



Serapis Framework



D2.3d – User Manual

Document Number: X72-HB-10051-004
SERAPIS, Task C67-02, Deliverable D2.3d

Report Number: 72-23-N-350-U, 02.005

Date: 03/06/2024

Author(s): Sharma, P., d'Avezac, M. (Roke Manor Research Limited)

Supplied under terms of Contract No. DSTL-0000010627

© Crown owned Copyright (2024) Supplied to MOD under DEFCON 703 in accordance with Contract No [DSTL-0000010627, Task C67-02]

Version history

Version	Date	Author	Action
01-000	07/07/2023	Butcher, I	Initial Release
02-000	08/09/2023	Butcher, I	Updated with latest features and details implemented into Apex. Changes include: <ul style="list-style-type: none"> - Terminology changes from HLDDM to DMM and sensors to nodes, details on new features. - New Features: Database rollover, Protobuf support, Validation, version database, updated screenshots for GUI changes. - Updated steps in setup and usage.
02-001	14/09/2023	Butcher, I	Up-issue following PMO Review.
02-002	05/03/2024	Sharma, P	Updates for latest features including. <ul style="list-style-type: none"> - Installation steps with poetry - Delivery formats of Apex - Support for BSI Flex 335 v2.0 - Simultaneous handling of Nodes with different SAPIENT versions and changes to database structure - REST API & Supported Endpoints.
02-003	25/03/2024	Sharma, P	Up-issue following PMO Review.
02-004	10/05/2024	d'Avezac, M	Up-issue following DSTL comments.
02-005	03/06/2024	Sharma, P	Updated version format for icd_version parameters.

Authorisation

Role	Name	Signature	Date
Author:	Sharma, P	P.Sharma	03/04/2024
Approved by:	Islip, S	S. Islip	06/06/2024
Authorised by:	Woods Mestre, N	N.Woods Mestre	06/06/2024

1 Executive Summary

The SERAPIS Lot 1 C67-02 – Robust Middleware phase ST2.1 identified Roke’s Apex software as a suitable replacement for the existing SAPIENT middleware and created a backlog of tasks and requirements which will further enhance the software. During phases ST2.2 & ST2.3 the backlog task list will be implemented in priority order and provide the Authority with iterative releases of the enhanced Apex software.

The purpose of this User Manual is to serve as a comprehensive guide to help users effectively navigate and utilise the features and functionalities of the Apex middleware software.

Once this document has been formally issued, amendments to this document shall be controlled in the same manner as for other project specification documents.

This page is intentionally blank.

2 Contents

1	Executive Summary	1
2	Contents	3
3	Introduction	6
4	Overview	7
4.1	Message routing.	8
4.2	Message validation.	9
4.3	Message archiving.	9
5	Quick Start	10
6	Installing Apex	12
6.1	System Requirements	12
6.2	Apex Releases & Installation Types	12
6.3	Installation	13
6.4	Starting Apex	14
7	GUI Usage	17
7.1	Connections Tab	17
7.2	Message Types Tab	18
7.3	Messages Tab	18
8	Apex REST API	20
8.1	REST Endpoints	20
8.2	REST API Error Codes	22
9	Configuration	24
9.1	Configuration Options	24
9.2	Applying Configuration Changes	29
10	Database Structure	31
10.1	Connection Table	31
10.2	Messages Table	32
10.3	RolloverFilename Table	35

10.4	Version Table	35
11	Validation	36
12	Scripts	37
12.1	Replay	37
13	Known Limitations	39
14	User Advice	40
14.1	Virtual Environment	40
14.2	Configuration Backups	40
14.3	Run GUI with Apex	40
15	Licensing	41
16	Intellectual property	42
17	Support	43
18	Abbreviations & Definitions	44
18.1	List of Abbreviations	44
19	Distribution List	45
20	Report Documentation Page v1.0	46

List of Figures

Figure 1 - High level Apex Interactions	7
Figure 2 - Quick Start Empty Connections	10
Figure 3 - Quick Start Populated Connections	11
Figure 4 - GUI No Connections	16
Figure 5 - Connections Tab Collapsed	17
Figure 6 - Connections Tab Expanded	17
Figure 7 - Message Types Tab	18
Figure 8 - Messages Tab	19

List of Tables

Table 1 - Column Meanings	18
Table 2: REST Endpoints	22
Table 3: REST API Error Codes.....	23
Table 4 - Configuration Options.....	27
Table 5 - Connection Options	28
Table 6 - Elasticsearch Configuration	29
Table 7: REST Server Configuration	29

Table 8 - Connection Table 32

Table 9 - Messages Table 35

Table 10 - RolloverFileName Table..... 35

Table 11 - Version Table 35

Table 12 - Replay Script Config 38

3 Introduction

This user manual serves as a comprehensive guide to help users effectively navigate and utilise the features and functionalities of the Apex middleware software program. Provided within are step-by-step instructions, best practices, and detailed explanations to assist in maximising the benefits of Apex. The user manual is structured into distinct sections, each addressing specific aspects of the Apex functionality and is designed to provide the necessary guidance to utilise Apex effectively:

- **Overview:** High level details on what Apex is and its main functions.
- **Quick Start:** A fast and short guide for installing and running Apex.
- **Installing Apex:** Detailed guide outlining system requirements, dependencies, and installation steps.
- **GUI Usage:** Information and screenshots of the different GUI tabs.
- **Configuration:** Detailed information on the configuration.
- **Database Structure:** Overview of the information stored within the Database.
- **Validation:** Detailed information on the supported validation types.
- **Scripts:** Details around how to use scripts provided within the source code.
- **Known Limitations:** Any known limitations.
- **User Advice:** Useful tips and advice.

4 Overview

The Apex software is a SAPIENT middleware software that underpins the operation of SAPIENT. The middleware provides the layer enabling Edge nodes (a.k.a. Child nodes) and Fusion nodes (a.k.a. Peer nodes) to communicate. It interacts with the Fusion nodes, Edge nodes, and its own SQL database or elastic search instance as shown in Figure 1 - High level Apex Interactions. Optionally, it can also interact with a Parent node, within a hierarchical SAPIENT setup. It can also interact with an SQL or Elastic search database. This last feature allow users to examine the SAPIENT network though a GUI or a rest API.

