TransSECS OPCUA Guide for a GEM Host

Using TransSECS Servers to Create an OPCUA SECS/GEM Host Application

TransSECS Guide Overview

This documentation will guide your through creating a GEM Host application in TransSECS. The host application will monitor and control your tool.

TransSECS will deploy the application as an OPCUA server. You can then use your OPCUA client to add logic to the interface – monitoring values, making decisions with those values and sending and receiving SECS messages.
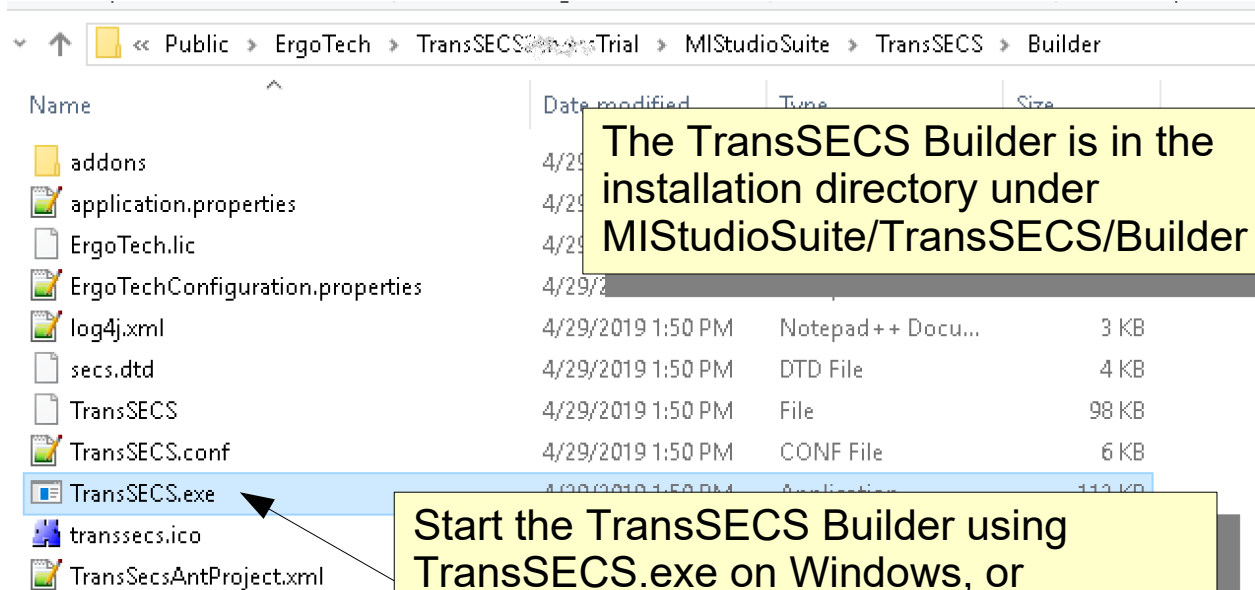
The example uses the *StandAloneGEMTool* provided with TransSECS as a tool simulator. The *StandAloneGEMTool* behaves just as a real GEM tool would behave and you can use it for testing before monitoring and controlling your real tool.

Installing TransSECS
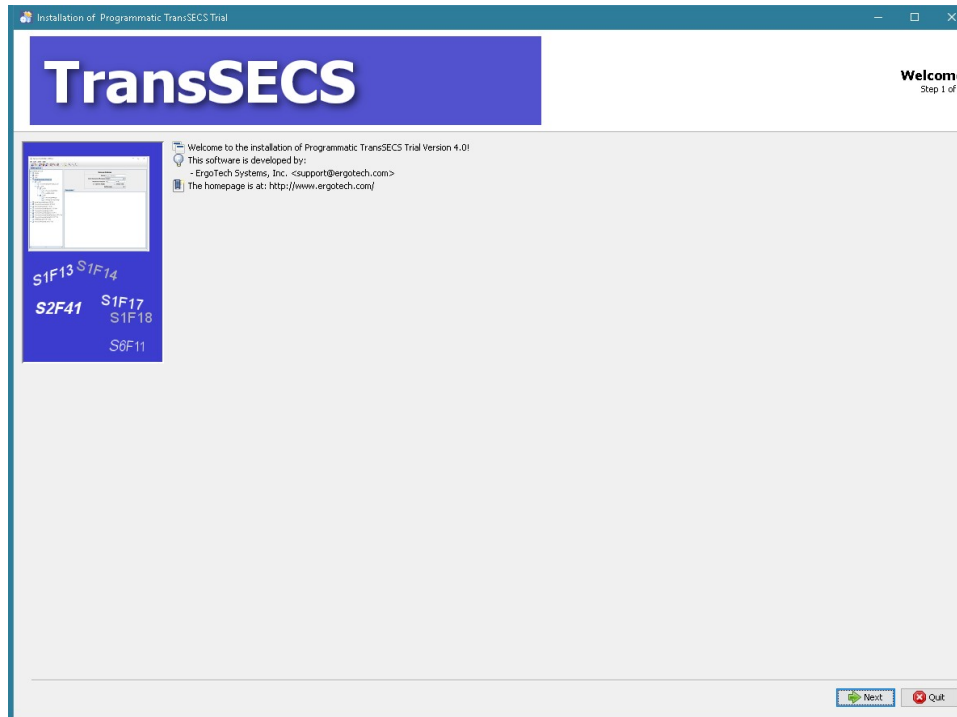
Double click on the installer and click **Next** when prompted. Once installed, start the TransSECS Builder application (MIStudioSuite/TransSECS/Builder/TransSECS.exe)
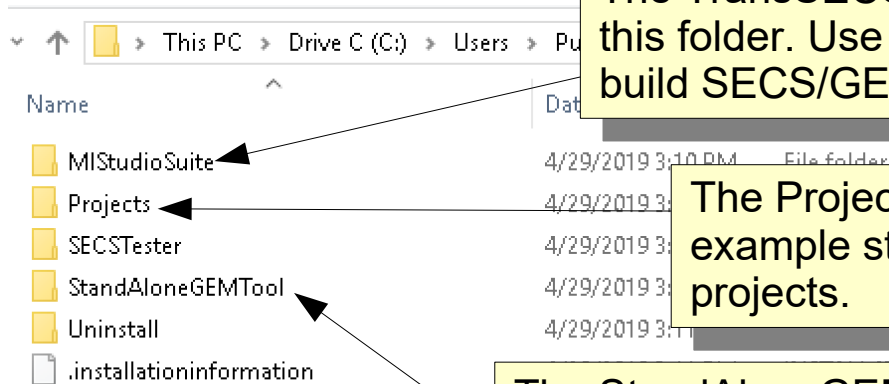
Start the TransSECS Builder

<< Public > ErgoTech > TransSECS░░░Trial > MIStudioSuite > TransSECS > Builder

Name	Date modified	Type	Size
addons	4/29		
application.properties	4/29		
ErgoTech.lic	4/29		
ErgoTechConfiguration.properties	4/29/2		
log4j.xml	4/29/2019 1:50 PM	Notepad++ Docu...	3 KB
secs.dtd	4/29/2019 1:50 PM	DTD File	4 KB
TransSECS	4/29/2019 1:50 PM	File	98 KB
TransSECS.conf	4/29/2019 1:50 PM	CONF File	6 KB
TransSECS.exe	4/29/2019 1:50 PM	Application	112 KB
transsecs.ico			
TransSecsAntProject.xml			

The TransSECS Builder is in the installation directory under MIStudioSuite/TransSECS/Builder

Start the TransSECS Builder using TransSECS.exe on Windows, or TransSECS on Linux systems. For Windows systems there may be a shortcut you can use on your Desktop or Start Menu.

