Contents

1. Author and Contact Information

- Mohit Agarwal – mohit.agarwal@atlascopco.com
    - **Editor** of VDMA OPC UA Industrial Joining Technologies Working Group.
- Contact for any questions/updates/support on using the demo and extending it.

2. Prerequisites

- **Windows Binary**
    - Windows 10 or later (Built using Windows SDK Version: **10.0.26100**).
    - Download from the following link: Windows_SDK_Dowload
    - Download **Visual Studio 2022 Redistributable**: VC-Redist_Download
- **Docker Image: Ensure** that Docker is installed and running.
- **OPC UA Test Client:** Download and install any OPC UA Client. **Example**: UaExpert_Download

3. Installation of Application Files

- Download the following files in the **Installation Directory: OPC_UA_IJT_Server_Simulator**.
    - **opcua_ijt_demo_application.exe**
    - Contains several NodeSet files in XML format as below:
        - Opc.Ua.XXX.NodeSet2.xml
    - **Optional Files**
        - server_configuration.json
        - simulated_data.json
        - Dockerfile

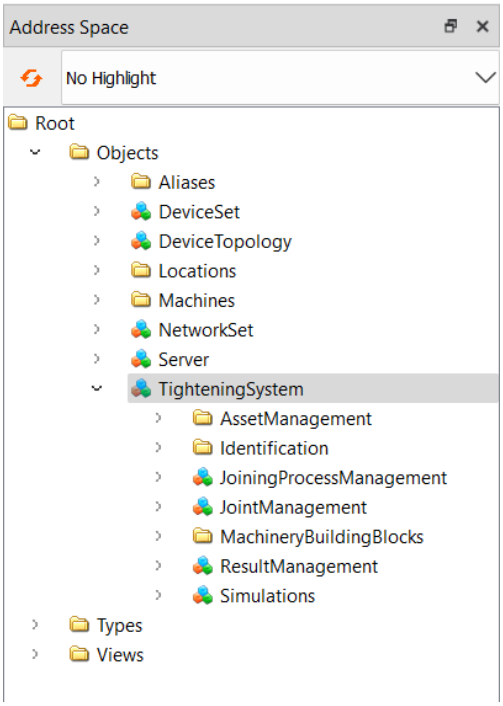
4. Running the application

- **Common Steps**
    - **Go** to the **"OPC_UA_IJT_Server_Simulator"** directory.
    - The **EndpointUrl** of the OPC UA Server would be:
        - **opc.tcp://localhost:40451** or **opc.tcp://YourComputerName:40451**.
- **Windows Binary**
    - Ensure that the user has **Read/Write access** to the **Installation Directory**.
    - **Launch** the binary file (**opcua_ijt_demo_application.exe**).
        - Run as Administrator or at least with **Read/Write** access.
- **Docker Image**
    - **Run** the following commands which will run the simulator in a docker container:
        - docker build -t opcua_ijt_demo_application .
        - docker run --rm -p 40451:40451 opcua_ijt_demo_application

5. Access data from OPC UA Client

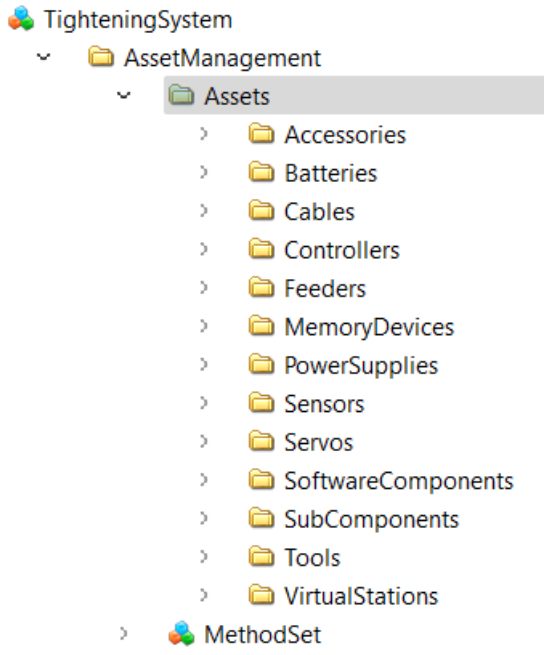
- Launch the OPC UA Client and connect to the given **EndpointUrl**.
- It will show the primary entry point: **TighteningSystem.**
- **All** the **Nodes** shown below are as per the **Companion Specification.**
- The **Simulations** node is the Application Node.

