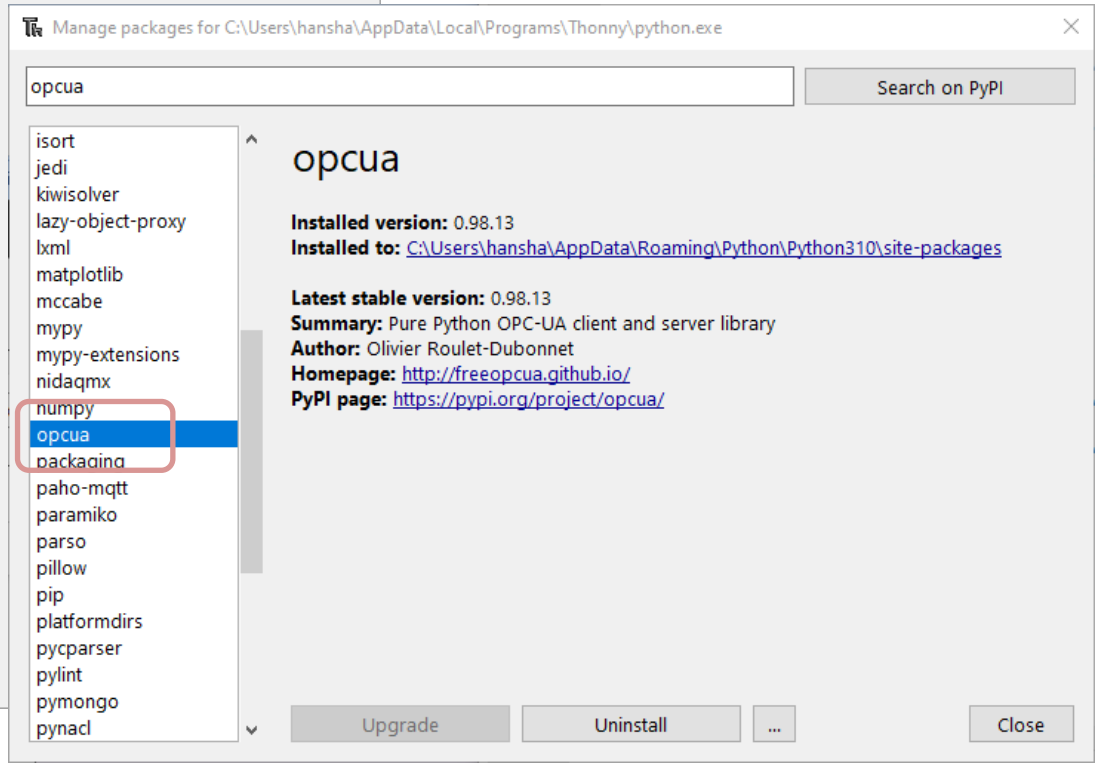


OPC UA Write Client

OPC UA Read Client

Installation of “opcua” Thonny

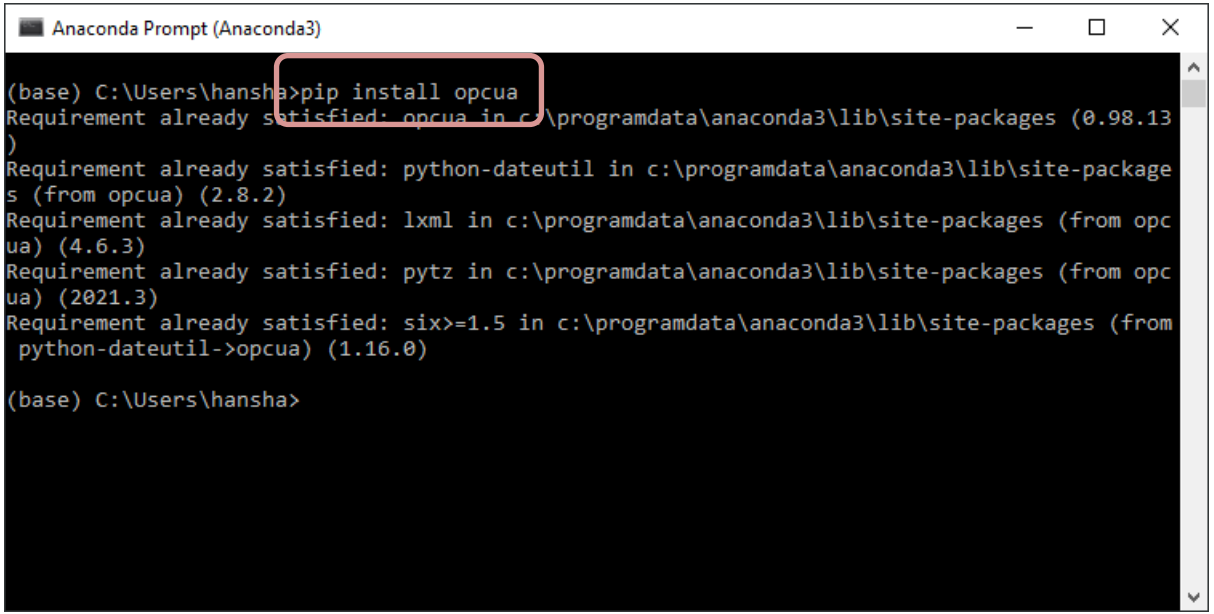




Installation of “opcua” Anaconda

You can use the “**Anaconda Prompt**” that comes with the Anaconda Distribution

pip install opcua



https://www.anaconda.com/download

https://pypi.org/project/opcua/

References

- PyPi: https://pypi.org/project/opcua/
- Python OPC-UA Documentation: https://python-opcua.readthedocs.io/en/latest/
- GitHub: https://github.com/FreeOpcUa/python-opcua/tree/master/examples
- GitHub Client GUI: https://github.com/FreeOpcUa/opcua-client-gui

OPC UA Simulation Server

Integration Objects

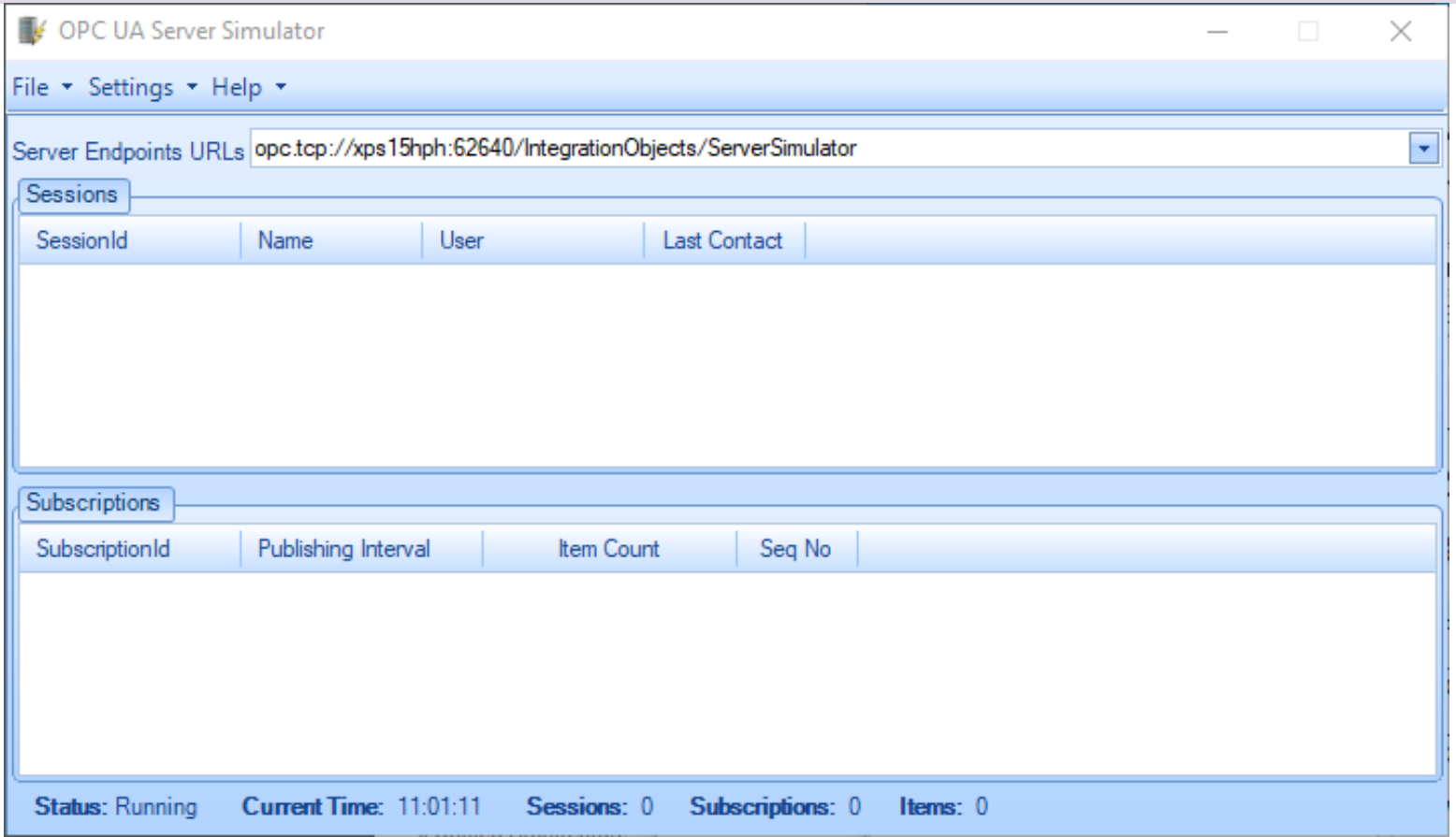
Hans-Petter Halvorsen

Table of Contents

OPC UA Server Simulator

- “OPC UA Server Simulator” from Integration Objects is a free OPC UA Server
- You can use it for Test and Demonstration purposes.
- It runs 48 hours before you need to restart it
- Users can configure their own OPC Tags via CSV files
- Download: https://integrationobjects.com/sioth-opc/sioth-opc-unified-architecture/opc-ua-server-simulator/