TransSECS Installed

> This PC > Drive C (C:) > Users > Pu

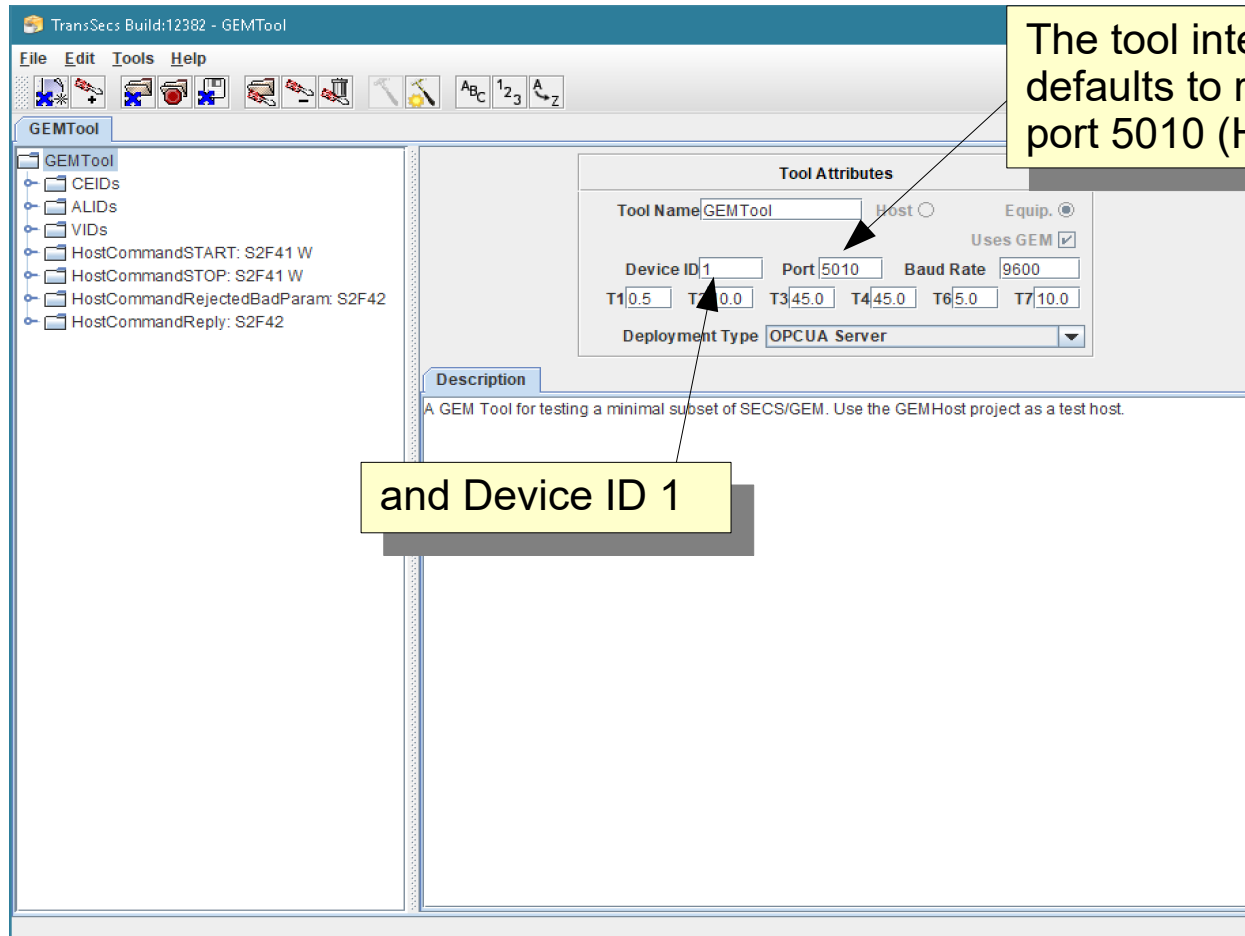| Name	Dat
MIStudioSuite	4/29/2019 3:10 PM File folder
Projects	4/29/2019 3:
SECSTester	4/29/2019 3:
StandAloneGEMTool	4/29/2019 3:
Uninstall	4/29/2019 3:
.installationinformation	

The TransSECS Editor is in this folder. Use the "Builder" to build SECS/GEM deployments.

The Projects directory contains example starting SECS/GEM projects.

The StandAloneGEMTool is a GEMTool simulator specifically for testing the example GEMHost project.

The installation directory contains the main MIStudioSuite directory, along with the Projects directory (where you will find GEMTool and GEMHost projects), and two helpful utilities, StandAloneGEMTool and the SECSTester, which is a simple GEM Host for basic tool characterization.

The GEMTool example project will be loaded when you start the TransSECS Builder for the first time

TransSecs Build:12382 - GEMTool

File Edit Tools Help

GEMTool

- GEMTool
  - CEIDs
  - ALIDs
  - VIDs
  - HostCommandSTART: S2F41 W
  - HostCommandSTOP: S2F41 W
  - HostCommandRejectedBadParam: S2F42
  - HostCommandReply: S2F42

Tool Attributes

Tool Name GEMTool   Host ○   Equip. ◉

Uses GEM ☑

Device ID 1   Port 5010   Baud Rate 9600

T1 0.5   T2 0.0   T3 45.0   T4 45.0   T6 5.0   T7 10.0

Deployment Type OPCUA Server ▼

Description

A GEM Tool for testing a minimal subset of SECS/GEM. Use the GEMHost project as a test host.

> The tool interface defaults to running on port 5010 (HSMS)

> and Device ID 1

Load the GEMHost Project

Use the File "Open Project" menu to open another project

Load the GEMHost Project

Select the GEMHost project
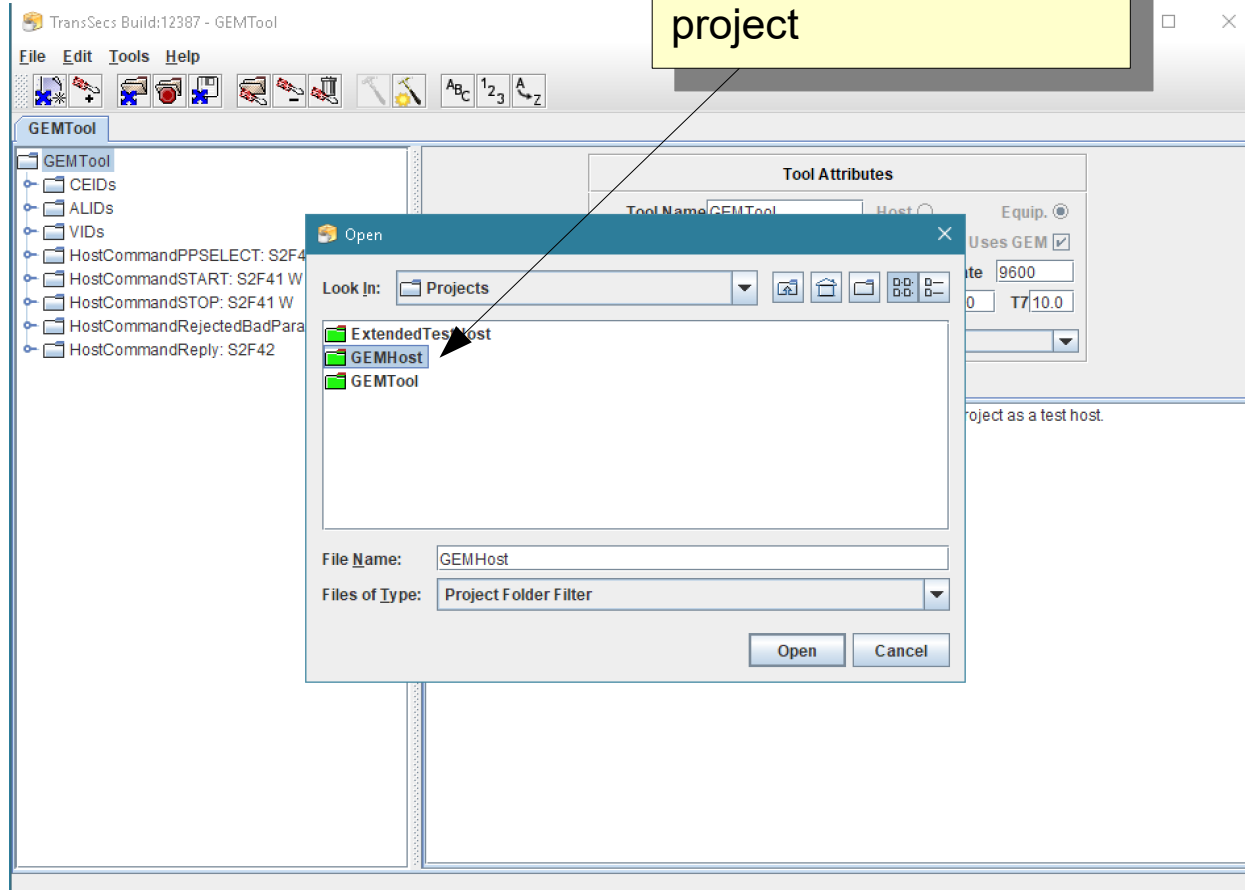
Modifying the GEMHost for Process Tool Data Collection

Once the GEMHost is loaded, you may customize it for data collection from your process tool. The **TransSECS GEM Host Tool Characterization Guide** goes into greater detail on easily importing the process tools VIDs and CEIDs from the tool and the procedure to set up reports (configuring RPTIDs).

https://www.ergotech.com/files/guides/TransSECSGEMHostToolCharacterizationGuide.pdf

Automatic data collection in TransSECS GEM Host applications is accomplished entirely from data in event reports (S6F11s). Each of the VIDs defined in the report through the report definition setup will automatically be updated in the OPCUA variables/vids node when a new event report is received by the host containing those VIDs, as well as the associated CEID and RPTID values.