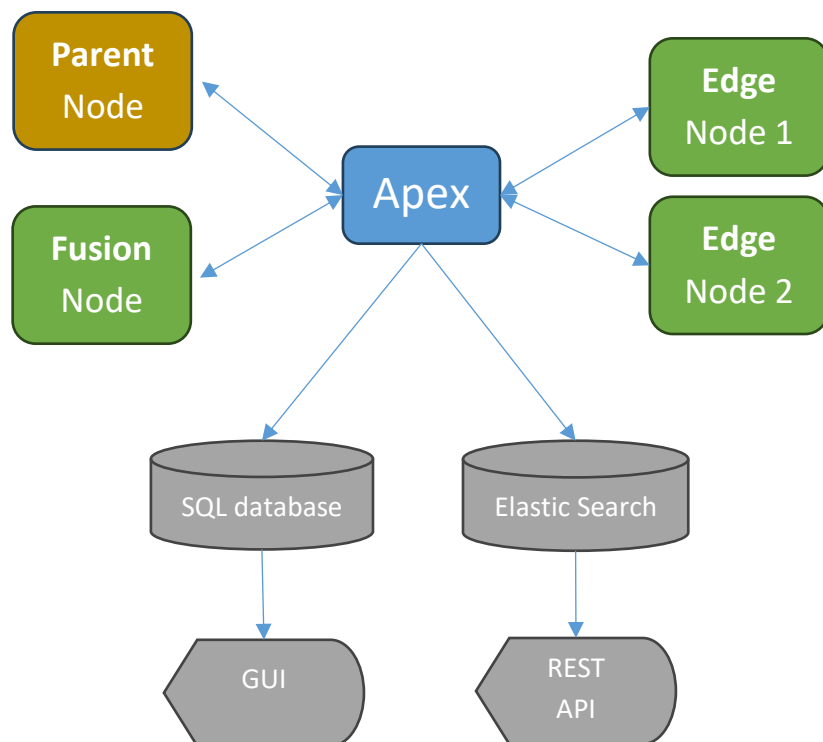


Figure 1 - High level Apex Interactions

It has three main functions:

1. Routing messages between the other components (Fusion nodes, Edge nodes).
2. Validating that the messages comply with the message ICD.
3. Archiving the messages.

Apex currently supports multiple SAPIENT Standards

- BSI Flex 335 v2.0
- BSI Flex 335 v1.0
- ICD Location: <https://www.bsigroup.com/en-GB/standards/bsi-flex-3352023/>
- Backwards Compatibility with SAPIENT Version 6

It is possible to have Nodes which are using different SAPIENT standards connect simultaneously to Apex, using the pre-defined Apex ports which specify the port format & icd_version.

4.1 Message routing.

The main job of Apex is to route messages between connections. The most notable rules are:

- There are ports for all Edge nodes to connect to
 - The default ports below specify the data format and icd_version used.
 - 5020: For BSI Flex 335 v2.0 & the Protobuf message format
 - 5010: For BSI Flex 335 v1.0 & the Protobuf message format
 - 5000: For SAPIENT Version 6 & the XML message format
 - The node ID of the connection is taken from the registration message sent to this connection.
 - Any message sent by the Edge nodes are forwarded to the connected Fusion node.
- There is a port for the Fusion node to connect to (5001 by default).
 - Any message with a non-zero node ID is sent to the relevant Edge node connection.
 - Any Alert or DetectionReport is sent to the Listener connection.
- There is a port for a “Parent” to connect to (5002 by default).
 - This allows a connection to receive Fusion-level track and alert data.
 - Provides option to forward all messages, not only high-level.
 - No incoming messages are expected on this connection.
- There is a port for a “Recorder” connection (5003 by default).
 - Incoming messages are not routed anywhere (but, as with all messages, they’re still archived - see below)

There are some additional minor rules layered on top of that, including:

- For each Edge node, some messages are cached and sent to the Fusion node when it (re)connects so that a Fusion node is aware of any previously connected nodes. Those are:
 - The registration message (specifically, the most recent one that was not an error)
 - The most recent “New” status report (if sent after the registration message)
 - The most recent “Unchanged” status report (if sent after the most recent “New” status report)
 - New / Unchanged status reports act as a flag to tell the Fusion if the information in this message is new or unchanged.
- The first message received from an Edge node should be a registration message. Any messages received before a registration message are considered errors and not forwarded to the Fusion node. Later registration messages with the same node ID are allowed, but later registration messages (or indeed any other messages) with a different node ID are an error.
- If multiple Edge node connections are open with the same node ID, then only the most recent one (the one that sent its first registration message most recently) is considered active, and only its messages are forwarded to the Fusion node. The first connection is not closed, to stop two sources from fighting over a node ID.
 - Conflicting node IDs can occur within incorrectly configured Edge nodes.

4.2 Message validation.

Several validation options are supported within Apex which validate that the messages comply with the configured message ICD. Apex validation requires an annotated version of Protobuf files, which are included within the Apex source code by default. These annotations to the Protobuf files, are Protobuf custom options which are used to identify fields that are mandatory or if they should be a ULID or UUID. See the validation section for more details on validation.

Messages that fail validation always result in the message being dropped (i.e. not forwarded to any other connections), and an error is generated. Errors in Apex are handled in one of three different ways:

- **FATAL errors:** These messages result in the connection being immediately disconnected. For example, if the maximum message size is exceeded.
- **NOISY errors:** These messages result in a standard ICD “Error” message being sent back to describe the source of the problem. This is the most common result, with examples being invalid Protobuf validation errors, or message-specific errors such as “Unchanged” status report being received before receiving any “New” reports.
- **SILENT errors:** These messages are not reported back to the originating connection. The only effect is that the message is not forwarded elsewhere. An example is receiving an “Error” Protobuf message (to avoid two middleware’s infinitely bouncing error messages between each other).

Failed validation errors are marked as noisy and recorded within the database. Details on the validation error(s) are stored in the database with the message to assist with investigations.

4.3 Message archiving.

Messages are saved to a local SQLite database for later review and playback. The database structure details can be found within the Database Structure page.

5 Quick Start

The following steps are a quick outline of how to go from obtaining the Apex code, to installing, and running.