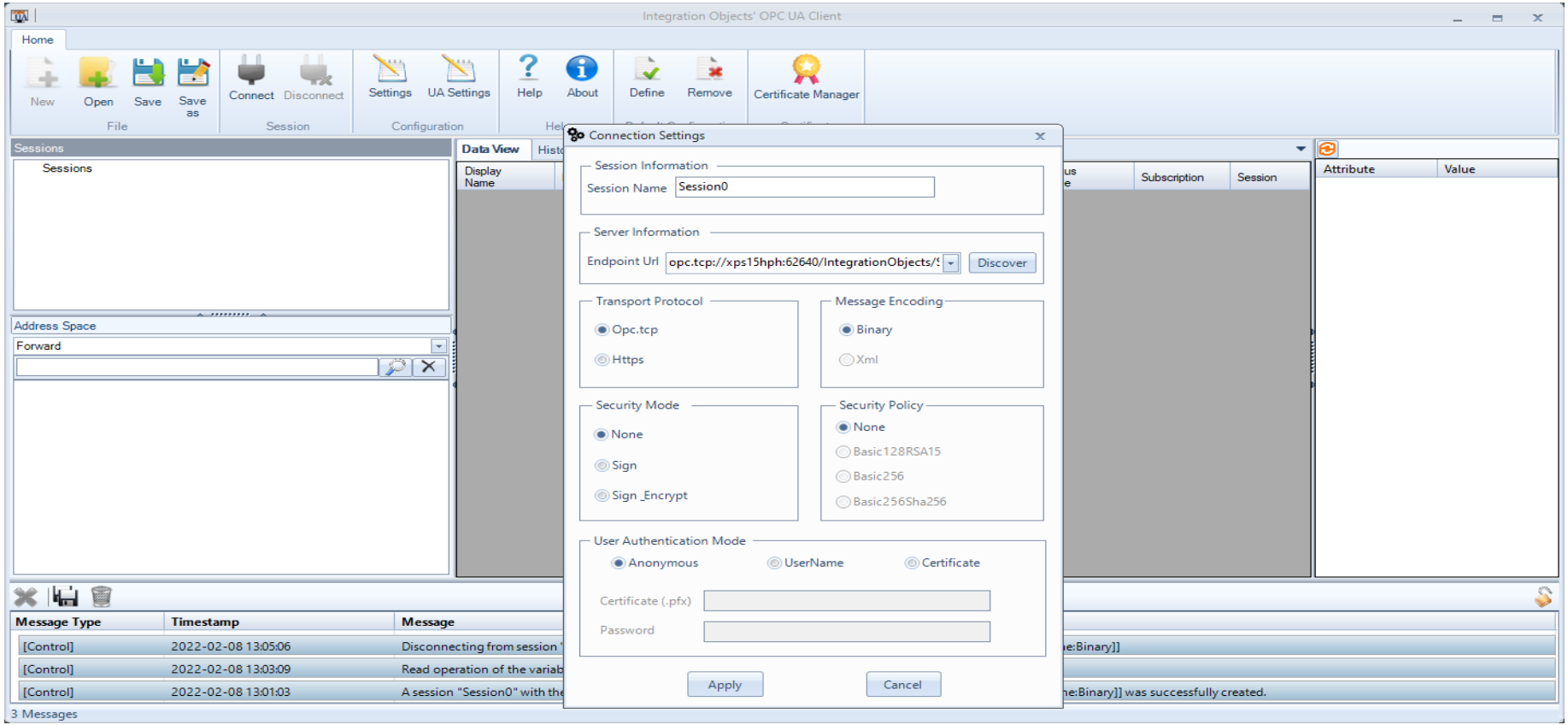
OPC UA Server Simulator



OPC UA Client

- Integration Objects also offer a free OPC UA Client
- You can use it for Test and Demonstration purposes
- Download: https://integrationobjects.com/sioth-opc/sioth-opc-unified-architecture/opc-ua-client/

OPC UA Client



UaExpert
OPC UA Client

Unified Automation

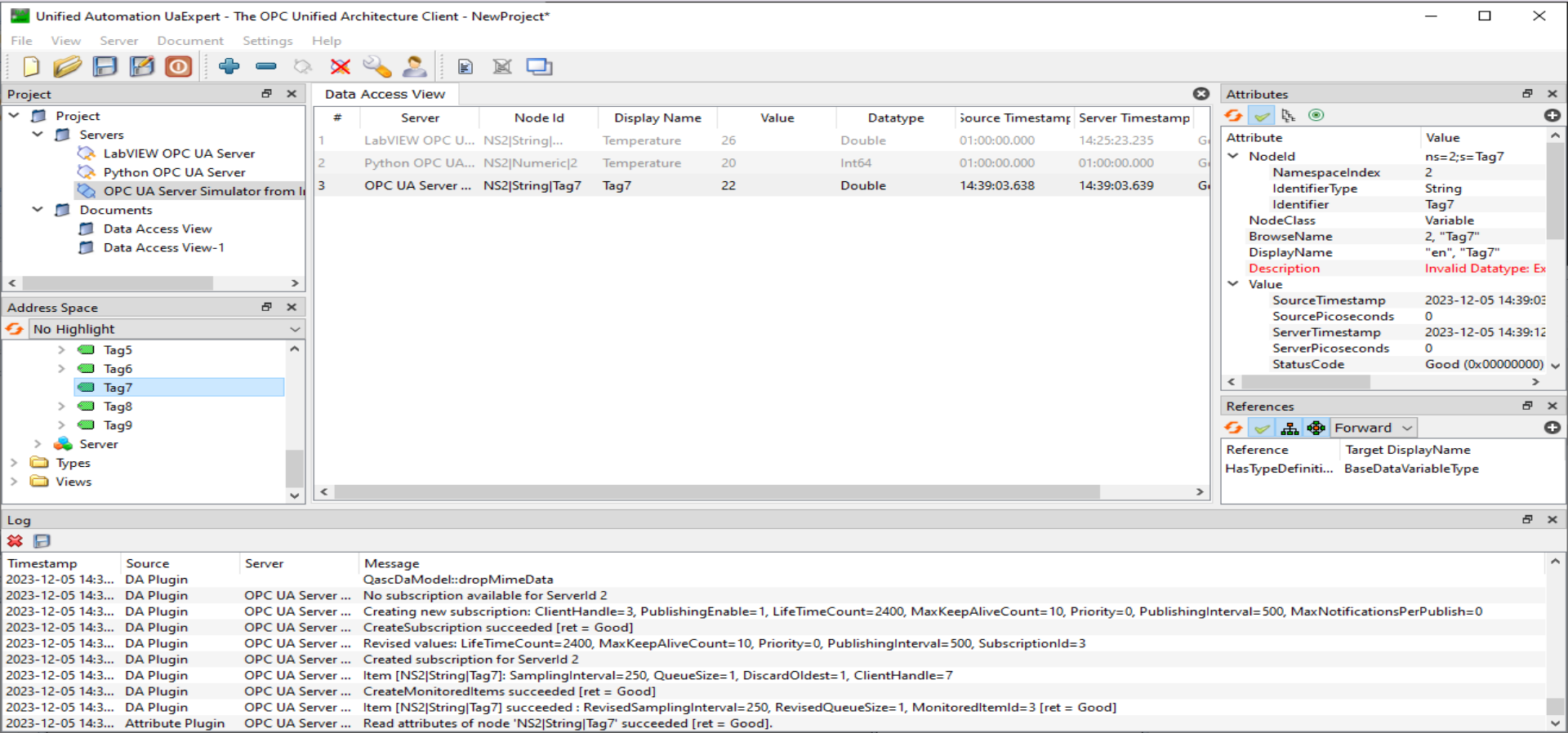
Hans-Petter Halvorsen

Table of Contents

UaExpert

- UaExpert is an OPC UA Client from Unified Automation
- Homepage: https://www.unified-automation.com
- Download UaExpert: https://www.unified-automation.com/downloads/opc-ua-clients.html

UaExpert OPC UA Client



Python OPC UA Client Examples

Hans-Petter Halvorsen

Table of Contents

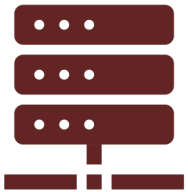
Python OPC UA Client Examples

- We start with creating OPC UA Clients with Python
- Typically, most of the times we need to create OPC UA clients, while we use existing OPC UA Servers from different vendors that already exists in our system
- Here, we will use the “OPC UA Server Simulator” from Integration Objects for Testing our OPC UA Python examples

Python OPC UA Client Examples

OPC UA Server

Normally, the OPC Server and the different OPC Clients runs on different computers distributed in a network



"OPC UA Server Simulator" from Integration Objects

Python Script

OPC UA Write Client

Python Script

OPC UA Read Client

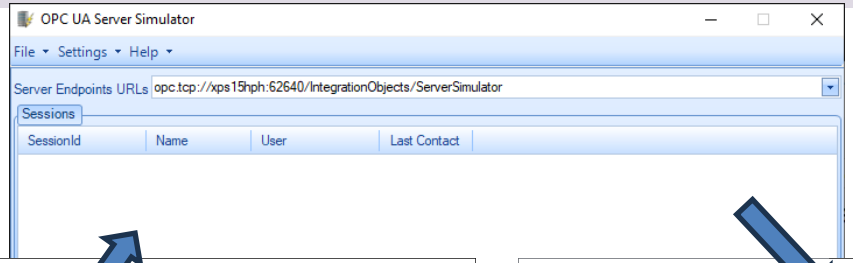
OPC UA Client Write

```
from opcua import Client

url = "xxx"
nodeId = "xxx"

client = Client(url)
client.connect()

node = client.get_node(nodeId)
value = float(20)
node.set_data_value(value)

client.disconnect()
```

OPC UA Client Read

```
from opcua import Client

url = "xxx"
nodeId = "xxx"

client = Client(url)
client.connect()

node = client.get_node(nodeId)
value = node.get_value()
print(value)

client.disconnect()
```

Python OPC UA Client Examples



Thonny - C:\Users\hansha\OneDrive\Documents\Python\Python Programming\Code Examples\OPC\opcua\Integration Objects Examples\opcua_client_write_ex.py @ 11 : 17

File Edit View Run Tools Help

opcua_client_write_ex.py | opcua_client_read_ex.py

```
1  from opcua import Client
2
3  #OPC UA Server Simulator from Integration Objects
4  url = "opc.tcp://localhost:62640/IntegrationObjects/ServerSimulator/"
5  nodeId = "ns=2;s=Tag7"
6
7  client = Client(url)
8  client.connect()
9
10 node = client.get_node(nodeId)
11 value = float(22)
12 node.set_data_value(value)
13
14 client.disconnect()
```

Write Data to OPC UA Server

Shell

```
>>> %Run opcua_client_write_ex.py
>>>
```

Local Python 3 • Thonny's Python

Thonny - C:\Users\hansha\OneDrive\Documents\Python\...rogramming\Code Examples\OPC\opcua\Integration Objects Examples\opcua_client_read_ex.py @ 10 : 31

File Edit View Run Tools Help

opcua_client_write_ex.py | opcua_client_read_ex.py

```
1  from opcua import Client
2
3  #OPC UA Server Simulator from Integration Objects
4  url = "opc.tcp://localhost:62640/IntegrationObjects/ServerSimulator/"
5  nodeId = "ns=2;s=Tag7"
6
7  client = Client(url)
8  client.connect()
9
10 node = client.get_node(nodeId)
11 value = node.get_value()
12 print(value)
13
14 client.disconnect()
```