Likewise, when an alarm (S5F1) is received by the host, the last alarm data in the OPCUA Server will be updated with this alarm data from the message.

Select OPCUA Server for the Deployment Type
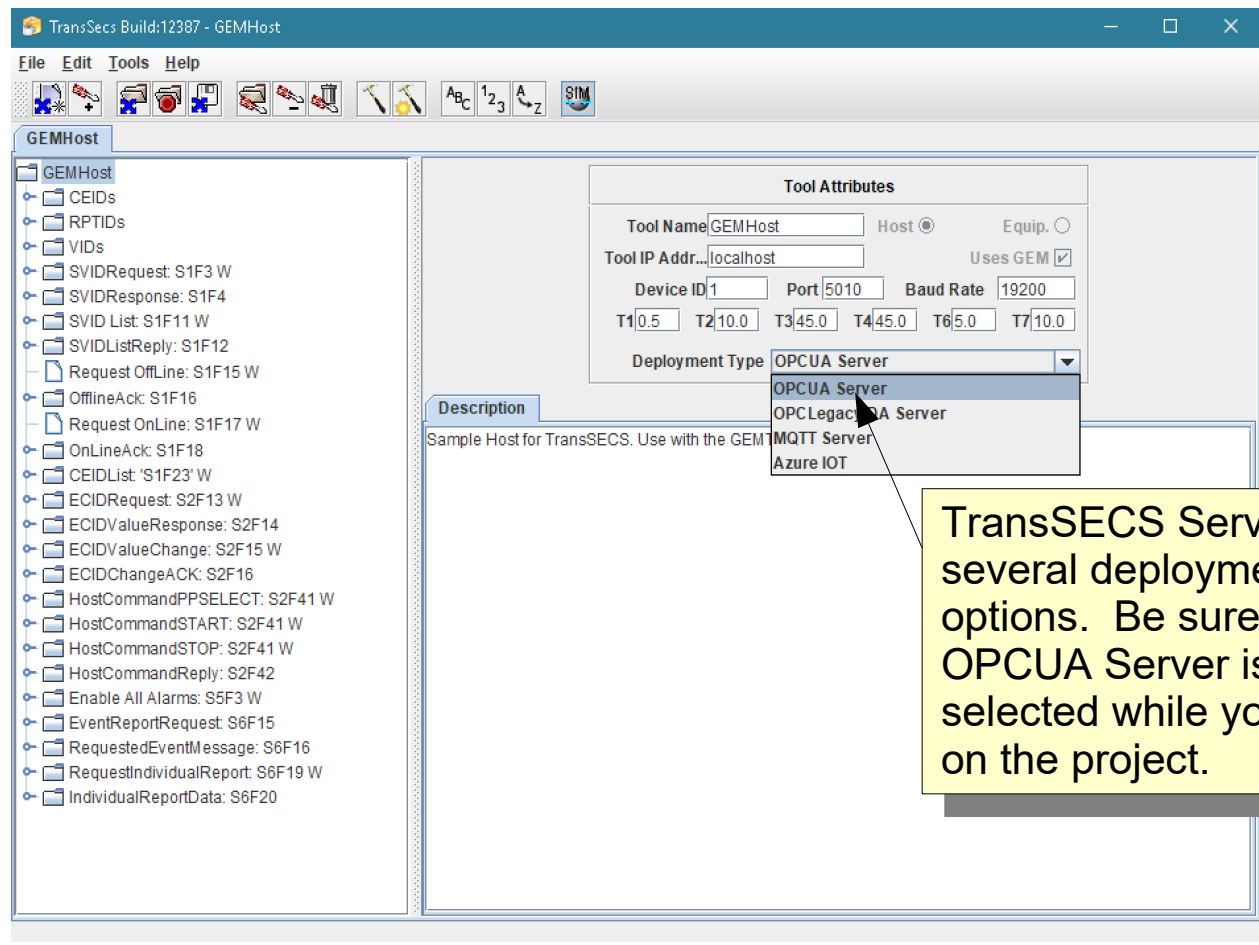
TransSECS Servers has several deployment options. Be sure OPCUA Server is selected while you work on the project.

Data Collection

The primary data collection mechanism for SECS/GEM is to associate value with "Events". An Event is defined by the tool and indicates something of interest has happened, for example, loading, starting, complete, recipe changed, etc.

Values (VIDs) are grouped into "Reports", represented by Report IDs (PRTIDs) these reports as then associated with one ore more Events. The following pages show you do this in TransSECS.

Defining RPTIDs

ErgoTech Systems, Inc.

After the VIDs and CEIDs are obtained from the tool, you can add RPTIDs.

TransSecs Build:12387 - GEMHost

File   Edit   Tools   Help

GEMHost

- GEMHost
  - CEIDs
  - RPT
    - **Add RPTID**
    - **Online RPTIDs from Tool**
    - **Remove All RPTIDs**
    - **Sort By Name**
    - **Sort By Id**
    - **Sort By Name**
    - **Sort By Id**
  - VIDs
  - CEID
  - ECID
  - ECID
  - ECID
  - ECID
  - Enal
  - EventReportRequest: S6F15
  - HostCommandPPSELECT: S2F41 W
  - HostCommandReply: S2F42
  - HostCommandSTART: S2F41 W
  - HostCommandSTOP: S2F41 W
  - IndividualReportData: S6F20
  - OfflineAck: S1F16
  - OnLineAck: S1F18
  - Request OffLine: S1F15 W
  - Request OnLine: S1F17 W
  - RequestedEventMessage: S6F16
  - RequestIndividualReport: S6F19 W
  - SVID List: S1F11 W
  - SVIDListReply: S1F12
  - SVIDRequest: S1F3 W
  - SVIDResponse: S1F4

Right-Click on the RPTID node to add a RPTID

Defining RPTIDs

A RPTID has been added with a default name "RPTID0"

Defining RPTIDs

The RPTID has been renamed to "RPTID100" and assigned a RPTID of 100

These four highlighted VIDs have been selected for the report

Press this button to complete adding the selected VID items to the Report VIDs list

You may choose any numeric id for the report (RPTID) unless the process tool manual documents a restriction, such as only an unsigned integer, but it's usually best to not choose one that is the same as a CEID or VID.

Defining RPTIDs

The RPTID100 definition is complete

Next we will associate this report with the LOADED event

Defining RPTIDs

Select a CEID to be associated with one or more reports (RPTIDs)

We have one RPTID defined which is selected for this event

Build the Project using the Hammer/Star Button

When the "Compilation" popup closes the build is complete. This may take a minute or so.
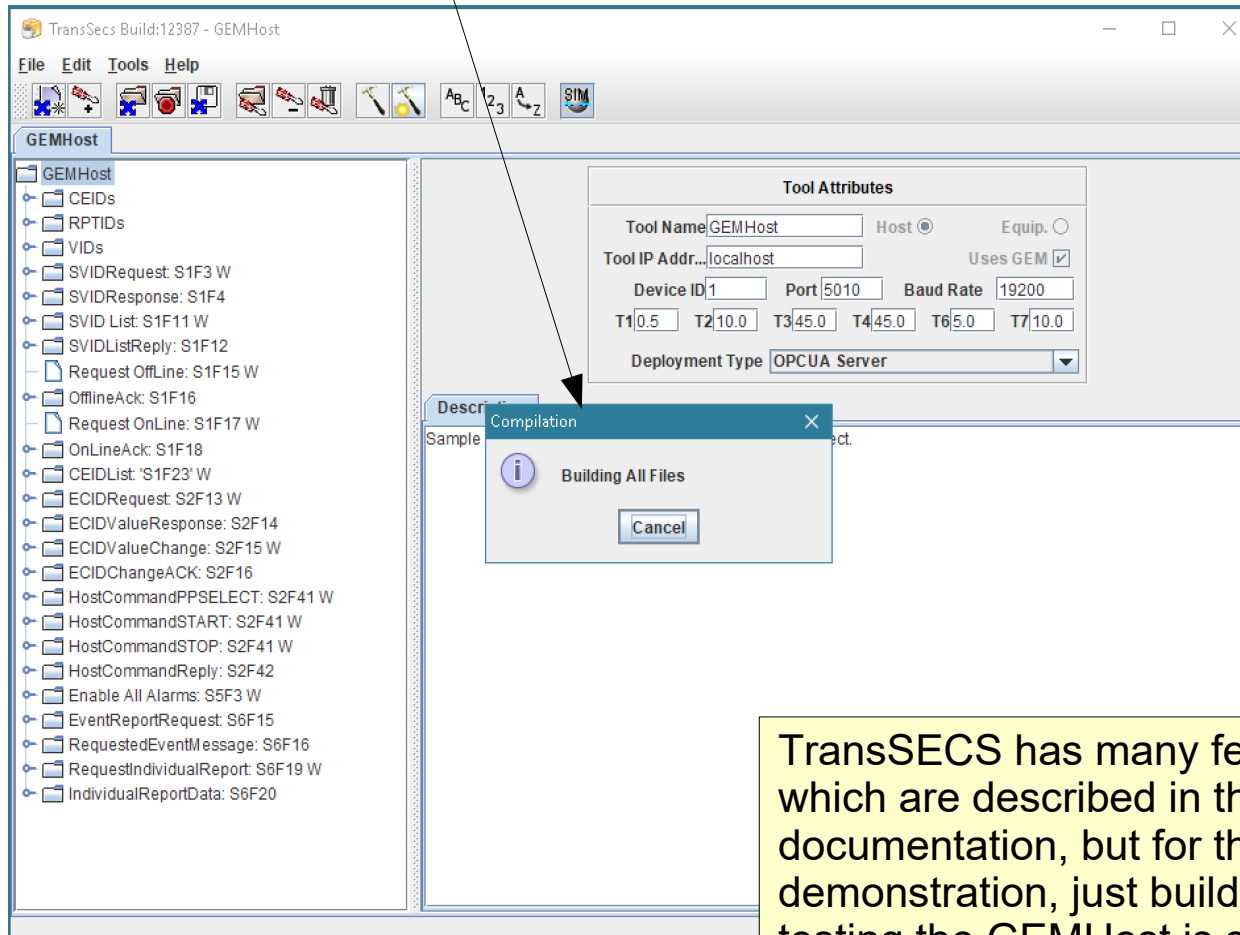
TransSECS has many features which are described in the full documentation, but for this demonstration, just building and testing the GEMHost is sufficient.