1. Download and extract Apex source code.
2. Install poetry

```
pip install poetry
```

3. Create and activate new python virtual environment:

```
python -m venv  
venv\Scripts\activate
```

4. Using poetry, install dependencies

```
poetry install --all-extras
```

5. compile Protobuf files

```
poetry run python run_protoc.py
```

6. Start Apex:

```
apex
```

7. Start Apex GUI:

```
apex_gui
```

8. When the Apex GUI is first started, it will appear with no connections in the connections tab as show in Figure 2 - Quick Start Empty Connections

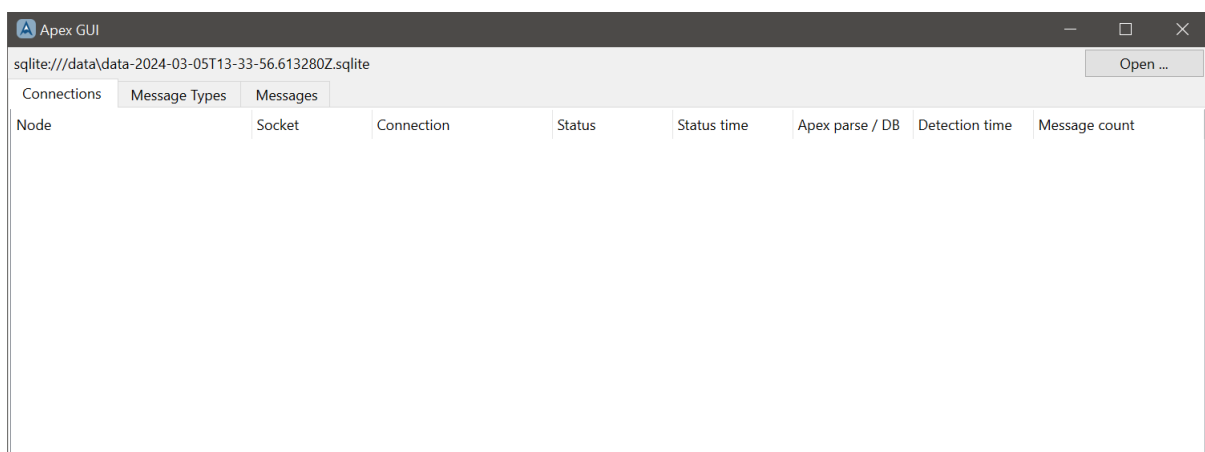
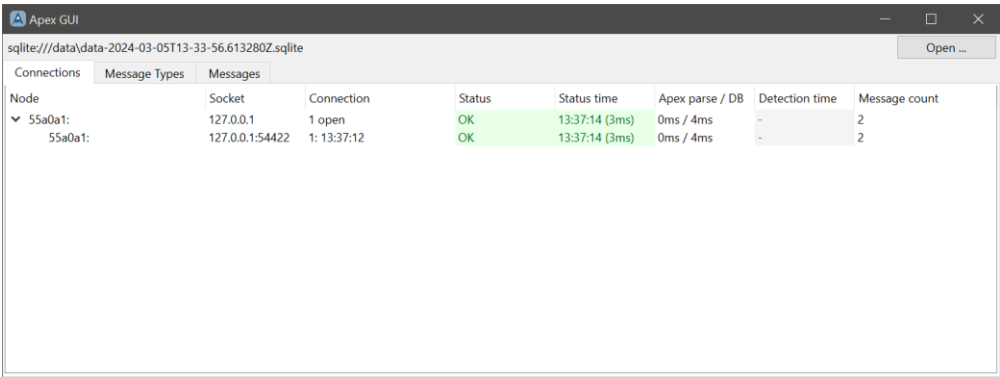


Figure 2 - Quick Start Empty Connections

9. Next, connect Edge Nodes on ports 5000 for XML, 5010 BSI Flex 335 V1.0 Protobuf, 5020 for BSI Flex 335 V2.0 Protobuf and Fusion nodes on port 5001/

10. Once an Edge or Fusion node is connected the GUI will appear like Figure 3 - Quick Start Populated Connections:



The screenshot shows the Apex GUI window with the title bar 'Apex GUI'. Below the title bar is a file path 'sqlite:///data/data-2024-03-05T13-33-56.613280Z.sqlite' and an 'Open ...' button. The main area has three tabs: 'Connections', 'Message Types', and 'Messages'. The 'Connections' tab is active, displaying a table with the following data:

Node	Socket	Connection	Status	Status time	Apex parse / DB	Detection time	Message count
55a0a1:	127.0.0.1	1 open	OK	13:37:14 (3ms)	0ms / 4ms	-	2
55a0a1:	127.0.0.1:54422	1: 13:37:12	OK	13:37:14 (3ms)	0ms / 4ms	-	2

Figure 3 - Quick Start Populated Connections

See Installing Apex section for full details on system requirements, installation, and starting Apex.

6 Installing Apex

6.1 System Requirements

Before proceeding with the installation of Apex, ensure that your system has met the following system requirements and prerequisites.

Minimum System Requirements

- **Memory:** 4GB
- **CPU/Cores:** 4 Cores
- **Storage:** 64GB (Preferably SSD)
- **Operating System:** Windows 10 / Ubuntu 18.04 or Newer

Prerequisites

- Python 3.9 or newer
- Access to the internet or an up-to-date pip package repository

6.2 Apex Releases & Installation Types

Apex releases are made available in multiple forms, depending on the use case.

6.2.1 Apex Binaries

- This is for end-users and in the form of compiled Apex executables for Windows OS.
- Apex is ready to use.
- Does not need an internet connection.

6.2.2 Apex Sources + Python Virtual Environment

- This is meant for technical users.
- Apex is provided as python source code and included is a pre-configured python virtual environment with all required packages already installed.
- Does not need an internet connection.
- The virtual environment should be activated.
- A matching & working python installation is required, matching the version of python provided in the virtual environment. The location of the installed python should also match. Python 3.10 should be installed such that it's found at
C:\Program Files\Python310\python.exe

6.2.3 Apex Sources

- Meant for technical users or developers.
- Contains the Apex (python) source code.
- Needs a working internet connection.
- Needs a working python, poetry installation.