Read Data from OPC UA Server

Shell

```
>>> %Run opcua_client_read_ex.py
 22.0
>>>
```

Local Python 3 • Thonny's Python

OPC UA Client **Write** Example

```
from opcua import Client

#OPC UA Server Simulator from Integration Objects
url = "opc.tcp://localhost:62640/IntegrationObjects/ServerSimulator/"
nodeId = "ns=2;s=Tag7"

client = Client(url)
client.connect()

node = client.get_node(nodeId)
value = float(20)
node.set_data_value(value)

client.disconnect()
```

OPC UA Client **Read** Example

```
from opcua import Client

#OPC UA Server Simulator from Integration Objects
url = "opc.tcp://localhost:62640/IntegrationObjects/ServerSimulator/"
nodeId = "ns=2;s=Tag7"

client = Client(url)
client.connect()

node = client.get_node(nodeId)
value = node.get_value()
print(value)

client.disconnect()
```

Improvements

We improve the basic Write and Read Examples

- Adding While Loop, etc.
- Error Handling
- Getting Data from a real Sensor, e.g., a Temperature Sensor
  - I will create a “Simulator” that simulates a real Temperature Sensor using the Random Generator in Python

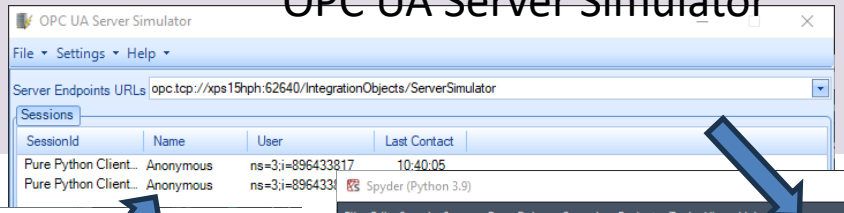
OPC UA Client **Write** Example

```
from opcua import Client
import time
import random

#OPC UA Server Simulator from Integration Objects
url = "opc.tcp://localhost:62640/IntegrationObjects/ServerSimulator/"
nodeId = "ns=2;s=Tag7"

client = Client(url)
client.connect()
print("OPC UA Client Connected")
print("Press Ctrl-C to Stop Program")

try:
    while True:
        value = random.randint(20,30)
        print(value)

        node = client.get_node(nodeId)
        value = float(value)
        node.set_data_value(value)
        time.sleep(10)

except KeyboardInterrupt:
    pass

client.disconnect()
print("OPC UA Client Disconnected and Progam Stopped")
```

OPC UA Client **Read** Example

```
from opcua import Client
import time

#OPC UA Server Simulator from Integration Objects
url = "opc.tcp://localhost:62640/IntegrationObjects/ServerSimulator/"
nodeId = "ns=2;s=Tag7"

client = Client(url)
client.connect()
print("OPC UA Client Connected")
print("Press Ctrl-C to Stop Program")

try:
    while True:
        node = client.get_node(nodeId)
        value = node.get_value()
        print(value)
        time.sleep(10)

except KeyboardInterrupt:
    pass

client.disconnect()
print("OPC UA Client Disconnected and Progam Stopped")
```

“OPC UA Server Simulator”



Thonny Python Editor

Thonny - C:\Users\hansha\OneDrive\Documents\Python\Python Programming\Code Examples\OPC\opcua\Integration Object...

File Edit View Run Tools Help

opcua_client_write_while_loop.py

```
3  import random
4
5  #OPC UA Server Simulator from Integration Objects
6  url = "opc.tcp://localhost:62640/IntegrationObjects/S
7  nodeId = "ns=2;s=Tag7"
8
9  client = Client(url)
10 client.connect()
11 print("OPC UA Client Connected")
12 print("Press Ctrl-C to Stop Program")
13
14 try:
15     while True:
16         value = random.randint(20,30)
17         print(value)
18
19         node = client.get_node(nodeId)
20         value = float(value)
21         node.set_data_value(value)
22         time.sleep(10)
```

Shell

```
20
28
28
```

Local Python 3 • Thonny's Python

OPC UA Client **Write** Example

Spyder Python Editor

Spyder (Python 3.9)

File Edit Search Source Run Debug Consoles Projects Tools View Help

...hon Programming\Code Examples\OPC\opcua\Integration Objects Examples\opcua_client_read_while_loop.py

opcua_client_read_while_loop.py

```
1  from opcua import client
2  import time
3
4  #OPC UA Server Simulator from Integration Objects
5  url = "opc.tcp://localhost:62640/IntegrationObjects/ServerSimulator/"
6  nodeId = "ns=2;s=Tag7"
7
8  client = Client(url)
9  client.connect()
10 print("OPC UA Client Connected")
11 print("Press Ctrl-C to Stop Program")
12
13 try:
14     while True:
15         node = client.get_node(nodeId)
16         value = node.get_value()
17         print(value)
18         time.sleep(10)
19
20 except KeyboardInterrupt:
21     pass
22
23 client.disconnect()
24 print("OPC UA Client Disconnected and Progam Stopped")
```

Source Console Object

Usage

Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

Help Variable Explorer Plots Files

Console 1/A

```
Documents/Python/Python Programming/Code Examples/OPC/opcua/Integration Objects Examples/opcua_client_read_while_loop.py',
wdir='C:/Users/hansha/OneDrive/Documents/Python/Python Programming/Code Examples/OPC/opcua/Integration Objects Examples')
OPC UA Client Connected
Press Ctrl-C to Stop Program
30.0
23.0
27.0
26.0
20.0
28.0
28.0
28.0
29.0
```

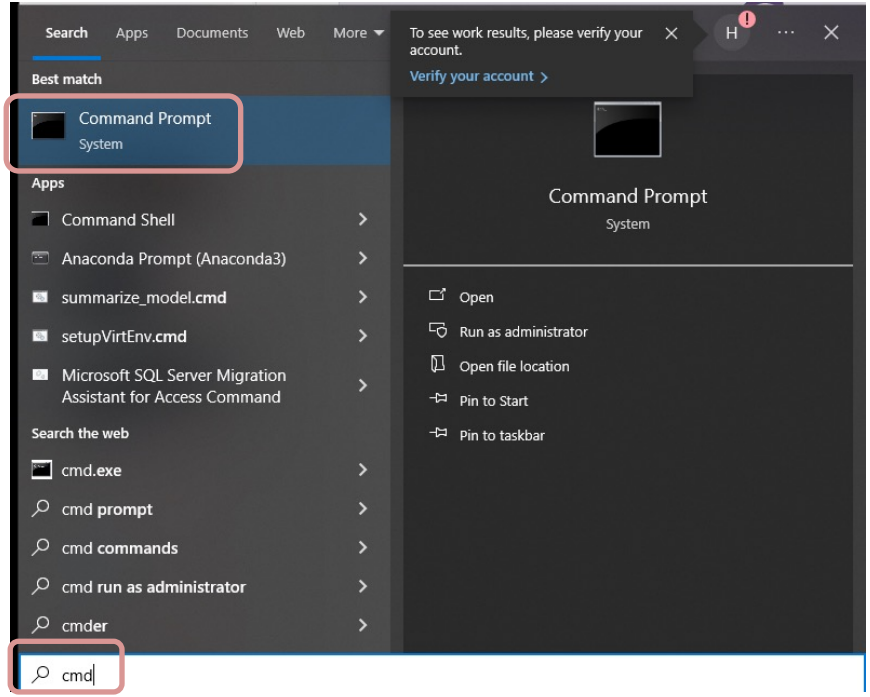
IPython console History

Kite: ready LSP Python: ready conda: base (Python 3.9.7) Line 24, Col 55 ASCII CRLF RW Mem 45%

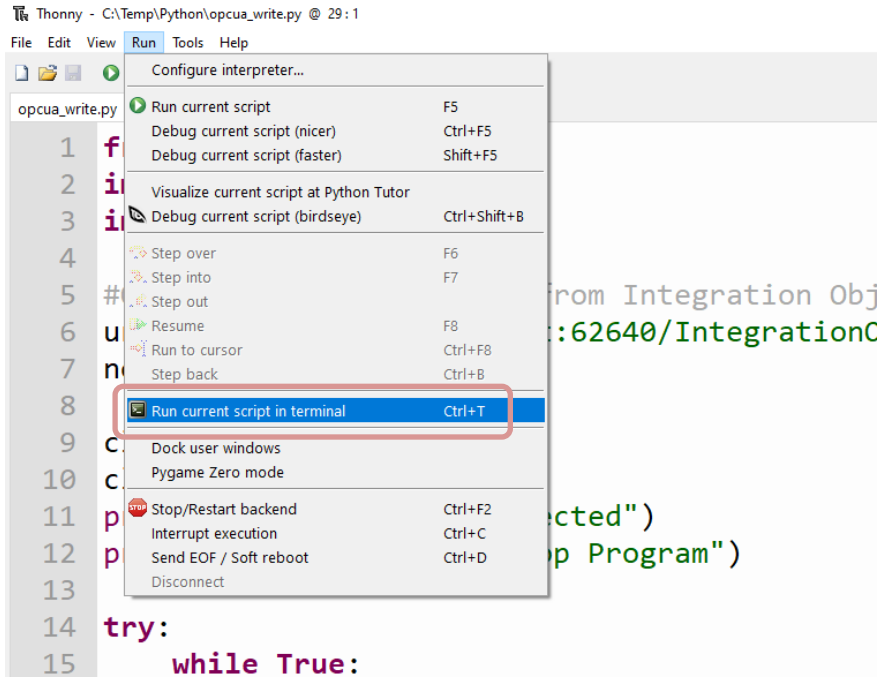
OPC UA Client **Read** Example

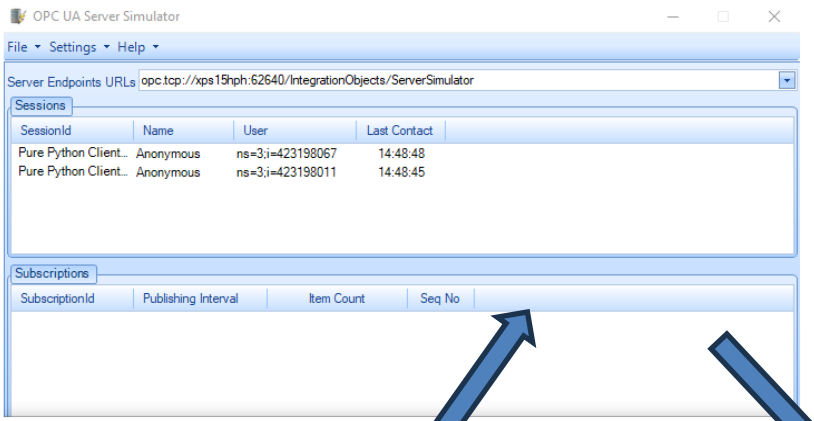
Running Multiple instances of the Command Prompt