After the code is generated the OPCUA Server, with the Tags from TransSECS, will be in the Projects/GEMHost/OPCUA directory.

Everything you need to run on Windows is in this directory. For Linux systems you will need install rxtxSerial on the system and make appropriate changes to the run.sh file.

You may need to edit the path to the jre in run.bat if you move the deployment location.

Run the server with the run.bat file (or run.sh on Linux).
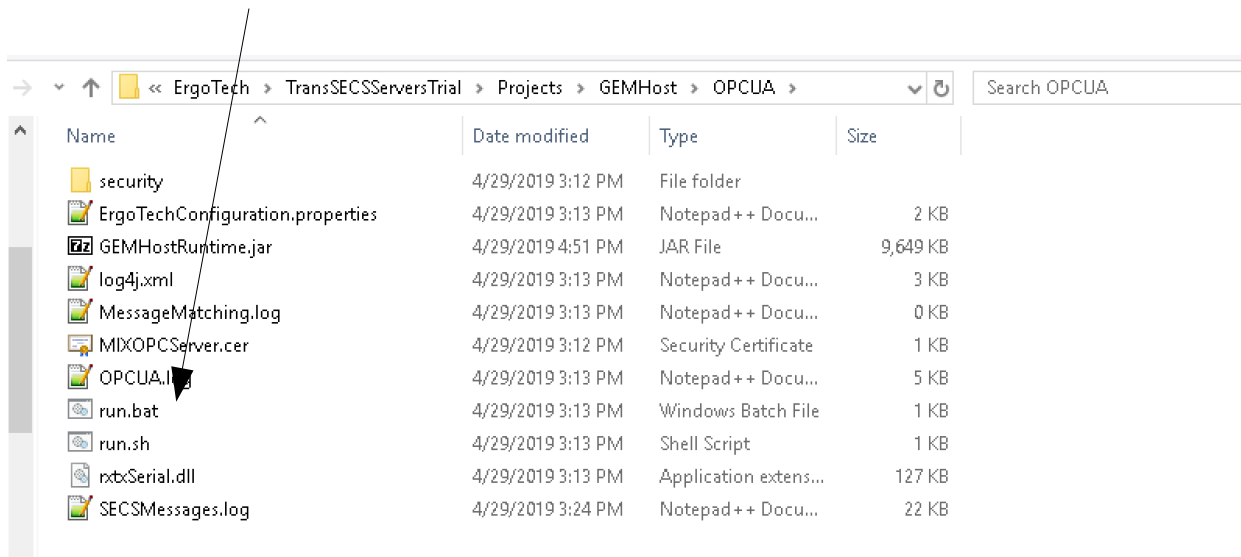
Run the SECS/GEM Interface as an OPCUA Server


```
C:\WINDOWS\system32\cmd.exe                                                    —    □    ×

Started GEMHost connecting to localhost on port 5010 with device id 1
at ip address localhost2019-05-01 17:52:29.149 [createFolderNode( gemhost:gemhost]2019-05-01 17:52:29.149
2019-05-01 17:52:29.149 [createFolderNode( ecidrequest:gemhost/ecidrequest]2019-05-01 17:52:29.149
2019-05-01 17:52:29.165 [creatingNode( ecid:deploy.GEMHost.ECIDRequest periodic false]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [vib.utils.VIBOPC > setValueObject( PrimaryOutPort:0 ) setValueObject="2000" )]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [creatingNode( responsestatus:deploy.GEMHost.ECIDRequest periodic false]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [creatingNode( errorstatus:deploy.GEMHost.ECIDRequest periodic false]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [creatingNode( goodstatus:deploy.GEMHost.ECIDRequest periodic false]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [creatingNode( sendmessage:deploy.GEMHost.ECIDRequest periodic false]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [createFolderNode( variables:gemhost/variables]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [createFolderNode( alarm:gemhost/variables/alarm]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [creatingNode( alarmset:com.ergotech.transsecs.secs.HostIDBean periodic false]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [vib.utils.VIBOPC > setValueObject( PrimaryOutPort:0 ) setValueObject="0" )]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [createFolderNode( vid:gemhost/variables/vid]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [creatingNode( wafercount:com.ergotech.transsecs.secs.HostIDBean periodic false]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [vib.utils.VIBOPC > setValueObject( PrimaryOutPort:1 ) setValueObject="0" )]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [creatingNode( alarm:com.ergotech.transsecs.secs.HostIDBean periodic false]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [vib.utils.VIBOPC > setValueObject( PrimaryOutPort:2 ) setValueObject="" )]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [creatingNode( altx:com.ergotech.transsecs.secs.HostIDBean periodic false]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [vib.utils.VIBOPC > setValueObject( PrimaryOutPort:3 ) setValueObject="" )]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [createFolderNode( ceid:gemhost/variables/ceid]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [creatingNode( started:com.ergotech.transsecs.secs.HostIDBean periodic false]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [vib.utils.VIBOPC > setValueObject( PrimaryOutPort:4 ) setValueObject="" )]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 Unknown type a
2019-05-01 17:52:29.165 [creatingNode( enabledalarms:com.ergotech.transsecs.secs.HostIDBean periodic false]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [vib.utils.VIBOPC > setValueObject( PrimaryOutPort:5 ) setValueObject="[]" )]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [creatingNode( localremotestate:com.ergotech.transsecs.secs.HostIDBean periodic false]2019-05-01 17:52:29.165
2019-05-01 17:52:29.165 [vib.utils.VIBOPC > setValueObject( PrimaryOutPort:6 ) se
2019-05-01 17:52:29.165 [createFolderNode( rptid:gemhost/variables/rptid]2019-05-
2019-05-01 17:52:29.165 [creatingNode( rptid103:com.ergotech.transsecs.secs.HostI
2019-05-01 17:52:29.165 [vib.utils.VIBOPC > setValueObject( PrimaryOutPort:7 ) se
2019-05-01 17:52:29.180 [creatingNode( rptid100:com.ergotech.transsecs.secs.HostI
```

When you use the generated run.bat to start the OPCUA Server, the GEMHost application will make a connection to a tool running on Port 5010 and Device ID 1.

Run the SECS/GEM Interface as an OPCUA Server

The default SECS/GEM OPCUA Server endpoint URL is opc.tcp://127.0.0.1:12686/MIXOPCServer

Review and Next Steps for Testing the OPCUA Host Server

The GEMHost project has been built as an OPCUA Server. When this GEMHost is running it will automatically attempt to find and connect to a process tool at ip address "localhost" using port 5010 and device id 1.

Now that the OPCUA server is running as a host application we can do two more things:

1) Allow the host to connect to a process tool simulator at "localhost" port 5010, device id 1for testing

and

2) Connect an OPCUA Client to the OPCUA Server so we can send messages to the process tool from the host and also get data from the tool

The next part of this guide goes through both of these steps, first connecting a tool simulator (StandAloneGEMTool), then using an OPCUA client to test the OPCUA Server.