6.3 Installation

6.3.1 Apex Binaries

The Apex binary is provided as a .zip file, following the naming convention of “apex-v<major>.<minor>.<patch>-**deploy**.zip”. Once downloaded, extract the contents into a separate directory on the system that Apex will run on.

1. To start Apex & Apex GUI, execute

```
run_all.bat
```

2. Alternatively, open bin sub-directory to launch Apex manually.

```
apex.exe
```

3. Apex GUI as

```
apex_gui.exe
```

6.3.2 Apex Sources + Python Virtual Environment

The Apex sources + python virtual environment is provided as a .zip file, following the naming convention of “apex-v<major>.<minor>.<patch>-**python**.zip”. Once downloaded, extract the contents into a separate directory on the system that Apex will run on.

1. Open a terminal and navigate into the Apex software directory.
2. Activate the included virtual environment:

```
venv\Scripts\activate
```

6.3.3 Apex Sources

The Apex source code is provided as a .zip file, following the naming convention of “apex-v<major>.<minor>.<patch>-**source**.zip”. Once downloaded, extract the Apex source code onto the system that Apex will run on.

3. Open a terminal and navigate into the Apex software directory.
4. For development, Poetry is recommended since it provides better dependency management. It can be installed with

```
pip install poetry
```

5. Create and activate a virtual environment:

```
python -m venv venv  
venv\Scripts\activate
```

6. Using poetry, install dependencies (including GUI and dev dependencies, as defined in pyproject.toml) and the apex packages in editable mode)

```
poetry install --all-extras
```

7. Create the python bindings from the SAPIENT Protobuf messages using

```
poetry run python run_protoc.py
```

6.4 Starting Apex

Now that Apex has now been successfully installed it can be started.

6.4.1 Apex Binaries

1. To start Apex & Apex GUI, execute

```
run_all.bat
```

2. Alternatively, open bin sub-directory to launch Apex manually.

```
apex.exe
```

3. Apex GUI as

```
apex_gui.exe
```

6.4.2 Apex sources

(After activating the provided or created python environment)

1. Start Apex:

```
apex
```

or

```
python sapient_apex_server\apex.py
```

When Apex starts it will print information to the console, including the read in config file and start up steps. An example output will look similar to the following:

```
$ python sapient_apex_server\apex.py
2023-08-31 10:04:06,510 INFO apex: Starting ...
2023-08-31 10:04:06,511 INFO apex: Running on Python 3.10.5
(tags/v3.10.5:f377153, Jun 6 2022, 16:14:13) [MSC v.1929 64 bit (AMD64)]
2023-08-31 10:04:06,511 INFO apex: Using config file: apex_config.json
2023-08-31 10:04:06,513 INFO apex: Using config:
{
  "logLevel": "INFO",
  "connections": [
    {
      "type": "Child",
      "port": 5000
    },
    {
```



```

    "type": "Peer",
    "port": 5001
  },
  {
    "type": "Parent",
    "port": 5002
  },
  {
    "type": "Recorder",
    "port": 5003
  },
  {
    "type": "Parent",
    "host": "127.0.0.1",
    "port": 10000,
    "outbound": true
  },
  {
    "type": "Parent",
    "host": "127.0.0.1",
    "port": 5004,
    "outbound": false,
    "forwardAll": true
  }
],
"enableTimeSyncAdjustment": false,
"messageMaxSizeKb": 1024,
"detectionConfidenceFiltering": {
  "enable": false,
  "threshold": 0.5,
  "storeInDatabase": true
},
"middlewareId": "5913c0f4-9f89-4c01-ab90-939099797c4f",
"allowPeerRegistration": true,
"rollover": {
  "enable": true,
  "unit": "days",
  "value": 1
},
"validationOptions": {
  "validationTypes": [
    "mandatory_fields_present",
    "mandatory_oneof_present",
    "mandatory_repeated_present",
    "no_unknown_fields",
    "no_unknown_enum_values",
    "id_format_valid",
    "message_timestamp_reasonable",
    "detection_timestamp_reasonable"
  ],
  "strictIdFormat": true,
  "messageTimestampRange": [
    -0.9,
    0.1
  ],
  "detectionMinimumGap": 0.08
}
}
}
2023-08-31 10:04:06,523 INFO apex: Opening SQLite database with filename:

```

```
data/data-2023-08-31T09-04-06.522716Z.sqlite
2023-08-31 10:04:06,546 INFO apex: Database opened and tables created
2023-08-31 10:04:06,551 INFO apex: Press Ctrl+C to exit
```

2. Start Apex GUI:

```
apex_gui
```

or

```
python sapient_apex_gui\apex_gui.py
```

When the Apex GUI starts there is a small number of prints to the console, it will look similar to the following:

```
$ python sapient_apex_gui\apex_gui.py
2023-08-31 10:07:15,122 INFO [db] apex_gui: Opening SQLite database with
filename: data\data-2023-08-31T09-04-06.522716Z.sqlite
```

The Apex GUI will look like Figure 4 - GUI No Connections, while there are no Edge or Fusion nodes connected:

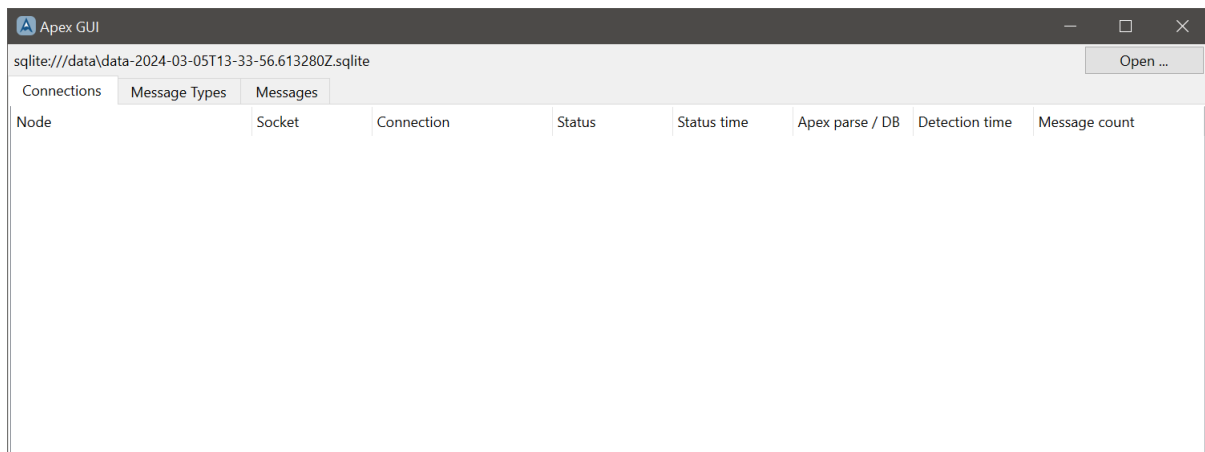


Figure 4 - GUI No Connections

7 GUI Usage

Once Apex is running and the Apex GUI is open, you will first be presented with the Connections tab view.