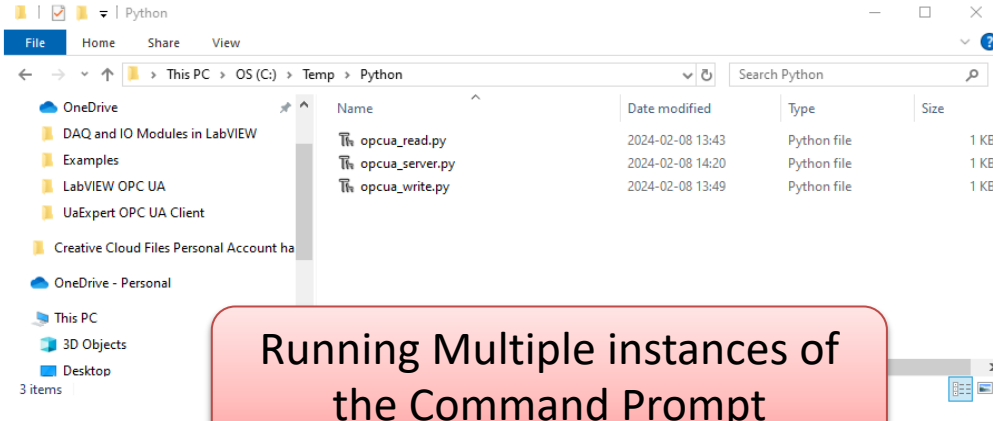
Open the Command Prompt from the Start menu



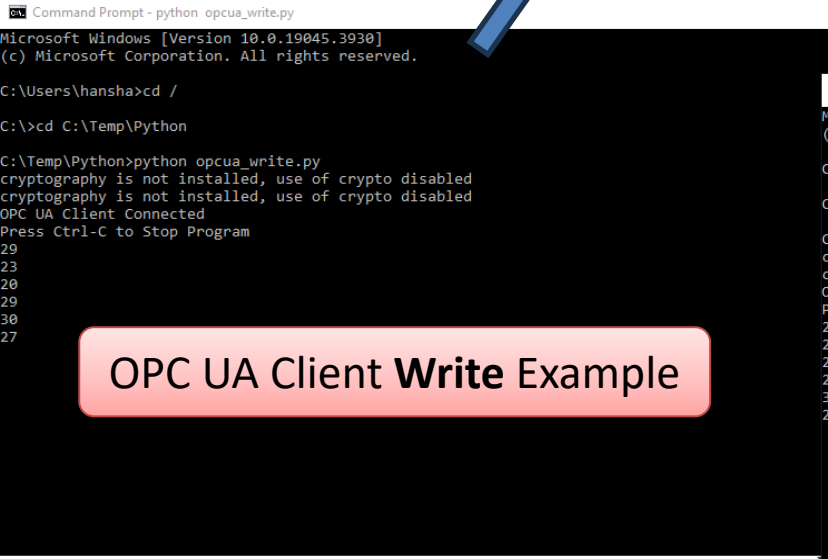
You can also do it from the Thonny Editor:



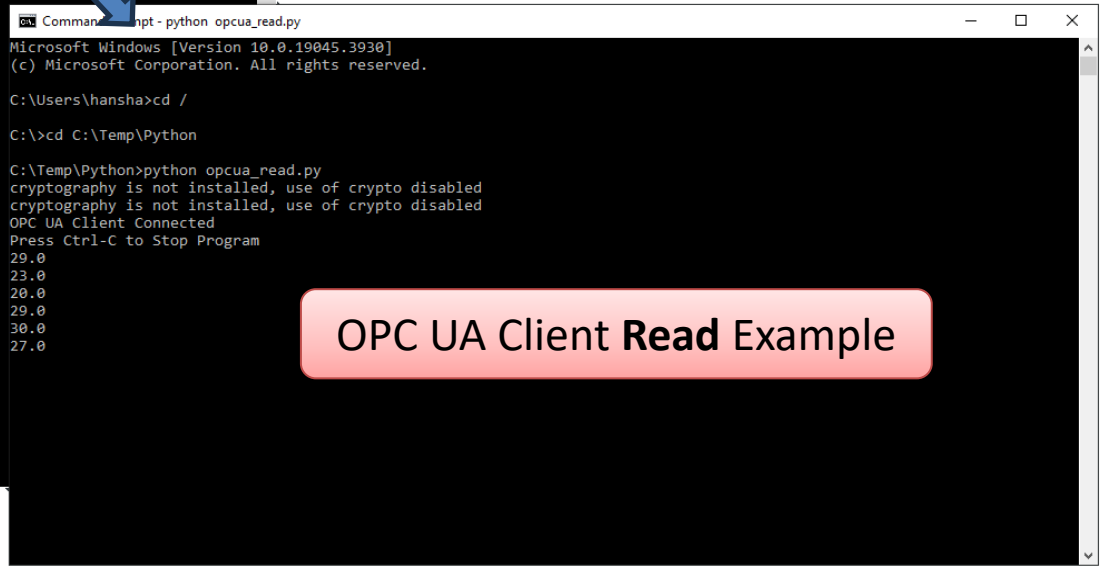




Running Multiple instances of the Command Prompt

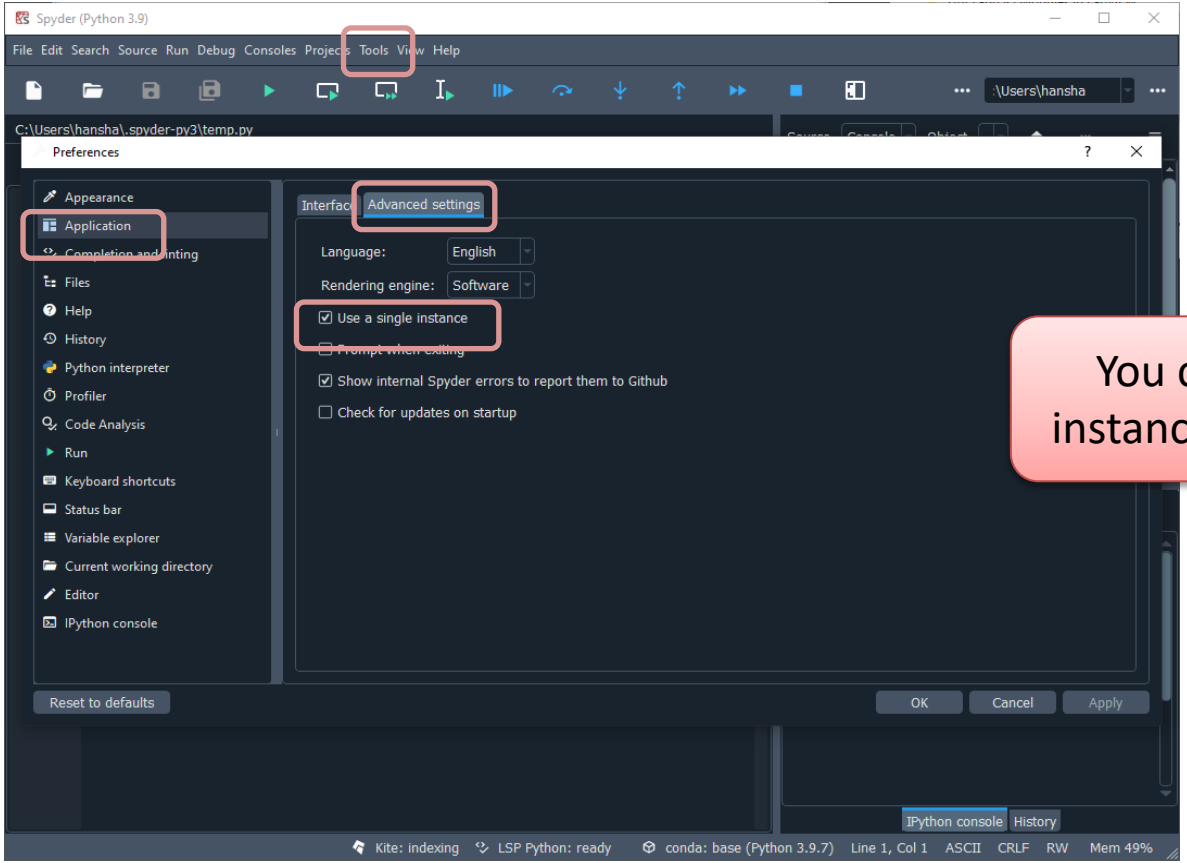


OPC UA Client **Write** Example

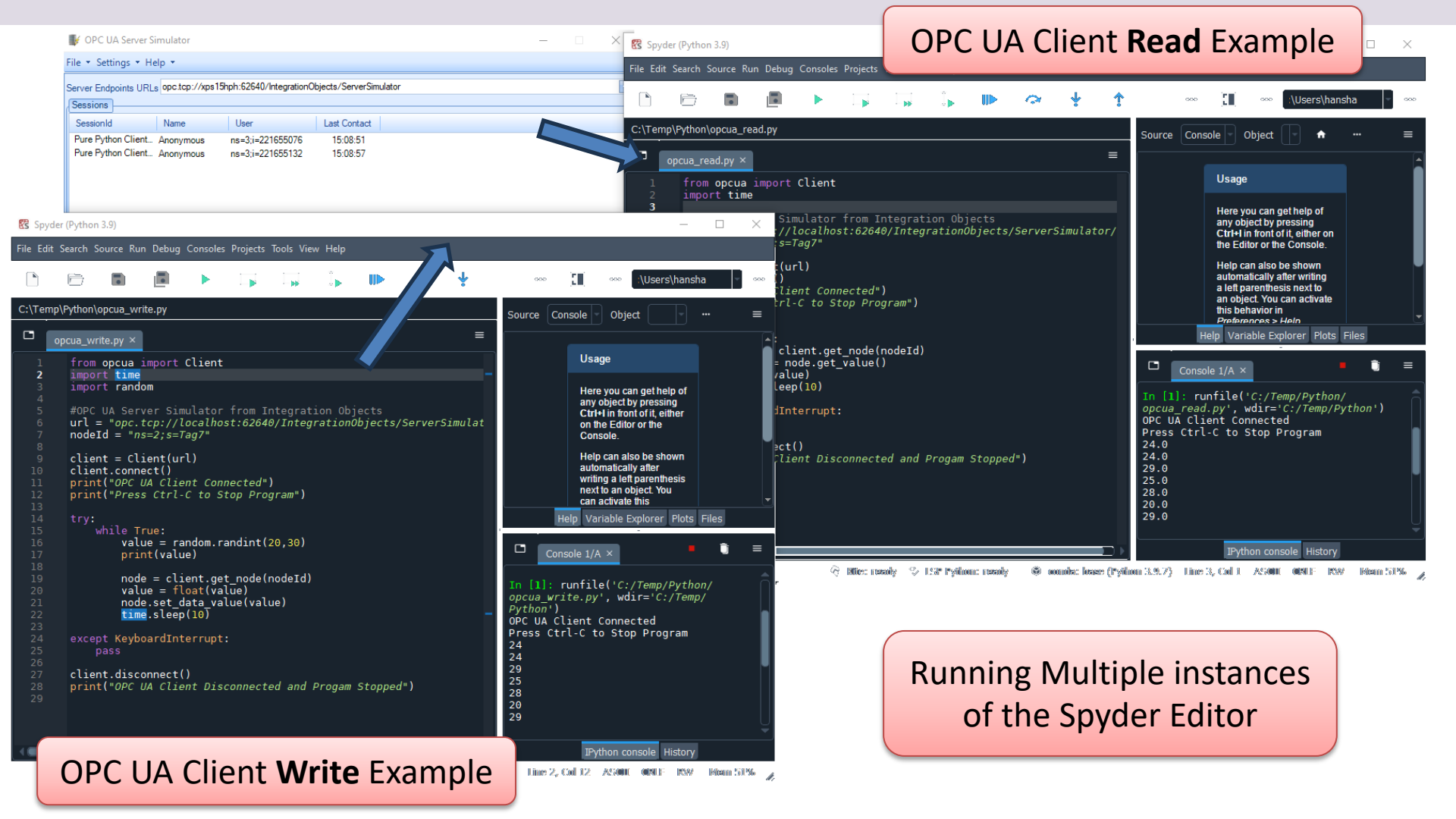


OPC UA Client **Read** Example

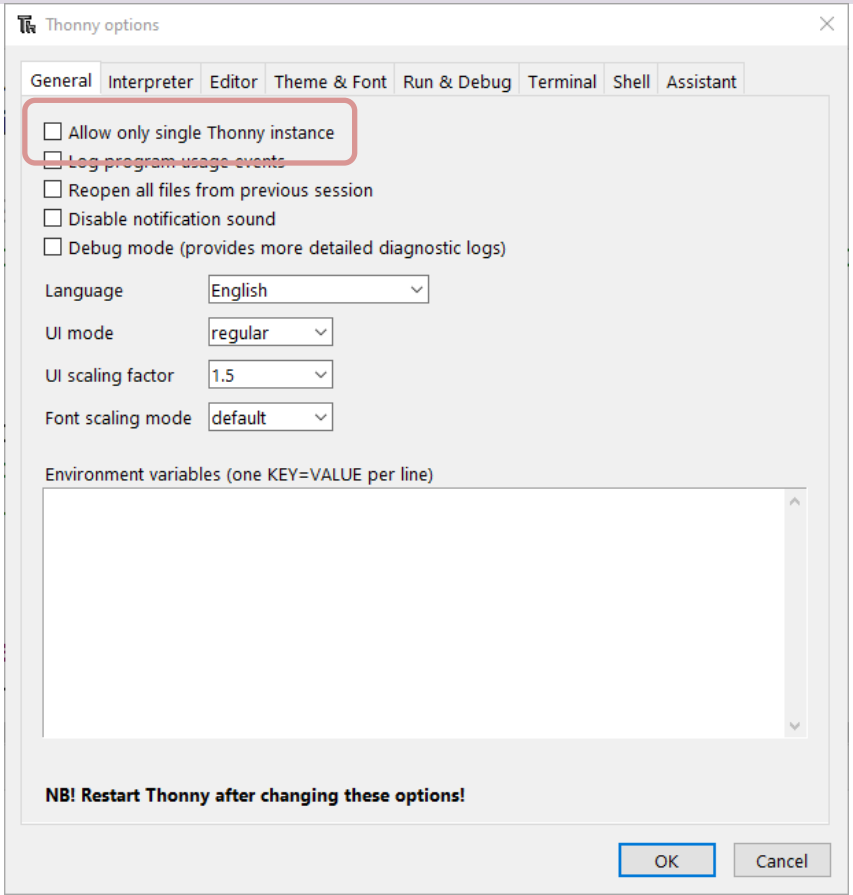
Multiple Spyder Editors



You can also run Multiple instances of the Spyder Editor


OPC UA Client Read Example
OPC UA Server Simulator
File Settings Help
Server Endpoints URLs opc.tcp://xps15hph:62640/IntegrationObjects/ServerSimulator
Sessions
SessionId Name User Last Contact
Pure Python Client... Anonymous ns=3;i=221655076 15:08:51
Pure Python Client... Anonymous ns=3;i=221655132 15:08:57
Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects
C:\Temp\Python\opcua_read.py
opcua_read.py
from opcua import Client
import time
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help
Help Variable Explorer Plots Files
Console 1/A
In [1]: runfile('C:/Temp/Python/opcua_read.py', wdir='C:/Temp/Python')
OPC UA Client Connected
Press Ctrl-C to Stop Program
24.0
24.0
29.0
25.0
28.0
20.0
29.0
IPython console History
Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Temp\Python\opcua_write.py
opcua_write.py
from opcua import Client
import time
import random
#OPC UA Server Simulator from Integration Objects
url = "opc.tcp://localhost:62640/IntegrationObjects/ServerSimulat
nodeId = "ns=2;s=Tag7"
client = Client(url)
client.connect()
print("OPC UA Client Connected")
print("Press Ctrl-C to Stop Program")
try:
while True:
value = random.randint(20,30)
print(value)
node = client.get_node(nodeId)
value = float(value)
node.set_data_value(value)
time.sleep(10)
except KeyboardInterrupt:
pass
client.disconnect()
print("OPC UA Client Disconnected and Progam Stopped")
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this
Help Variable Explorer Plots Files
Console 1/A
In [1]: runfile('C:/Temp/Python/opcua_write.py', wdir='C:/Temp/Python')
OPC UA Client Connected
Press Ctrl-C to Stop Program
24
24
29
25
28
20
29
IPython console History
OPC UA Client Write Example
Running Multiple instances of the Spyder Editor