6. Tightening System Entry Point - AddressSpace View

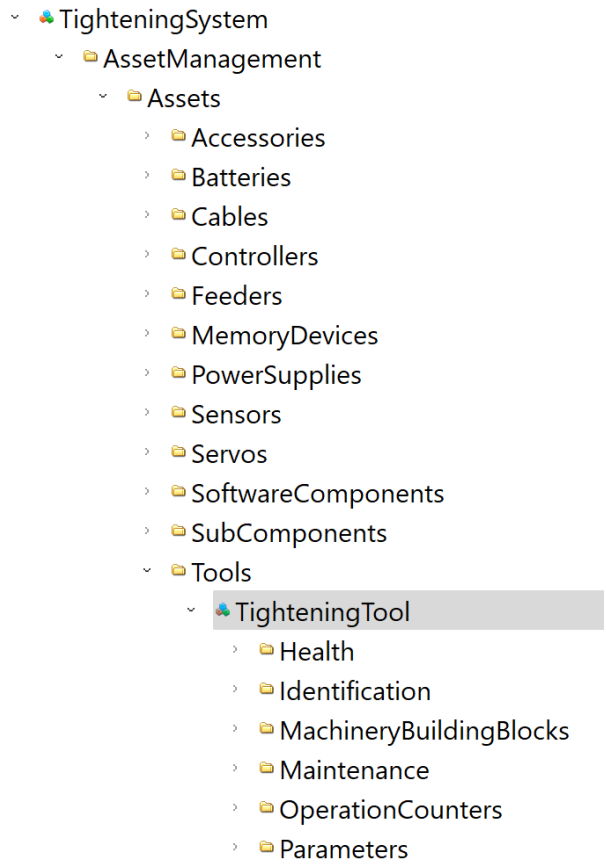


7. Asset Data

Browse the respective Asset Nodes from the address space and subscribe/read the respective data.



Example Asset Address Space View

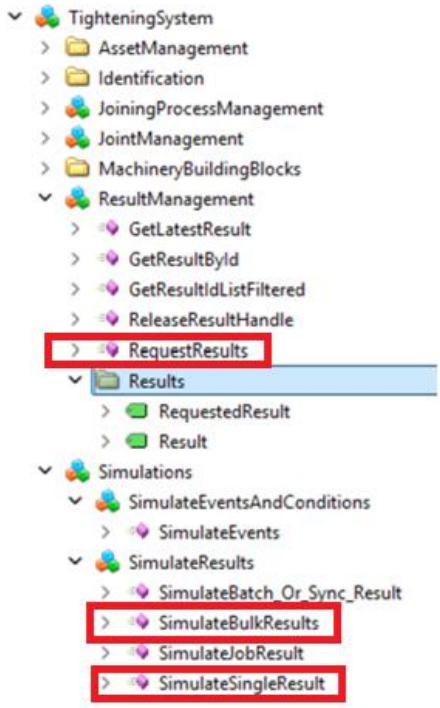


8. Result Data

- **Result Access Options:**
  - Subscribe to the **Result variable** shown below.
  - Subscribe to **events** by subscribing to the **Server** node in the Event View.
- **Simulation Options**
  - Use the following **three methods** to simulate different types of **Results**. A new **Result** is generated upon the execution of the following **methods**.
    - **SimulateBatch_or_Sync_Result**
    - **SimulateJobResult**
    - **SimulateSingleResult**
    - **SimulateBulkResults**
  - The simulated data is similar to the examples **defined** in the **Annexure** sections of the Companion Specification.
- To generate a **new live** Result, execute the **SimulateSingleResult method** shown in the below image.
- To generate a **historical** Result, execute the **RequestResults method** shown in the below image.