7.1 Connections Tab

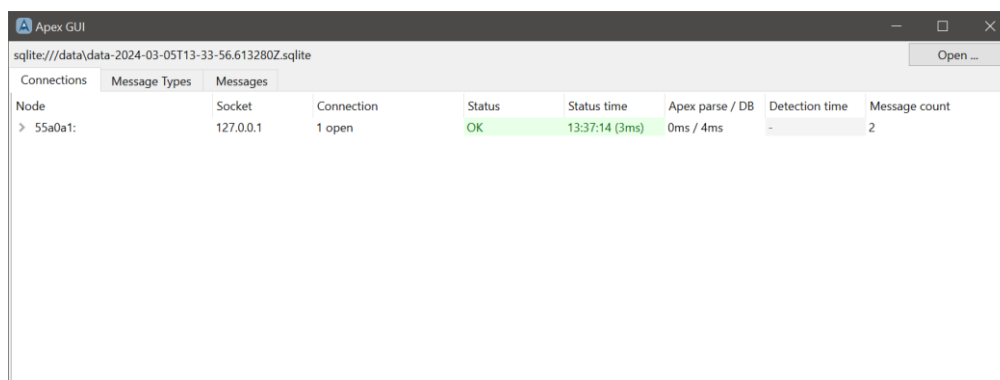
This tab is used to display all inactive and active connections being handled by Apex. It provides visibility into useful information for these connections, such as:

- Connection details (host:port)
- Status details (last message time and delays)
- Message count.

This tab serves as an overview of all the connections and is useful to leave active to provide a quick assessment of the Apex connect statuses.

The connections tab as shown in Figure 5 - Connections Tab Collapsed, has two-level tree of rows:

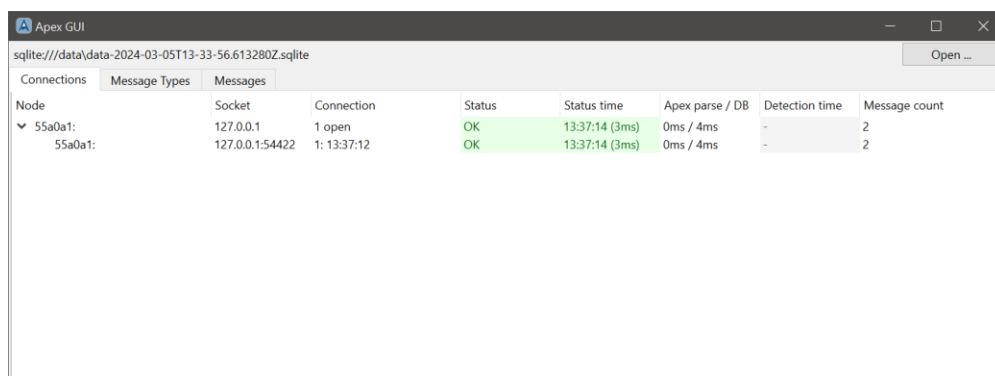
- Each top-level row represents the Fusion (Peer), Listener, or an Edge (Child) node ID. It includes information from the currently active connection (or most recent inactive connection).



Node	Socket	Connection	Status	Status time	Apex parse / DB	Detection time	Message count
> 55a0a1:	127.0.0.1	1 open	OK	13:37:14 (3ms)	0ms / 4ms	-	2

Figure 5 - Connections Tab Collapsed

- Expanding a top-level row reveals the individual connections for that connection type / node ID as shown in Figure 6 - Connections Tab Expanded.



Node	Socket	Connection	Status	Status time	Apex parse / DB	Detection time	Message count
▼ 55a0a1:	127.0.0.1	1 open	OK	13:37:14 (3ms)	0ms / 4ms	-	2
55a0a1:	127.0.0.1:54422	1: 13:37:12	OK	13:37:14 (3ms)	0ms / 4ms	-	2

Figure 6 - Connections Tab Expanded

Column meanings

Column Name	Description
Node	Identifies the connection type and (if an Edge) node ID and node type
Socket	Remote Socket with IP address
Connection	(Top level) number of connections; (individual connection) connection ID, start time, end time (tooltip has more info)
Status	Info from most recent status report
Status time	When most recent status report was received, and the timestamp delay when received (tooltip has more info)
Apex parse / DB	Internal delays in Apex, to show if system is becoming overloaded
Message count	(Top level) Total sum of messages received; (individual connection) number of messages

Table 1 - Column Meanings

7.2 Message Types Tab

This tab shows the number of messages and when most recently received, grouped by message type as show in Figure 7 - Message Types Tab. Type a connection ID and press enter to populate or refresh. This is useful to obtain message counts for a specific connection.

Type	Count	Last time	Error count	Error last time	Error last description
detection_r...	1	09:19:55 (1ms)	0	-	-
registration	1	09:19:41 (3ms)	0	-	-
status_report	2118	09:26:57 (1ms)	0	-	-

Figure 7 - Message Types Tab

7.3 Messages Tab

This tab shows a complete list of all messages as shown in Figure 8 - Messages Tab. Enter a connection ID, or leave blank to get messages for all connections, and press enter to populate / refresh. Enter a page number or use the buttons to change page. The number of pages (in the “Of:” box) will update

along with the results when changing connection ID or page. Click on a message to show its original JSON representation of the message below. There will also be two more tabs: Proto which displays the raw Protobuf message bytes received and errors which will display any error generated by the message.