Testing your OPCUA GEMHost with the StandAloneGEMTool

Use the tool simulator in the installation StandAloneGEMTool to test the GEMHost project. It is set up to use port 5010 and device id 1.

Browse to the installation directory and open the StandAloneGEMTool folder.

ErgoTechConfiguration.properties	4/29/2019 1:50 PM	Notepad++ Docu...	2 KB
ErgoTechStandAloneGEMToolLicense.txt	4/29/2019 1:50 PM	Notepad++ Docu...	12 KB
GEMToolStandAlone.jar	4/29/2019 1:50 PM	JAR File	3,863 KB
log4j.xml	4/29/2019 1:50 PM	Notepad++ Docu...	3 KB
MessageMatching.log	4/29/2019 3:25 PM	Notepad++ Docu...	0 KB
ProjectLog.log	4/29/2019 3:25 PM	Notepad++ Docu...	0 KB
run.bat	4/29/2019 1:50 PM	Windows Batch File	1 KB
run.sh	4/29/2019 1:50 PM		
SECSMessages.log			
sqlite3.dll			
ToolParams.sqlite			

Use run.bat on Windows or run.sh on Linux to start the StandAloneGEMTool Simulator

Testing your OPCUA GEMHost with the StandAloneGEMTool

Leave the StandAloneGEMTool running while you work on the GEMHost. The StandAloneGEMTool starts up in Remote/Online mode, ready for a host connection.

GEMToolStandAlone

GEM Tool
Use the GEMHost for testing

1	Wafer Count
50.12	Temperature
36.69	Gas Flow

50.00 Set Point
ECID 2000
50-400 Allowed Range

Remote   Online

PPID
Lot ID

SELECT ALARM ACTION ▼   Select An Alarm Action

Do Selected Alarm Action
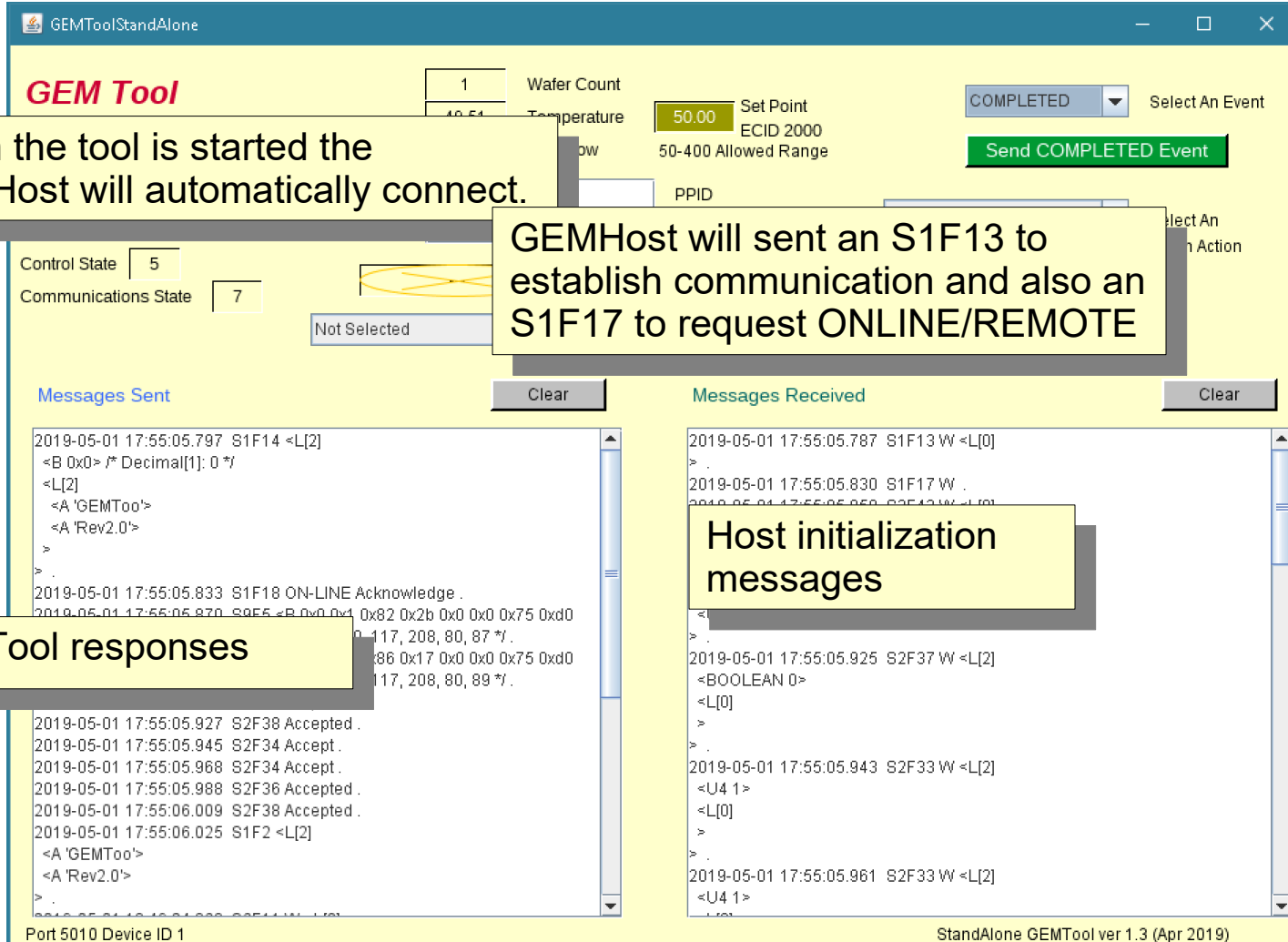
Control State   3
Communications State   1

Last Host Command

Not Selected ▼ ⟵ Select Reply Code For Next Host Command Reply

Messages Sent   Clear   **Messages Received**   Clear

The GEMHost will automatically connect to the StandAloneGEMTool when the host is built and in run mode.

Port 5010 Device ID 1

StandAlone GEMTool ver 1.3 (Apr 2019)

Testing your OPCUA GEMHost with the StandAloneGEMTool

GEMToolStandAlone

GEM Tool

1	Wafer Count
49.51	Temperature
	ow

Set Point **50.00** ECID 2000 50-400 Allowed Range

PPID

COMPLETED ▼ Select An Event

Send COMPLETED Event

Select An Action

When the tool is started the GEMHost will automatically connect.

GEMHost will sent an S1F13 to establish communication and also an S1F17 to request ONLINE/REMOTE

Control State | 5
Communications State | 7

Not Selected

Messages Sent   Clear   **Messages Received**   Clear

```
2019-05-01 17:55:05.797  S1F14 <L[2]
 <B 0x0> /* Decimal[1]: 0 */
 <L[2]
  <A 'GEMToo'>
  <A 'Rev2.0'>
 >
> .
2019-05-01 17:55:05.833  S1F18 ON-LINE Acknowledge .
2019-05-01 17:55:05.870  S9F5 <B 0x0 0x1 0x82 0x2b 0x0 0x0 0x75 0xd0
              0, 117, 208, 80, 87 */.
              86 0x17 0x0 0x0 0x75 0xd0
              117, 208, 80, 89 */.

2019-05-01 17:55:05.927  S2F38 Accepted .
2019-05-01 17:55:05.945  S2F34 Accept .
2019-05-01 17:55:05.968  S2F34 Accept .
2019-05-01 17:55:05.988  S2F36 Accepted .
2019-05-01 17:55:06.009  S2F38 Accepted .
2019-05-01 17:55:06.025  S1F2 <L[2]
 <A 'GEMToo'>
 <A 'Rev2.0'>
> .
```

```
2019-05-01 17:55:05.787  S1F13 W <L[0]
> .
2019-05-01 17:55:05.830  S1F17 W .
2019-05-01 17:55:05.960  S2F43 W <L[0]

>
> .
2019-05-01 17:55:05.925  S2F37 W <L[2]
 <BOOLEAN 0>
 <L[0]
 >
> .
2019-05-01 17:55:05.943  S2F33 W <L[2]
 <U4 1>
 <L[0]
 >
> .
2019-05-01 17:55:05.961  S2F33 W <L[2]
 <U4 1>
```

Host initialization messages

Tool responses

Port 5010 Device ID 1   StandAlone GEMTool ver 1.3 (Apr 2019)

Testing your OPCUA GEMHost with the StandAloneGEMTool

GEMHost will also set up reports which have been defined in the GEMHost project, and also link these reports to events. It also enables all alarms and events in the tool.