9. Subscribe to Result and Requested Result Variable

- **Subscribe** to **Result** Variable or **RequestedResult** Variable shown below.
- **Result** Variable is intended for getting **live** results.
- **RequestedResult** Variable is intended for getting **historical** results.



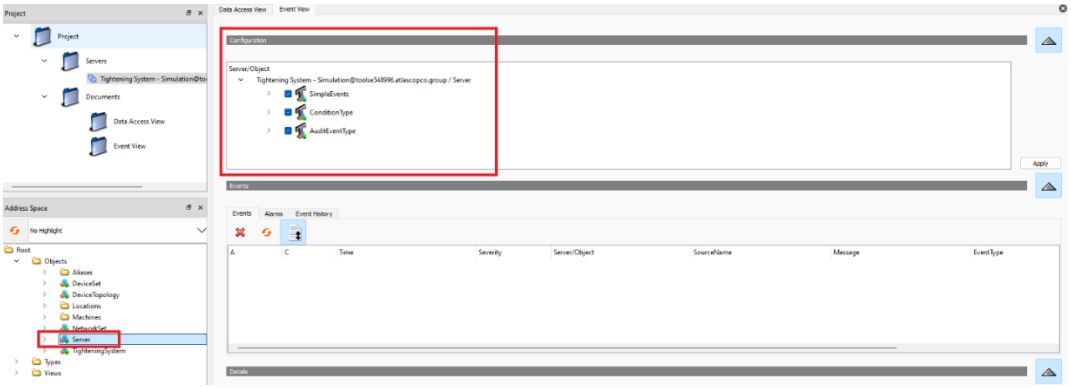
- The outcome can be visualized in the **Data Access View** or **Event View** if the respective **Result variable or Event is subscribed.**