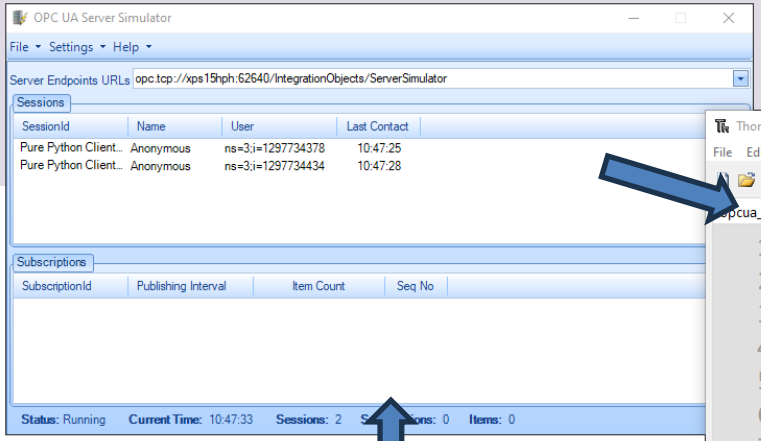
Multiple Thonny Editors

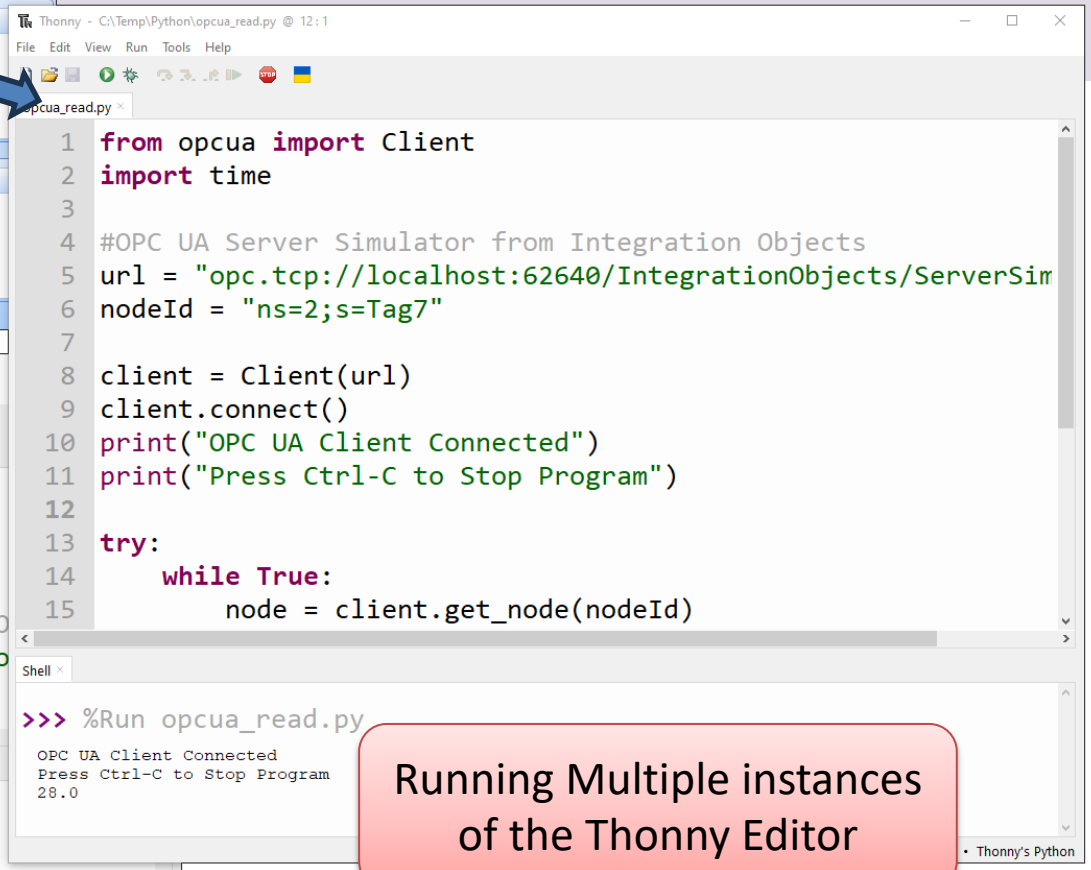


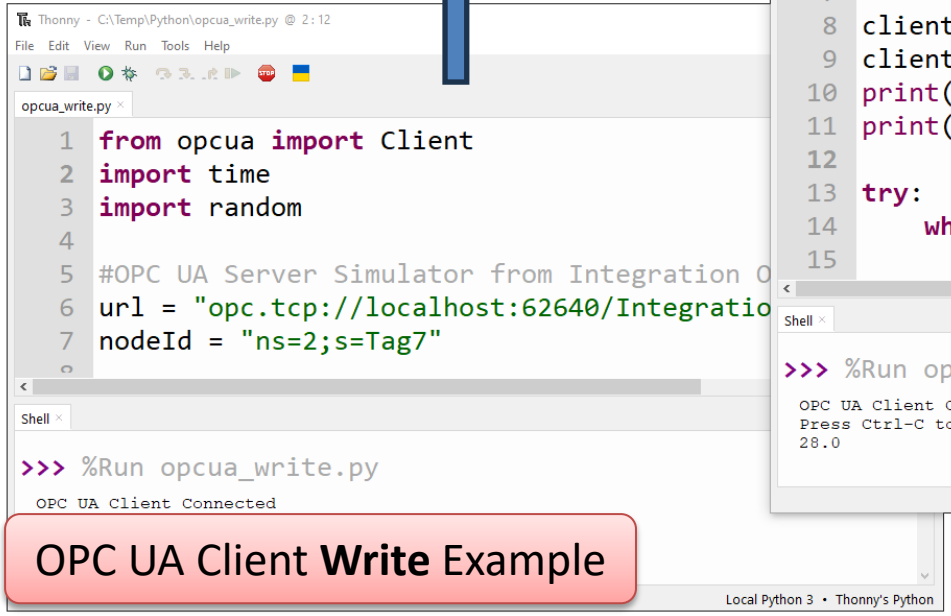
Tools -> Options …

You can also run Multiple instances of the Thonny Editor

OPC UA Client **Read** Example







OPC UA Client **Write** Example

Running Multiple instances of the Thonny Editor

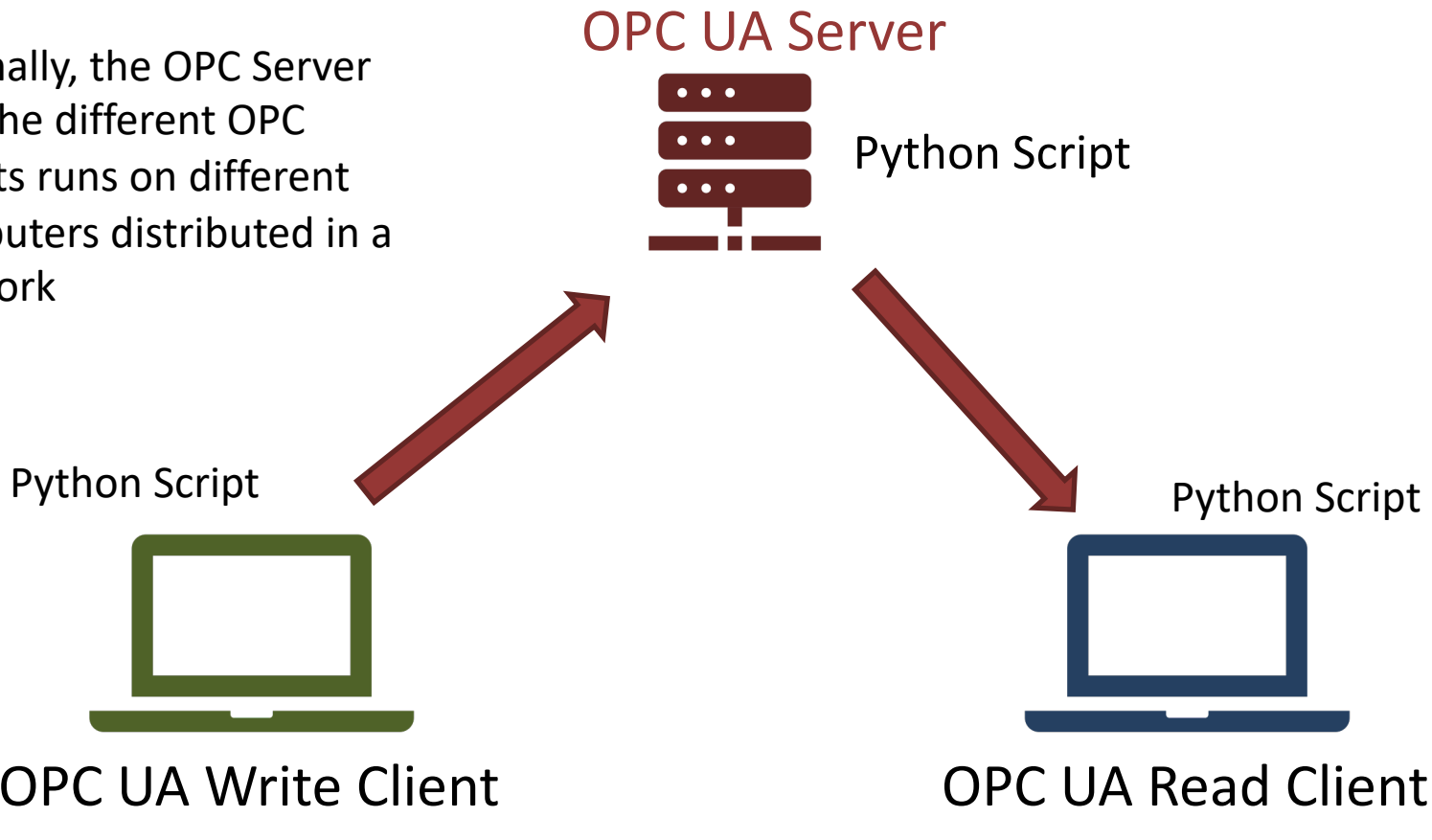
Python OPC UA Server Examples

Table of Contents

Hans-Petter Halvorsen

Python OPC UA Server

Normally, the OPC Server and the different OPC Clients runs on different computers distributed in a network



OPC UA Server

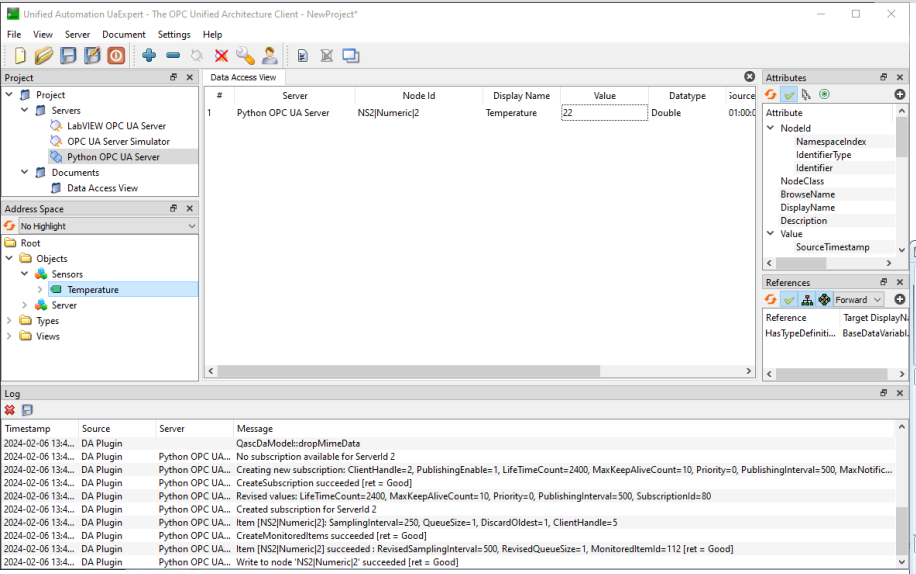
```
from opcua import Server

server = Server()
server_url = "opc.tcp://127.0.0.1:1234"
server.set_endpoint(server_url)

name ="opcuapython"
namespace = server.register_namespace(name)

node = server.get_objects_node()
param = node.add_object(namespace, "Sensors")

var = param.add_variable(namespace, "Temperature",0)
var.set_writable()

server.start()
```

This is the main Python code for creating an OPC UA Server

OPC UA Server

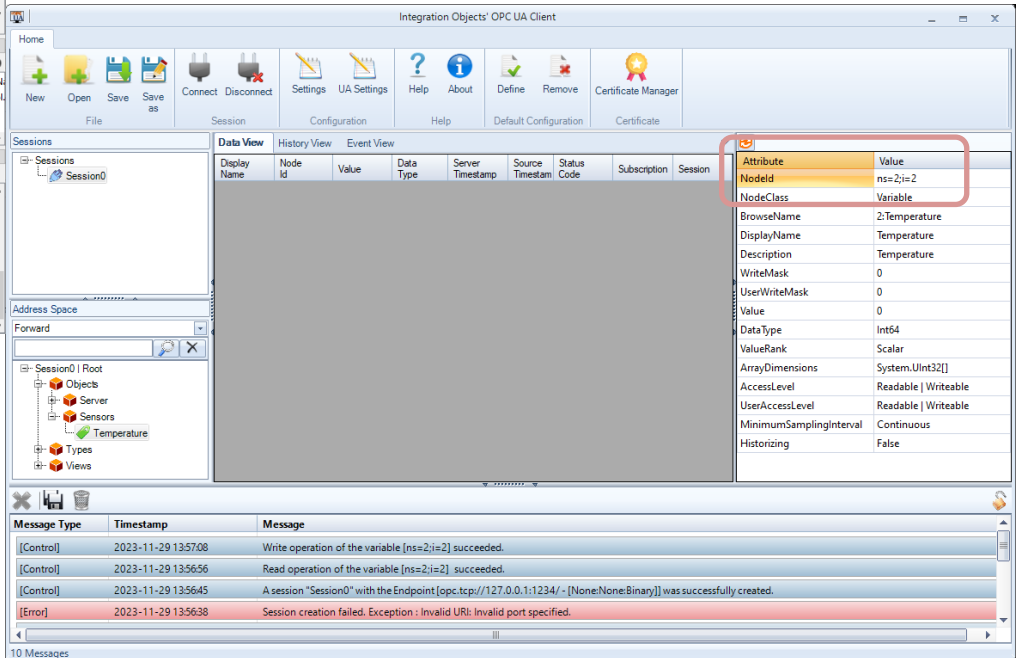
```
from opcua import Server
import time
from datetime import datetime

server = Server()
server_url = "opc.tcp://127.0.0.1:1234"
server.set_endpoint(server_url)

name ="opcuapython"
namespace = server.register_namespace(name)
node = server.get_objects_node()
param = node.add_object(namespace, "Sensors")
var = param.add_variable(namespace, "Temperature",0)
var.set_writable()

server.start()
print("OPC UA Server Started")
print("Press Ctrl-C to Stop Program")

try:
    while True:
        now = datetime.now()
        current_time = now.strftime("%H:%M:%S")
        print("OPC UA Server Running", current_time)
        value = var.get_value()
        print("Current Value:", value)
        time.sleep(1)

except KeyboardInterrupt:
    pass

server.stop()
print("OPC UA Server Stopped")
```