Host event and report set-up messages

Tool responses

Testing your OPCUA GEMHost with the StandAloneGEMTool

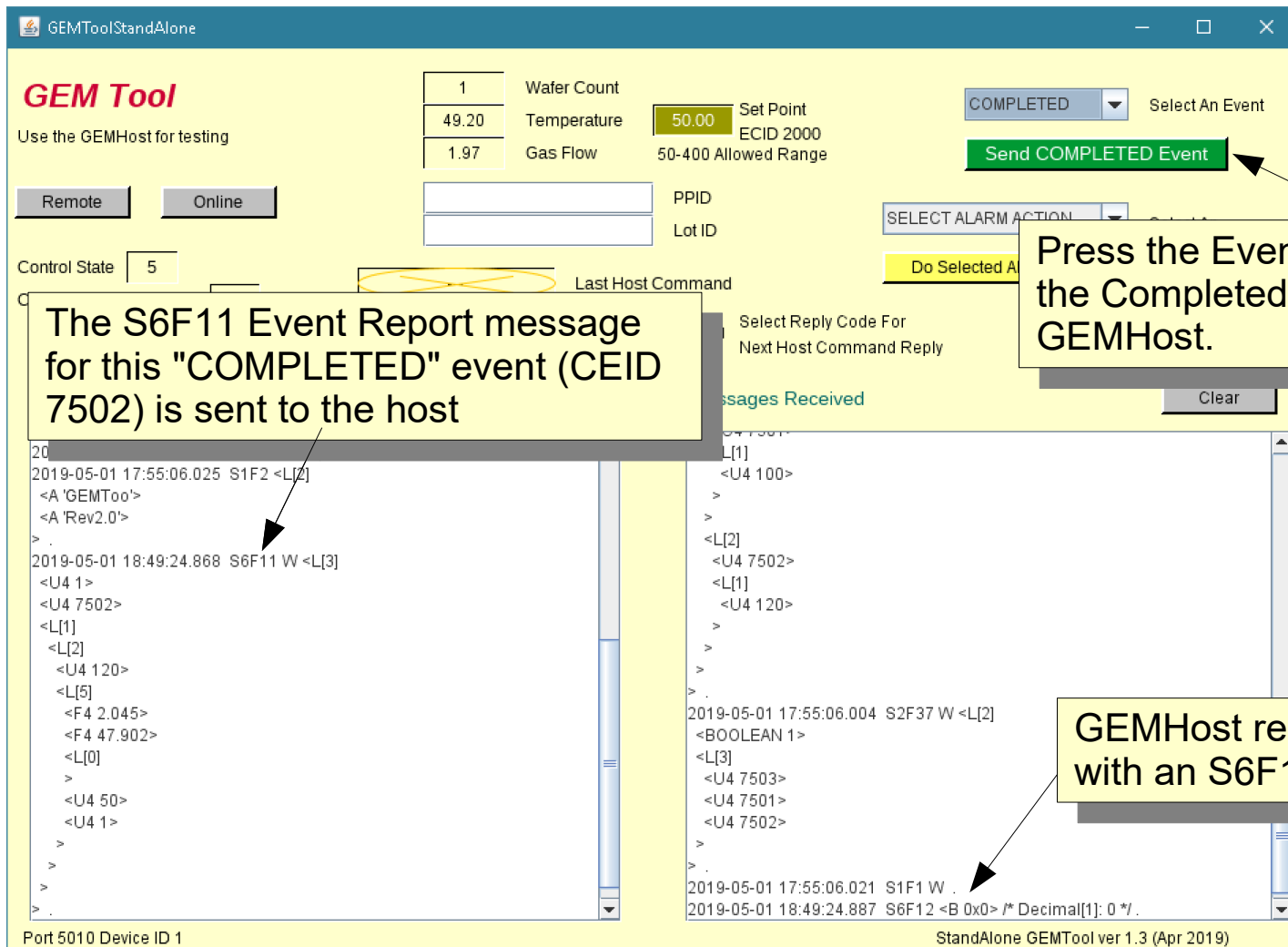The StandAloneGEMTool can simulate an event and send an S6F11 Event Report to the host

Select an event to trigger for testing, for example "COMPLETED"

Testing your OPCUA GEMHost with the StandAloneGEMTool

GEM Tool
Use the GEMHost for testing

1	Wafer Count
49.20	Temperature
1.97	Gas Flow

Set Point 50.00
ECID 2000
50-400 Allowed Range

COMPLETED ▼ Select An Event

Send COMPLETED Event

Remote   Online

PPID
Lot ID

SELECT ALARM ACTION ▼
Do Selected A...

Control State 5

Last Host Command

Press the Event send button to s
the Completed Event to the
GEMHost.

The S6F11 Event Report message for this "COMPLETED" event (CEID 7502) is sent to the host

Select Reply Code For
Next Host Command Reply

ssages Received

Clear

```
20
2019-05-01 17:55:06.025  S1F2 <L[2]
 <A 'GEMToo'>
 <A 'Rev2.0'>
> .
2019-05-01 18:49:24.868  S6F11 W <L[3]
 <U4 1>
 <U4 7502>
 <L[1]
  <L[2]
   <U4 120>
   <L[5]
    <F4 2.045>
    <F4 47.902>
    <L[0]
    >
    <U4 50>
    <U4 1>
   >
  >
 >
> .
```

```
L[1]
  <U4 100>
 >
>
<L[2]
 <U4 7502>
 <L[1]
  <U4 120>
 >
 >
>
> .
2019-05-01 17:55:06.004  S2F37 W <L[2]
 <BOOLEAN 1>
 <L[3]
  <U4 7503>
  <U4 7501>
  <U4 7502>
 >
>
> .
2019-05-01 17:55:06.021  S1F1 W .
2019-05-01 18:49:24.887  S6F12 <B 0x0> /* Decimal[1]: 0 */ .
```

GEMHost responds with an S6F12

Port 5010 Device ID 1

StandAlone GEMTool ver 1.3 (Apr 2019)

Set up an OPCUA Client Connection

The next step to testing the GEMHost is to use an OPCUA Client. Examples in this guide uses the UAExpert OPCUA Test Client which can be downloaded from:

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

Please use the TransSECS OPCUA Certificate Exchange Guide for a detailed example of connecting an OPCUA Client to the running SECS/GEM OPCUA Server.

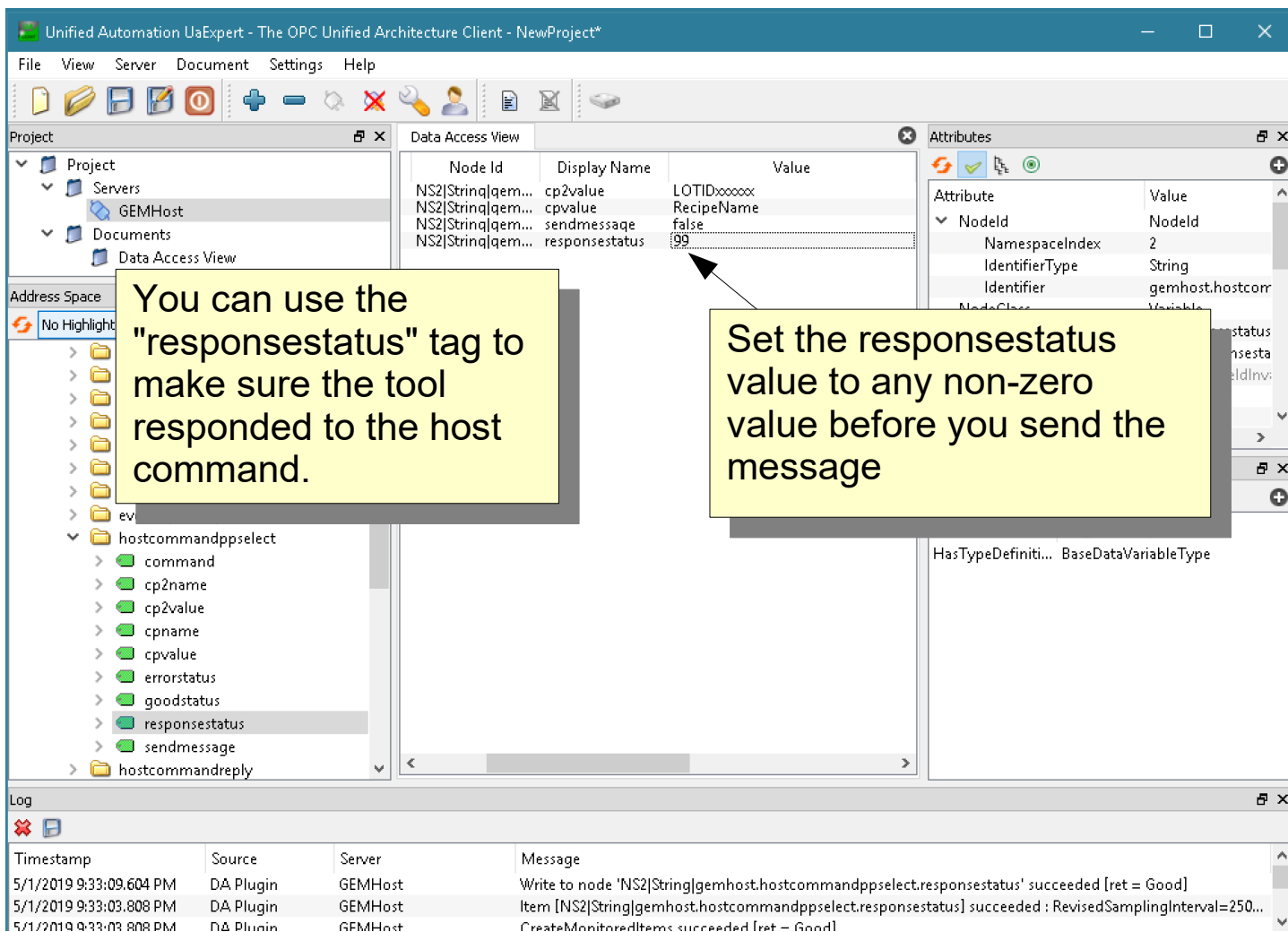# OPCUA Client Test – Send a Host Message to a Tool (PP-Select)