The screenshot shows the Apex GUI with the 'Messages' tab selected. The interface includes a search bar for 'Connection ID' (set to 1) and pagination controls (Page: 105, Of: 113). A table lists messages with columns: ID, Connection, Node ID, Type, Forwarded, Timestamp, Received, Saved, Error Severity, and Error Mes. Message 2091 is selected. Below the table, the 'JSON' tab is active, displaying the message's JSON structure. To the right of the JSON view are 'Copy' and 'Highlight' buttons.

ID	Connection	Node ID	Type	Forwarded	Timestamp	Received	Saved	Error Severity	Error Mes
2081	1	7421e23d-075c...	status_report	0	2023-08-31 09:...	2023-08-31 09:...	2023-08-31 09:...	-	-
2082	1	7421e23d-075c...	status_report	0	2023-08-31 09:...	2023-08-31 09:...	2023-08-31 09:...	-	-
2083	1	7421e23d-075c...	status_report	0	2023-08-31 09:...	2023-08-31 09:...	2023-08-31 09:...	-	-
2084	1	7421e23d-075c...	status_report	0	2023-08-31 09:...	2023-08-31 09:...	2023-08-31 09:...	-	-
2085	1	7421e23d-075c...	status_report	0	2023-08-31 09:...	2023-08-31 09:...	2023-08-31 09:...	-	-
2086	1	7421e23d-075c...	status_report	0	2023-08-31 09:...	2023-08-31 09:...	2023-08-31 09:...	-	-
2087	1	7421e23d-075c...	status_report	0	2023-08-31 09:...	2023-08-31 09:...	2023-08-31 09:...	-	-
2088	1	7421e23d-075c...	status_report	0	2023-08-31 09:...	2023-08-31 09:...	2023-08-31 09:...	-	-
2089	1	7421e23d-075c...	status_report	0	2023-08-31 09:...	2023-08-31 09:...	2023-08-31 09:...	-	-
2090	1	7421e23d-075c...	status_report	0	2023-08-31 09:...	2023-08-31 09:...	2023-08-31 09:...	-	-
2091	1	7421e23d-075c...	status_report	0	2023-08-31 09:...	2023-08-31 09:...	2023-08-31 09:...	-	-
2092	1	7421e23d-075c...	status_report	0	2023-08-31 09:...	2023-08-31 09:...	2023-08-31 09:...	-	-
2093	1	7421e23d-075c...	status_report	0	2023-08-31 09:...	2023-08-31 09:...	2023-08-31 09:...	-	-

```

{
  "timestamp": "2023-08-31T09:26:51.874372Z",
  "node_id": "7421e23d-075c-4a50-b237-ef6d19474f5c",
  "status_report": {
    "report_id": "01H95ENEQ2VXVPG25MRSFRKXSX",
    "system": "SYSTEM_OK",
    "info": "INFO_NEW",
    "node_location": {
      "x": 0.1,
      "y": 0.1,
      "coordinate_system": "LOCATION_COORDINATE_SYSTEM_UTM_M",
      "datum": "LOCATION_DATUM_WGS84_G"
    },
    "field_of_view": {
      "range_bearing": {
        "azimuth": 0.0,
        "range": 300.0,
        "horizontal_extent": 360.0,

```

Figure 8 - Messages Tab

8 Apex REST API

When Apex is launched, it also starts a REST Service, which provides a REST API.

The REST API allows a REST client to use the provided REST Endpoints to retrieve data from the Apex server. This can include node information, locations, detections both from edge nodes and fusion nodes. The endpoints also support a variety of parameters which can be used to filter the results which are returned.

By default, the REST server will be served on port 8080 on the local host, so the default URL will be <http://127.0.0.1:8080/>

A Swagger based GUI is also served on <http://127.0.0.1:8080/docs>

The Swagger Interface allows the REST endpoints.

- To be executed and obtain results.
- Documents the parameters and format/schema of the data sent/received for each of the commands.
- Expected error codes and responses.
- Example code/URLs for REST API clients/apps to use.

8.1 REST Endpoints

REST Endpoint (GETs)	Description	Response	Comment
/	Default/Root endpoint	The response has a description of the REST Service and the Apex version in use.	
/registered	Get list of registered nodes	If nodes have registered before it should contain the UUIDs of these nodes otherwise the list is empty.	

/locations	Get location details of all nodes	If nodes have been registered before, the list should contain a list of nodes along with their locations as per the defined schema.	Default to all nodes. Specify specific node(s) using the node_id parameter.
/field_of_views	Get field of view details of all nodes	If nodes have been registered before, the list should contain a list of nodes along with their field of views as per the defined schema.	Default to all nodes. Specify specific node(s) using the node_id parameter.
/detections	Get detections from all nodes	If detections have been generated before, the list should contain a list of nodes along with detection reports as per the defined schema.	Defaults to all detections. Refer to the Swagger interface & inline documentation for all the other parameters this endpoint takes.
/detections/locations	Get detection locations from all nodes	If detections have been generated before, the list should contain a list of nodes along with detection locations as per the defined schema.	Defaults to all detections. Refer to the Swagger interface & inline documentation for all the other parameters this endpoint takes.

/detections/associated_files	Get detection's associated files with parameters specified.	<p>A successful response should be returned.</p> <p>If detections have been generated before, the list should contain a list of nodes along with detection's associated files as per the defined schema.</p>	<p>Defaults to all detections.</p> <p>Refer to the Swagger interface & inline documentation for all the other parameters this endpoint takes.</p>
/node_definitions	Get node definitions of all nodes	<p>A successful response should be returned.</p> <p>If nodes have been registered before, the list should contain a list of nodes along with their node definitions as per the defined schema.</p>	<p>Default to all nodes.</p> <p>Specify specific node(s) using the node_id parameter.</p>

Table 2: REST Endpoints

8.2 REST API Error Codes

Typical Error Codes reported by the Apex REST API are listed.

HTTP Error Code	Description/Detail	Comment/Resolution
400	No Registered Nodes found.	No Nodes have connected and registered to the Apex middle
400	node_ids <one or more> could not be found	The endpoint specified node_ids parameter, which could not be found. This could mean that the node_id specified is incorrect or this node_id has not connected/registered yet.