Test Connection to Server



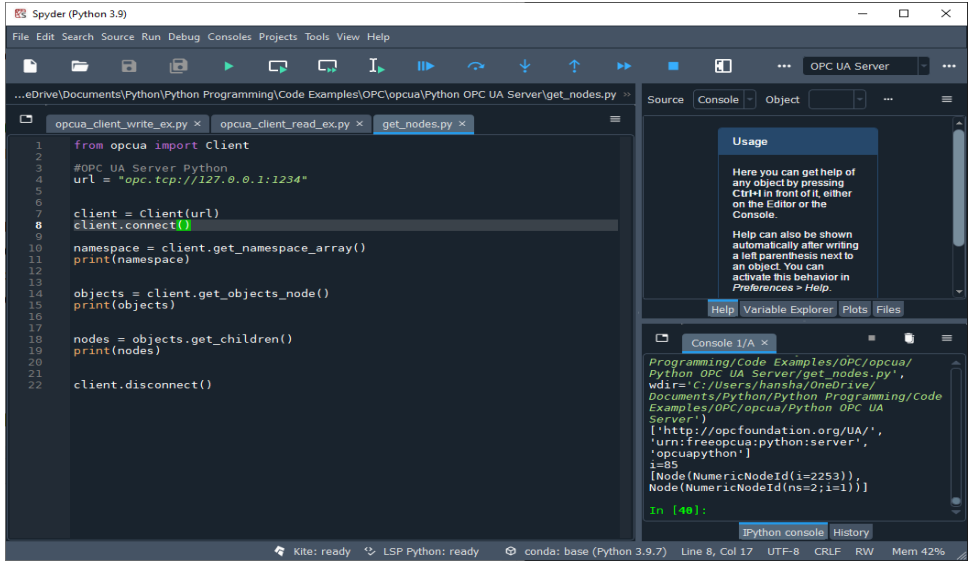
UaExpert OPC UA Client

OPC UA Client from Integration Objects



Get NodeId

```
from opcua import Client

#OPC UA Server Python
url = "opc.tcp://127.0.0.1:1234"

client = Client(url)
client.connect()

namespace = client.get_namespace_array()
print(namespace)

objects = client.get_objects_node()
print(objects)

nodes = objects.get_children()
print(nodes)

client.disconnect()
```



This gives:

NodeId ="ns=2;i=2"

OPC UA Client **Write** Example

```
from opcua import Client

#OPC UA Server Python
url = "opc.tcp://127.0.0.1:1234"
nodeId = "ns=2;i=2"

client = Client(url)
client.connect()

node = client.get_node(nodeId)
value = float(20)
node.set_data_value(value)

client.disconnect()
```

Write - Alternative Solution

```
from opcua import Client

#OPC UA Server Python
url = "opc.tcp://127.0.0.1:1234"

client = Client(url)
client.connect()

root = client.get_root_node()
nodeId = root.get_child(["0:Objects", "2:Sensors", "2:Temperature"])
print("NodeId:", nodeId)

node = client.get_node(nodeId)
value = float(20)
node.set_data_value(value)

client.disconnect()
```

OPC UA Client **Read** Example

```
from opcua import Client

#OPC UA Server Python
url = "opc.tcp://127.0.0.1:1234"
nodeId = "ns=2;i=2"

client = Client(url)
client.connect()

node = client.get_node(nodeId)
value = node.get_value()
print(value)

client.disconnect()
```

Read - Alternative Solution

```
from opcua import Client

#OPC UA Server Python
url = "opc.tcp://127.0.0.1:1234"

client = Client(url)
client.connect()

root = client.get_root_node()
nodeId = root.get_child(["0:Objects", "2:Sensors", "2:Temperature"])
print("NodeId:", nodeId)

node = client.get_node(nodeId)
value = node.get_value()
print("Value[°C]:", value)

client.disconnect()
```

Testing OPC Server and Clients

OPC UA Server

Thonny - C:\Users\hansha\OneDrive\Documents\Python\Python Programming\Code Examples\OPC\opcua\Python OPC UA Server\opcua_server_ex.py @ 22 : 27

File Edit View Run Tools Help

opcua_server_ex.py

```python
from opcua import Server
import time
from datetime import datetime

server = Server()
server_url = "opc.tcp://127.0.0.1:1234"
server.set_endpoint(server_url)

name ="opcuapython"
namespace = server.register_namespace(name)
node = server.get_objects_node()
param = node.add_object(namespace, "Sensors")
var = param.add_variable(namespace, "Temperature",0)
var.set_writable()

server.start()
print("OPC UA Server Started")
print("Press Ctrl-C to Stop Program")
```

Shell

```
Current Value: 20.0
OPC UA Server Running 13:39:06
Current Value: 20.0
OPC UA Server Running 13:39:07
Current Value: 20.0
OPC UA Server Running 13:39:08
Current Value: 20.0
OPC UA Server Running 13:39:09
Current Value: 20.0
```

Local Python

Spyder (Python 3.9)

File Edit Search Source Run Debug Consoles Projects Tools View Help

OPC UA Server

...ents\Python\Python Programming\Code Examples\OPC\opcua\Python OPC UA Server\opcua_client_write_ex.py

opcua_client_write_ex.py × opcua_client_read_ex.py ×

```python
from opcua import Client

#OPC UA Server Python
url = "opc.tcp://127.0.0.1:1234"
nodeId = "ns=2;i=2"

client = Client(url)
client.connect()

node = client.get_node(nodeId)
value = float(20)
node.set_data_value(value)

client.disconnect()
```

OPC UA Clients

Source Console Object

Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

Help Variable Explorer Plots Files

Console 1/A ×

```
Programming/Code Examples/OPC/opcua/
Python OPC UA Server/
opcua_client_read_ex.py', wdir='C:/
Users/hansha/OneDrive/Documents/Python/
Python Programming/Code Examples/OPC/
opcua/Python OPC UA Server')
20.0

In [28]: runfile('C:/Users/hansha/
OneDrive/Documents/Python/Python
Programming/Code Examples/OPC/opcua/
Python OPC UA Server/
opcua_client_write_ex.py', wdir='C:/
Users/hansha/OneDrive/Documents/Python/
Python Programming/Code Examples/OPC/
opcua/Python OPC UA Server')

In [29]:
```

IPython console History

Kite: ready LSP Python: ready conda: base (Python 3.9.7) Line 4, Col 32 ASCII CRLF RW Mem 42%