Example Result Data Access View

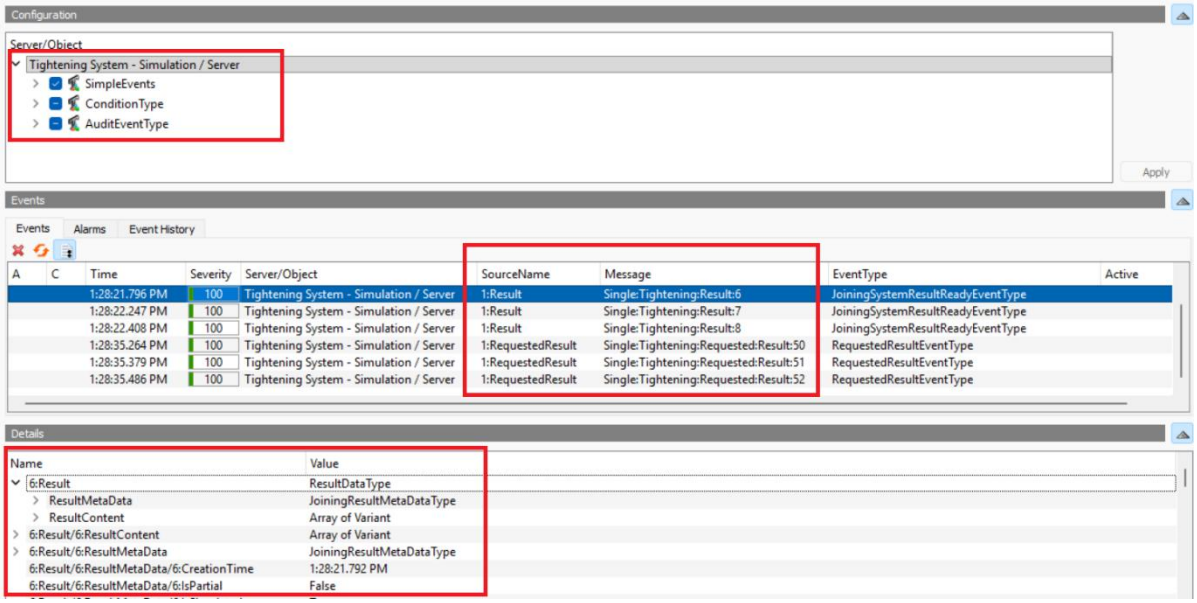
#	Display Name	Value	Datatype	Source Timestamp
1	Result	Double click to display value	ExtensionObject	1:26:59.057 PM
2	ResultContent	Double click to display value	Variant	1:26:59.058 PM
3	ResultMetaData	Double click to display value	ExtensionObject	1:26:59.060 PM
4	AssemblyType	1	Byte	1:24:58.812 PM
5	AssociatedEntities	Double click to display value	ExtensionObject	1:24:58.812 PM
6	Classification	1	Byte	1:24:58.812 PM
7	CreationTime	2025-10-05T11:26:58.969Z	DateTime	1:26:59.061 PM
8	Description	"en", "Single:Tightening:Result:4"	LocalizedText	1:26:59.061 PM
9	ExtendedMetaData	Double click to display value	ExtensionObject	1:24:58.812 PM
10	InterventionType	0	Byte	1:24:58.812 PM
11	IsGeneratedOffline	false	Boolean	1:24:58.812 PM
12	IsPartial	false	Boolean	1:24:58.812 PM
13	IsSimulated	true	Boolean	1:24:58.812 PM
14	JoiningTechnology	"en", "Tightening"	LocalizedText	1:24:58.812 PM
15	Name	Single:Tightening:Result:4	String	1:26:59.061 PM
16	OperationMode	2	Byte	1:24:58.812 PM
17	ProcessingTimes	Double click to display value	ExtensionObject	1:26:59.061 PM
18	ResultCounters	Double click to display value	ExtensionObject	1:24:58.812 PM
19	ResultEvaluation	1 (OK)	Int32	1:24:58.812 PM
20	ResultEvaluationCode	0	Int64	1:24:58.812 PM
21	ResultEvaluationDetails	"en", "OK TIGHTENING"	LocalizedText	1:24:58.812 PM
22	ResultId	2924FC62-C20F-C44C-A8D4-981B77B89A73	String	1:26:59.061 PM
23	ResultState	1	Int32	1:24:58.812 PM
24	SequenceNumber	4	UInt64	1:26:59.061 PM

10. Subscribe to Result or Other Events

- Connect to the OPC UA Server using UaExpert or any other OPC UA Client.
- Subscribe to “Server” Object. In UaExpert, Drag and Drop the “Server” Object in the following **Configuration Window** as shown below.



- Select the “Simple Event” checkbox and it should show as checked: ✔
- Generate a **new Result**, and the Result will be listed in the **Events Window** as shown below.



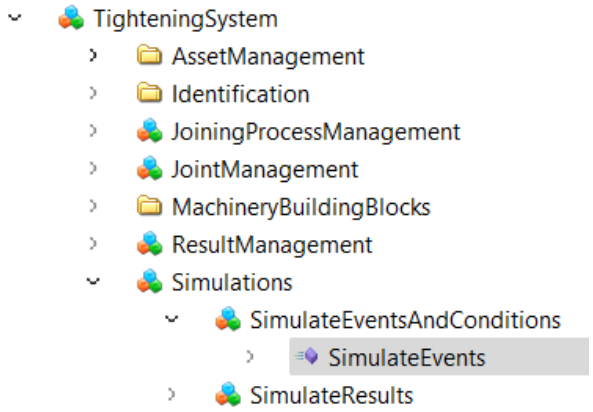
11. Validate if the Client Supports Required OPC UA Features

- OPC UA Client shall support OPC UA Extension Objects (Custom Structure Types) to consume the data as per the OPC UA IJT Standard.
- The quick test to validate if the Client application supports extension objects is to subscribe to Result variable or Result Event and visualize if the data is readable from the Client.
- Refer to previous sections on how to subscribe to Result variable or Result Event.