Use your OPCUA Client to set variables in the message.

Use the "sendMessage" tag to send the host message to the tool. Set this value in your OPCUA client to a non-false value to send the message.

OPCUA Client Test – Send a Host Message to a Tool

OPCUA Client Test – Send a Host Message to a Tool

Use the "sendMessage" tag to send the host message to the tool.

You can use the "responsestatus" tag to make sure the tool responded to the host command.

The responsestatus tag will revert to "0" after the response has been received.

Once the response has been received you can use your OPCUA Client to read values from the response message to check for errors from the tool.

OPCUA Client Test – Send a Host Message to a Tool

ErgoTech Systems, Inc.

Here are the two CPValues sent in the host command PP-Select

Select the host command reply code for testing (e.g., 0=OK) before sending the host command

The StandAloneGEMTool will display the command received

You can see the PPSELECT host command received by the StandAloneGEMTool, as well as the reply from the tool.

Use th...

Wafer Count

Temperature

50.00 Set Point
ECID 2000
50-400 Allowed Range

2.08 Gas Flow

SELECT EVENT ▼ Select An Event

Send LOADED Event

Remote   Online

RecipeName   PPID

LOTIDxxxxxx   Lot ID

SELECT ALARM ACTION ▼ Select An Alarm Action

Do Selected Alarm Action

Control State [5]

Communications State [7]

PP-SELECT   Last Host Command

0=OK Command Performed ▼   Select Reply Co...
Next Host Com...

Messages Sent   **Clear**   Messages Received

```
2019-05-02 12:45:33.109  S2F42 <L[2]
 <B 0x0> /* Decimal[1]: 0 */
 <L[0]
 >
 > .
```

```
2019-05-02 12:45:33.098  S2F41 W <L[2]
 <A 'PP-SELECT'>
 <L[2]
  <L[2]
   <A 'LOTID'>
   <A 'LOTIDxxxxxx'>
  >
  <L[2]
   <A 'PPID'>
   <A 'RecipeName'>
  >
 >
 > .
```
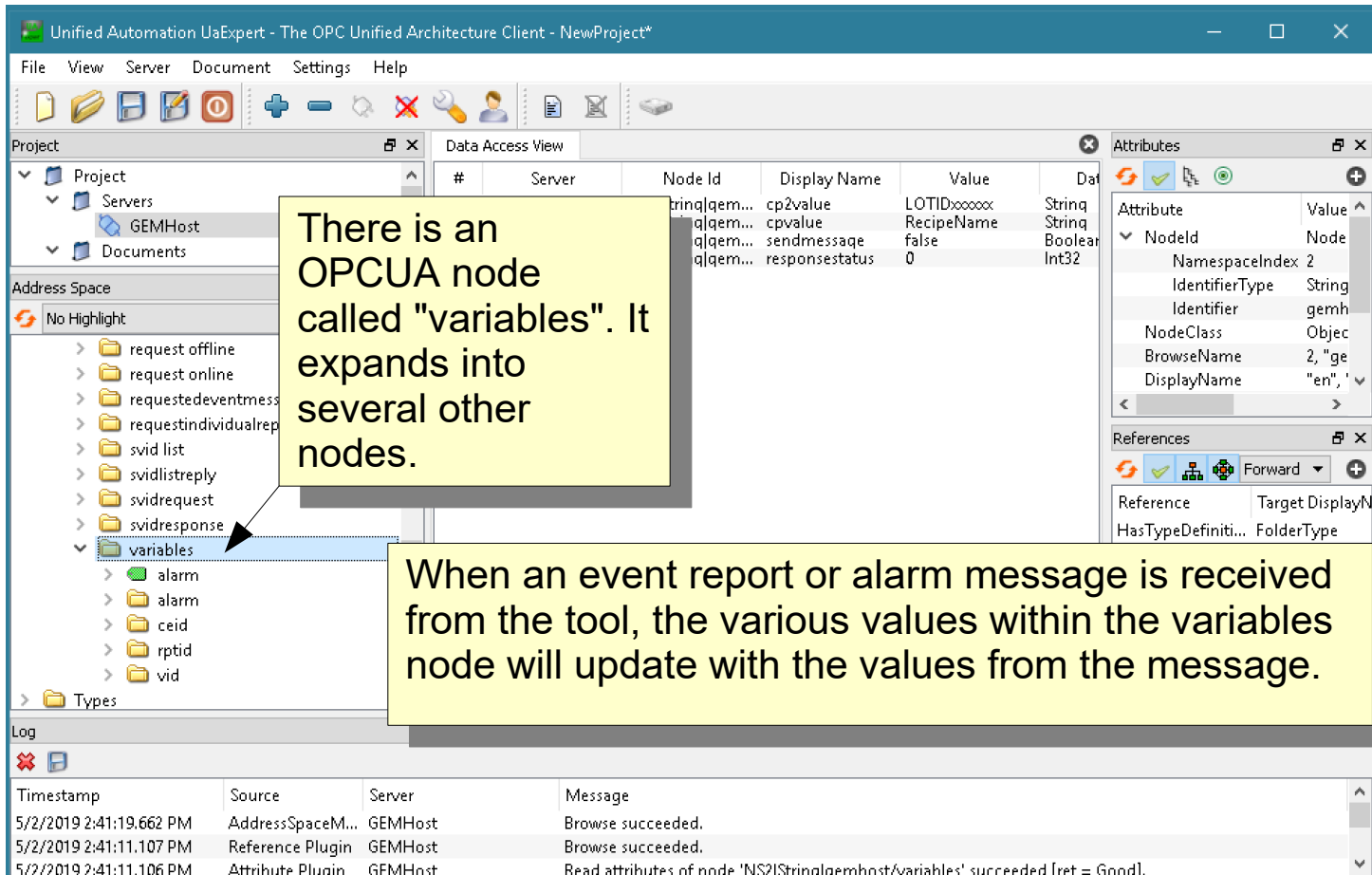
Port 5010 Device ID 1

StandAlone GEMTool ver 1.3 (Apr 2019)

Test Sending an Event Message From the Tool

ErgoTech Systems, Inc.

33.5
35.7
32.2
30.9

Trigger an event on the tool to send an event report to the host.

The LOADED event is selected

Wafer Count

Temperature **50.00** Set Point
ECID 2000
Gas Flow **2.10** 50-400 Allowed Range

LOADED ▼ Select An Event

Send LOADED Event

Remote   Online

LoadedRecipe1234 PPID
LOTIDxxxxxx Lot ID

SE

Press the send event button to trigger the event

Control State 5
Communications State 7

PP-SELECT Last Host Command

0=OK Command Performed ▼ ⬅ Select Reply Code For
Next Host Command Reply

Messages Sent   Clear   **Messages Received**   Clear

```
2019-05-02 12:45:33.109  S2F42 <L[2]
 <B 0x0> /* Decimal[1]: 0 */
 <L[0]
 >
> .
```

```
2019-05-02 12:45:33.098  S2F41 W <L[2]
 <A 'PP-SELECT'>
 <L[2]
  <L[2]
   <A 'LOTID'>
   <A 'LOTIDxxxxxx'>
  >
  <L[2]
   <A 'PPID'>
   <A 'RecipeName'>
  >
 >
> .
```

💡 The report for this event was defined in the GEMHost configuration and was defined and linked to the event by the GEMHost during initialization (initial connection to the tool).