OPC UA Python Client GUI

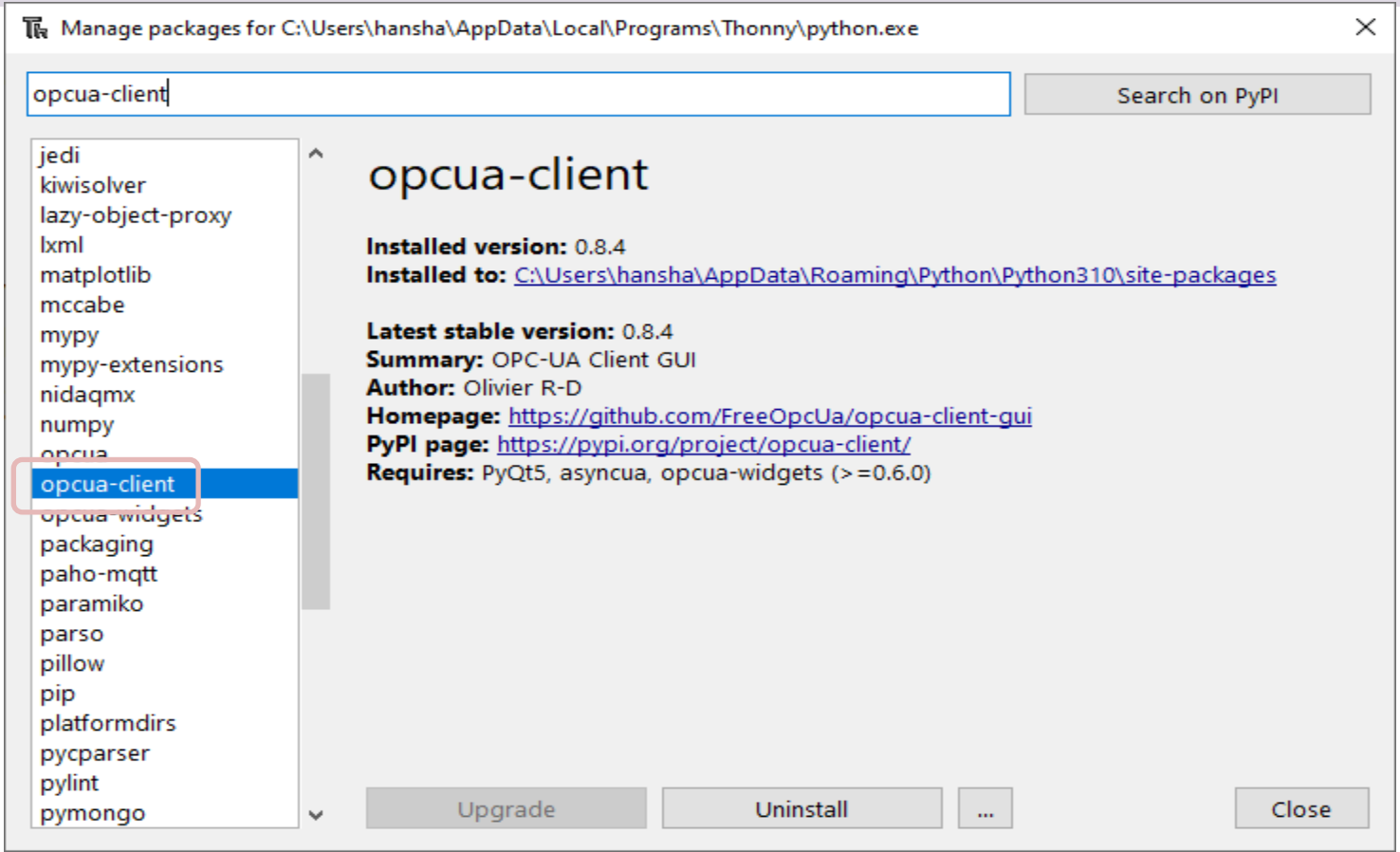
Hans-Petter Halvorsen

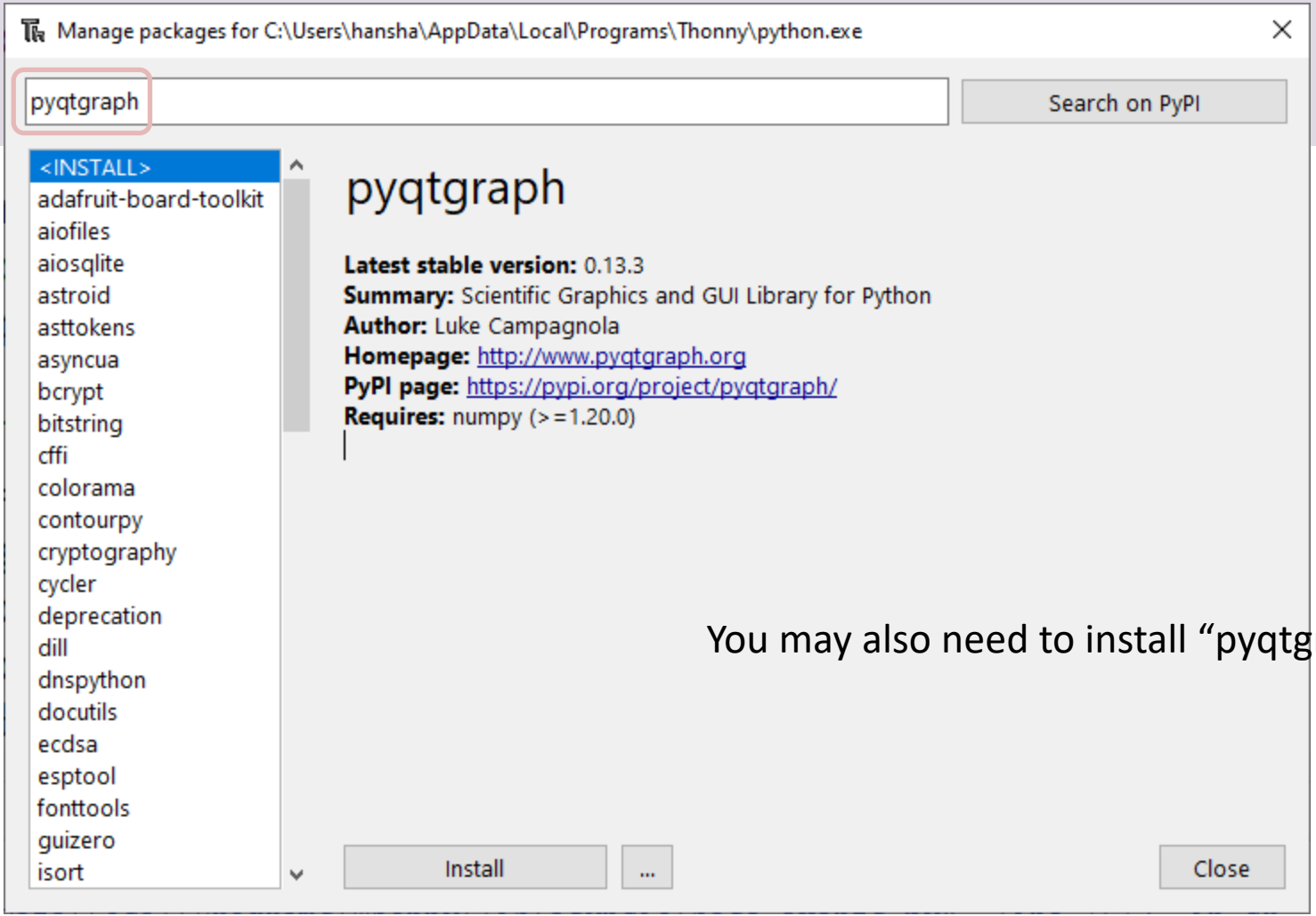
Table of Contents

Installation

- OPC UA Python Client GUI
- https://github.com/FreeOpcUa/opcua-client-gui

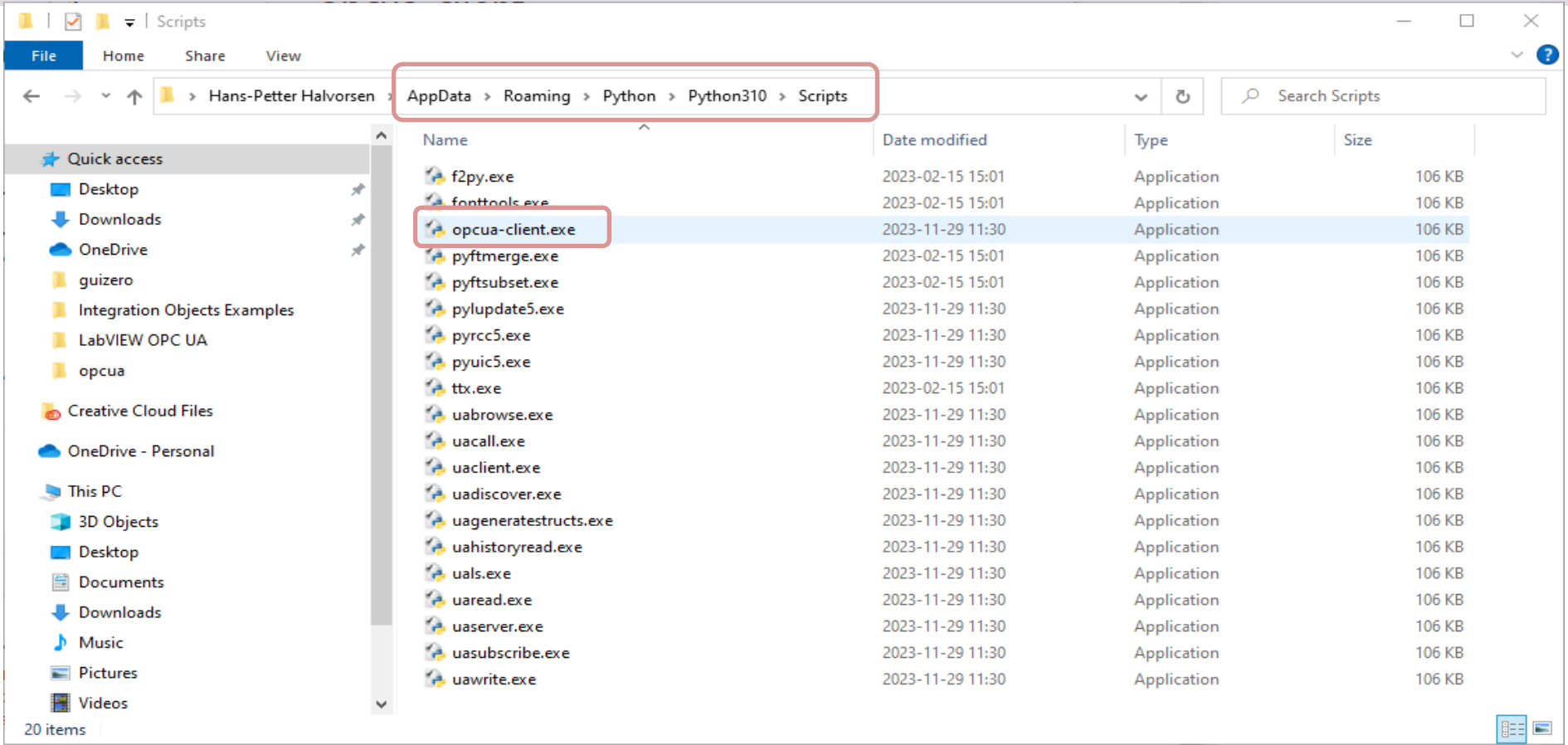
Installation using Thonny



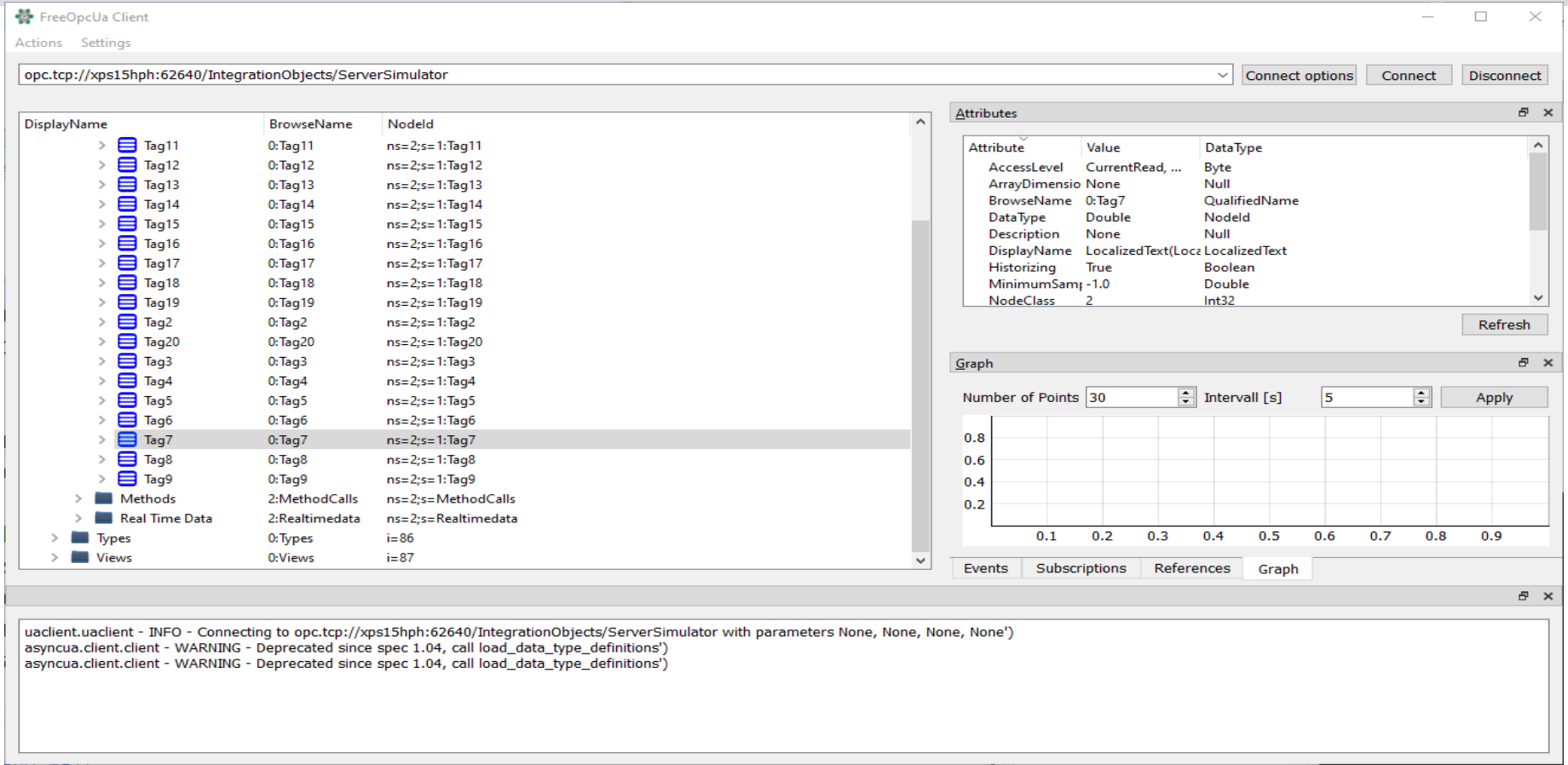


You may also need to install “pyqtgraph”

Location of “opcua-client.exe”



OPC UA Python Client GUI



Hans-Petter Halvorsen

University of South-Eastern Norway

www.usn.no

E-mail: hans.p.halvorsen@usn.no

Web: https://www.halvorsen.blog