500	Internal Error	REST Server encountered an error or could not connect to the Elasticsearch database instance.
503	Database Service Unavailable or not started	Elasticsearch is not configured correctly or enabled or has not started yet.

Table 3: REST API Error Codes.

9 Configuration

The configuration file for Apex allows you to customise options to meet your specific deployment requirements. By modifying the configuration file, you can tailor Apex to better suit your needs. These sections outline the options available in the apex_config.json configuration file.

9.1 Configuration Options

Setting	Type	Description	Options
loglevel	String	The logging level options controls the amount of details captured in the log files.	INFO DEBUG
Connections	Array	An array listing all the connections the Apex software should expect to handle on start-up	See Connections
enableTimeSyncAdjustment	Boolean	Enables an adjustment to deal with differences in clock sync	true false
messageMaxSizeKb	Integer	Maximum allowed buffered message size in Kilobytes; helps detects callers forgetting to null terminate	Default: 1024
detectionConfidenceFiltering		Ignores detections below a	

		confidence threshold	
detectionConfidenceFiltering. enable	Boolean	Enable detection filtering	true false
detectionConfidenceFiltering. threshold	Integer	Detections with a confidence below this confidence threshold will be filtered out	Default: 0.5
detectionConfidenceFiltering. storeInDatabase	Boolean	Store error messages in the database when a detection fails to meet the threshold	true false
autoAssignNodeIDInRegistration	Boolean	Allows node ID to be omitted from registration, in which case Apex will auto assign it	true false
allowPeerRegistration	Boolean	Allows a registration message to be sent to Fusion nodes, and uses "node" ID from it	true false
rollover		Database Rollover Options	

rollover.enable	Boolean	Enabled / Disable database rollover	true false
rollover.unit	String	Configures the SQLite Database rollover units	weeks days minutes seconds
rollover.value	Integer	The rollover value applied in combination with the rollover units	
validationOptions		Message Validation Options	
validationOptions. validationTypes	Array	Array listing all the validation types to enable	mandatory_fields_present mandatory_oneof_present mandatory_repeated_present no_unknown_fields no_unknown_enum_values id_format_valid message_timestamp_reasonable detection_timestamp_reasonable
validationOptions.strictIdFormat	Boolean	If true (and id_format_valid is present) then UUID must be UUID v4, lower case, with correct dashes, and no braces or URN prefix; ULID must be upper case	true false

validationOptions. messageTimestampRange	Array	Minimum and maximum relative times for message timestamp validation	Default: [-0.9, 0.1]
validationOptions. detectionMinimumGap	Float	Minimum time between detection messages	Default: 0.08

Table 4 - Configuration Options

9.1.1 Connections

This table outlines the “connections” arrays objects:

Setting	Type	Description	Options
type	String	The type of connection.	Child Peer Parent Recorder
outbound	Boolean	Whether to connect outbound to this connection, or open socket to listen for this incoming connection	true false
port	Integer	Port to connect to or listen on	Any valid port
host	String	Free form string - Follow IPv4 IP address formatting	Free form string - Follow IPv4 IP address formatting
forward_all	Boolean	Forward all messages, not just high-level messages - only if type is parent	true: Forward all messages to this connection false: Only forward high-level messages to this connection (Default: false)
format	String	Defines the format of the data used by the port. Defaults to PROTO	PROTO XML

icd_version	String	The SAPIENT version used by this port. Defaults to the latest version supported by Apex.	BSI Flex 335 v2.0 BSI Flex 335 v1.0 VERSION6
-------------	--------	--	--

Table 5 - Connection Options

9.1.2 API Configuration

Apex offers a REST API as an external interface for a web client to retrieve results via REST endpoints. This interface requires an Elasticsearch backend to be installed and Apex configured to connect to this instance.

Following tables shows the parameters which need to be configured

9.1.2.1 Elasticsearch Configuration

Elasticsearch configuration is under the **elasticConfig** section in apex_config.json

Setting	Type	Description	Options
enabled	Boolean	This flag if set to true, connects to the configured elastic service & allows the REST endpoints to return results.	false true
host	String	Specifies the host name or IP address where the elasticsearch instance is running	Free form string - Follow IPv4 IP address formatting
port	Integer	Port on which to connect to the elasticsearch instance.	Elastic instance port, typically 9200
useSsl	Boolean	Enable SSL connection to elastic	true false
certLocation	String	Path to the certificate file containing the credentials for connecting to the elastic instance.	Path to http_ca.crt
user	String	The user name to login to elasticsearch instance	Free form string

password	String	The password for the user to login to elasticsearch instance	Free form string
----------	--------	--	------------------

Table 6 - Elasticsearch Configuration

9.1.2.2 REST Server Configuration

The REST server configuration is under the apiConfig section in apex_config.json

Setting	Type	Description	Options
host	String	Specifies the host name or IP address where the Apex REST server is started	Free form string - Follow IPv4 IP address formatting
port	Integer	Port on which the Apex REST Server allows clients to connect	Typically 8080

Table 7: REST Server Configuration

In addition, the Apex source code is provided with a README file which contains a similar breakdown of the configuration options with descriptions.

9.2 Applying Configuration Changes

To modify and apply changes to the configuration file, follow these steps:

1. Stop any instances of Apex currently running.
2. Navigate into the Apex software directory.
3. Locate the configuration file named: apex_config.json.
4. Open the file with your chosen editor.
5. Make changes and save the file.
6. Start Apex for the changes to take effect.

Warning:

It is important to exercise caution while modifying the configuration file, as incorrect changes may lead to unexpected behavior or even for Apex to not start. If you are uncertain about any specific options or its impact, please consult the software documentation or refer to the support section.

Remember to keep a record of any modifications made to the configuration file for future reference, troubleshooting, or needing to roll back changes.

In the event that the apex_config.json file is lost or preventing Apex from starting, a default version can be found within the Apex source code zip.

10 Database Structure

The following sections detail the structure of each table within the Apex SQLite database.