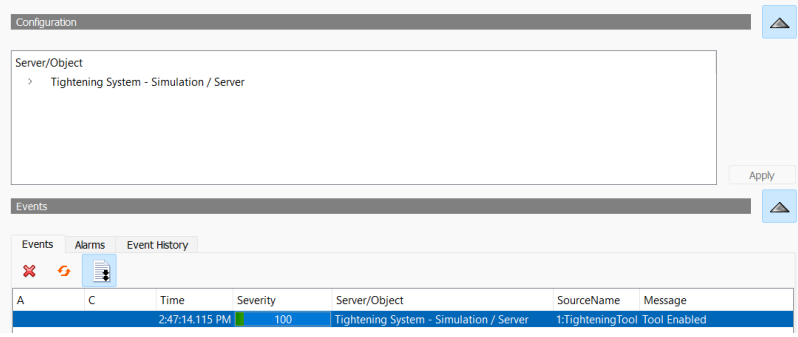
12. Event Data

Only a few events are added to the simulator. Execute the **SimulateEvents method** as shown below to generate a few types of events.

Note: Additional types of events will be added to the simulator in the future. The **content** of the Events would be similar to any type of event from a joining system.



Example Events View

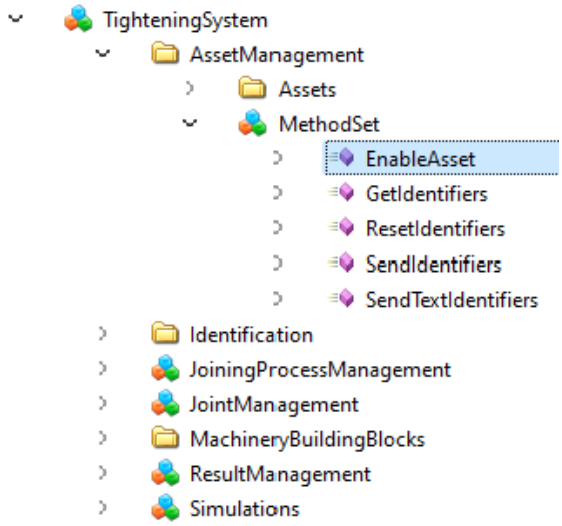


3:JoiningSystemEventContent/3:AssociatedEntities	Array of ExtensionObject
[0]	EntityDataType
Name	ProductInstanceUri
Description	ProductInstanceUri of the Asset
EntityId	www.atlascopco.com/wsqp082020
EntityOriginId	Null
IsExternal	True
EntityType	4
3:JoiningSystemEventContent/3:EventCode	0
3:JoiningSystemEventContent/3:EventText	"en", "Tool Enabled"
3:JoiningSystemEventContent/3:JoiningTechnology	"en", "Tightening"
3:JoiningSystemEventContent/3:ReportedValues	Array of ExtensionObject
ConditionClassId	NodeId
NamespaceIndex	0
IdentifierType	Numeric
Identifier	11166 [SystemConditionClassType]
ConditionClassName	"en", "SystemConditionClassType"
ConditionSubClassId	Array of NodeId
[0]	NodeId
NamespaceIndex	3
IdentifierType	Numeric
Identifier	1024
ConditionSubClassName	Array of LocalizedText
[0]	"en", "AssetEnabledConditionClassType"
EventId	len=1, 0x39

13. Commands Example

An example simulation of **EnableAsset** is provided. It takes the input of the ProductInstanceUri of the Tool.

Few error cases can be simulated when the input argument is empty or invalid. A respective error is shown in the output arguments.



Example Command View