Port 5010 Device ID 1

StandAlone GEMTool ver 1.3 (Apr 2019)

Test Sending an Event Message From the Tool

Test Sending an Event Message From the Tool
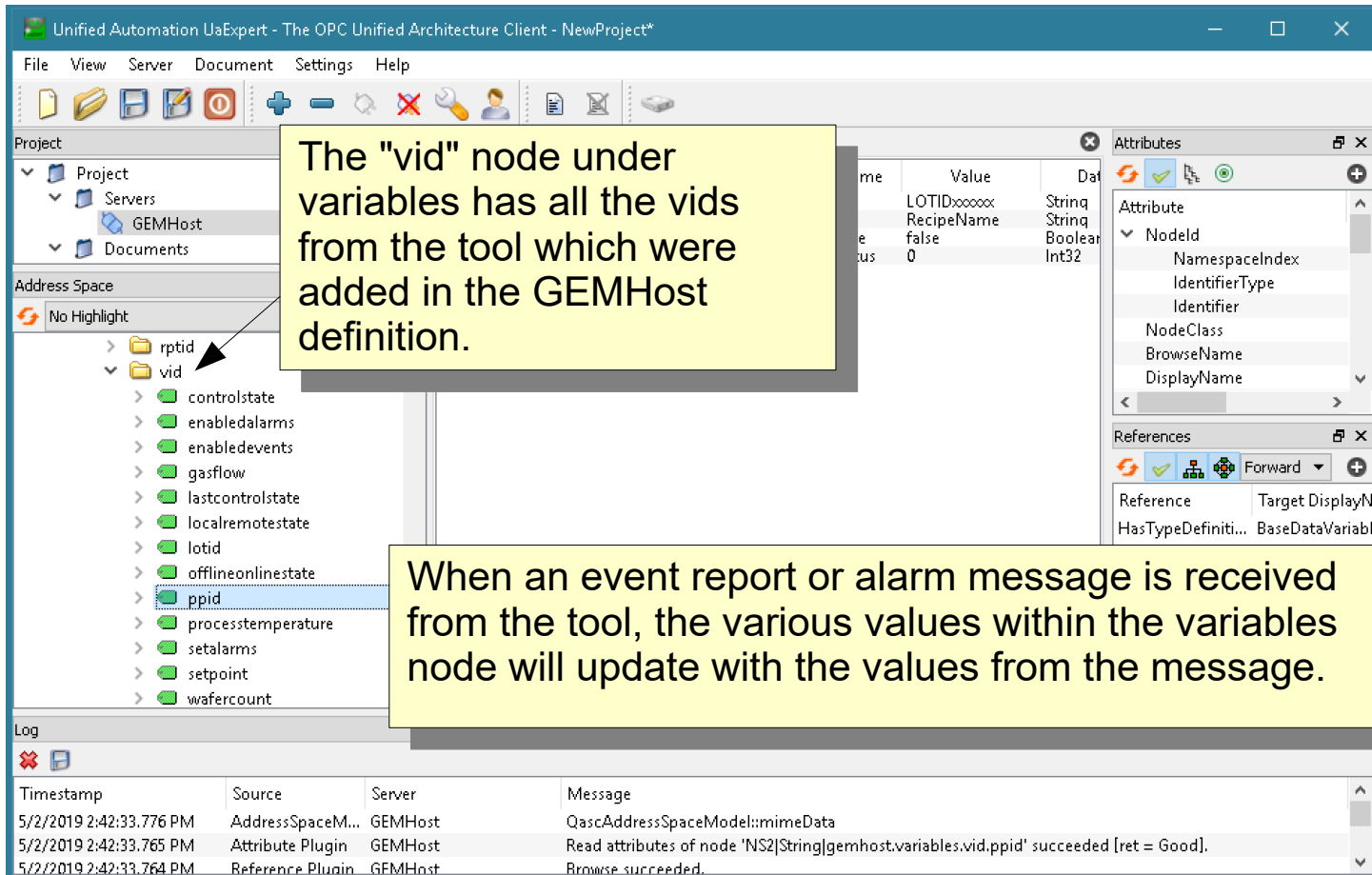
There is an OPCUA node called "variables". It expands into several other nodes.

When an event report or alarm message is received from the tool, the various values within the variables node will update with the values from the message.

Test Sending an Event Message From the Tool

The "vid" node under variables has all the vids from the tool which were added in the GEMHost definition.

When an event report or alarm message is received from the tool, the various values within the variables node will update with the values from the message.

Test Sending an Event Message From the Tool

The latest ppid and lotid values were updated from the LOADED event message data

When an event report or alarm message is received from the tool, the various values within the variables node will update with the values from the message.

JSON Formatted Message Data

JSON formatted data for list items will be demonstrated for the data received in an S1F4 message from the tool.

Use your OPCUA Client to send the S1F3 (SVID Request) with svid id value 1512 (process temperature).

JSON Formatted Message Data

JSON formatted data for list items will be demonstrated for the data received in an S1F4 message from the tool.

Use your OPCUA Client to read the svidlist value from the svidresponse message data.

The value for svidlist is in JSON format, { "values": ["50.422"] , "type":44 }

Notes on OPCUA Servers:
JSON List Formats

List elements are JSON formatted, such as { "values": [ ] }

For example, the recipe list for an S7F20 might look like:

 { "values": ["recipe1","recipe2", "recipe3", "recipe4"] }

There are more things you can do with the JSON format for the lists, including making lists of lists and specifying the SECS format for the data in the list.

Notes on OPCUA Servers:
JSON List Formats

If a specific type is required, it can be specified if needed. For example for the recipe list, the data is ASCII, type 20:

{ "values": ["recipe1","recipe2", "recipe3", "recipe4"], type:20 }

type is defined by the SECS Standard:

00 - List
10 - Binary (can be an array)
11 - Boolean
20 - String
30 - 8 byte floating point
34 - 4 byte floating point

40, 41, 42, 44 - 8, 1,2,4 byte signed integers
50, 51, 52, 54 - 4 byte signed integer

Notes on OPCUA Servers:
JSON List Formats

Each element of the array can have a different type in this format (this might be useful in some message replies with different data types):

```
{"values":[
        {"value":"1.2", "type":"34"},
        {"value":"Test", "type":"20"},
        {"value":[5,6,7,8], "type":"10"},
        {"value":1, "type":"10"}
     ]}
```

Notes on Tool OPCUA Servers: Triggering Events with DVVALs

Events can be triggered with a list of DVVALS, so if you need to do this you can define the list, then use this as the trigger input to the CEID. If you are using DVVALs in your project you would associate the DVVALS when you define your CEID.

DVVALS as a JSON String sent to the trigger of a CEID:

{ "values":[

   { "dvid":WaferCount, dvval:15 },

   { "dvid":23456, dvval:\"Hello World\" }

 ] }

You can specify any VID with either its name (for example WaferCount), or its vid number (for example 23456) as the above example demonstrates.

Notes on OPCUA Servers: Host Commands

If you publish the whole host command, the structure is a list (with the host command), then the list elements for the parameters for the command.

It may look something like:

{ "values": [{ "value":"PP-SELECT", type:"20 } } { "values":
[{ "values": [LOTID] , type:"20 } { "values":
[{ "value":"CHAMBER", type:"20 } } { "value":"1", type:"51 } }] }
{ "values": [PPID] , type:"20 }] }] }

Notes on OPCUA Runtime Configuration

The ErgoTechConfiguration.properties file has some settings which control some aspects of the OPCUA runtime. This file is located in the OPCUA deployment directory.

VIBOPC.NoMethodSupport

This defaults to false. If set to true opcua methods are generated for operations such as sending messages. Some OPCUA clients do not support methods.

VIBOPC.UnderscoreCreatePaths

This defaults to true. This allows you to define vid names with underscores which will generate a tree structure for the vids. For example a vid "robot_xaxis" will generate a tag structure /toolname/vids/robot/xaxis".

That's it.

Now you're ready to program your OPCUA Client application to complete integration of the SECS/GEM Host interface. Hopefully this guide is enough to get you started monitoring and controlling your tool. For the next steps, you'll need to make sure that the IDs in TransSECS match the documentation for your tool.

Use the characterization features of TransSECS for a GEM host to import the vids, etc. from your tool. For more information, see: TransSECS GEM Host Tool Characterization Guide

https://www.ergotech.com/files/guides/TransSECSGEMHostToolCharacterizationGuide.pdf

The host commands for your tool may be a little different, again, check the manual but the changes should be fairly straightforward.

Please let us know if you have any questions!