10.1 Connection Table

Field name	Type	Example value	Description
id	INTEGER	11	Primary key, just an incrementing value
client_type	TEXT	Edge	Type of connection (Edge, Fusion, Listener)
peer	TEXT	192.168.131.111:47386	Hostname and port
connect_time	INTEGER (timestamp)	1622035979573555 2021-05-26T13:32:59.573555Z	Time connection opened (stored as microseconds since epoch)
disconnect_time	INTEGER (timestamp)	1622038971878808 2021-05-26T14:22:51.878808Z	Time connection closed
disconnect_reason	TEXT	ConnectionResetError: [WinError 10054] An existing connection was forcibly closed by the remote host	Error message when connection closed
recent_msg_id_registration	INTEGER	499355	Message ID (i.e. id field from Message table) of most recent (valid) registration message for this connection
recent_msg_id_status_new	INTEGER	740965	Message ID of most recent "New" status report for this connection
recent_msg_id_status_unchanged	INTEGER	741004	Message ID of most recent "Unchanged"

			status report for this connection (if newer than most recent “New” report)
--	--	--	--

Table 8 - Connection Table

10.2 Messages Table

Field name	Type	Example value	Description
id	INTEGER	500930	Primary key, just an incrementing value
connection_id	INTEGER	11	Which connection this message was received from; foreign key into Connection table
timestamp_received	INTEGER (timestamp)	1622035979584791 2021-05-26T13:32:59.584791Z	When the message was received on the socket
timestamp_decoded	INTEGER	1622035979585520 2021-05-26T13:32:59.585520Z	When the message was parsed (measures internal delay in Apex due to parse thread backlog)
timestamp_saved	INTEGER	1622035979586531 2021-05-26T13:32:59.586531Z	When the message was saved to this database (measures internal delay

			in Apex due to SQLite thread backlog)
sapient_version	TEXT	BSI_FLEX_335_V2_0	The version of the SAPIENT standard used by this message
xml	TEXT	?xml version="1.0"?> <SensorRegistration> <timestamp>2024-03-05T15:29:46.3856735Z</timestamp> >...	The full original message in XML format if the node originally was a SAPIENT version6 node or the XML representation of the proto message converted to SAPIENT Version6
json	TEXT	{"timestamp": "2023-07-20...", "node_id": "01H0N6FP..."}	Parsed proto message into JSON format
proto	TEXT	0a 0c 08 a5 a4 e5 a5 06 10 d0 ee 9b e9 02 12 1a 30 31	Full original message bytes. Maybe empty if the Node is communicating using XML/SAPIENT version6
forwarded_count	INTEGER	1	How many other connections

			the message was forwarded to (not including re-forwarding to later Fusion node connections)
parsed_type	TEXT	Registration / StatusReport	Type of message
parsed_node_id	INTEGER	1	Node ID, taken from the SAPIENT message
parsed_timestamp	INTEGER	1622035979484791 2021-05-26T13:32:59.484791Z	Timestamp, extracted from the message ("timestamp" field)
registration_node_type	TEXT	Axis PTZ	If type is registration, "node_type" element from the message
status_report_system	TEXT		If type is status report, "system" element from the message (OK, Warning, Error, Tamper, GoodBye)
status_report_is_unchanged	BOOLEAN		If type is status report, whether "info" element is "New" or "Unchanged"

			(anything else is an error)
error_severity	TEXT		NULL if no error, otherwise "SILENT", "NOISY" or "FATAL" (see overview page for more details)
error_description	TEXT		Description of the error, if any (usually exception type and message)

Table 9 - Messages Table

10.3 RolloverFilename Table

Field name	Type	Example value	Description
relative_filepath	TEXT	data/data-2023-06-15T15-00-00.161185Z.sqlite	Relative filepath to new SQLite database
absolute_filepath	TEXT	C:/apex/data/data-2023-06-15T15-00-00.161185Z.sqlite	Absolute filepath to new SQLite database

Table 10 - RolloverFileName Table

10.4 Version Table

Field name	Type	Example value	Description
variant	TEXT	Apex	Variant name of the recorded version
version	TEXT	1	The SQLite database version

Table 11 - Version Table

11 Validation

Apex supports validation of messages it received against the BSI Flex 335 ICD. The supported validation types include:

- All mandatory fields are present in the message.
- All mandatory “oneof” fields are present in the message.
- All mandatory “repeated” fields are present in the message.
- There are no unknown fields within the message.
- There are no unknown enums within the message.
- UUID and ULID fields are in their correct respective formats.
- The message timestamp is within a reasonable time from current time.
- The detection timestamp is within a reasonable time from previous detection.

If a message fails validation, it is not forwarded on to its intended destination and instead an error message is sent back to the node the message originated from. In addition, the message will be recorded within the Apex database along with the error message. Some validation options, such as checking for reasonable timestamps, do not require any additional files to perform validation. However, several validation types require the annotated version of the BSI Flex 335 Protobuf files which are supplied with Apex.

These annotated Protobuf files have custom options which detail options such as: if a field is mandatory, or if they should be a ULID or UUID. These options act as flags that the validator uses to check the fields within messages. Validation can only be performed against the BSI Flex 335 ICD protos which contain these custom validation fields, which are supplied with the latest version of Apex.

These annotated Protobuf files can be found within the `sapient_msg` directory within the Apex source code. Details on Protobuf custom options can be found within the official Protobuf documentation located here: <https://protobuf.dev/programming-guides/proto3/>

Details on the configuration of validation within the config file can be found within the Configuration section.

12 Scripts

Within the Apex source code there is a directory named scripts which contains useful scripts that can be used alongside Apex. This section provides details around these scripts and how they're used.

12.1 Replay

The replay script ("replay.py") takes an Apex database, waits for an incoming TCP connection, then sends messages from that database to that connection. It takes various configuration options including what time to start the replay from as well as playback speed. It starts by sending registration and status messages from before that time so the client can understand what nodes were already up and running.

Configuration of the replay script is performed via its config file alongside it named: replay_config.json. The following table describes the configuration options available.

Setting	Type	Description	Options
log_level	String	The logging level options controls the amount of details captured in the log files.	INFO DEBUG
filename	String	Filepath to the Apex SQLite database to replay	
is_outbound	Boolean	Whether to connect outbound to this connection, or open socket to listen for this incoming connection	true false
host	String	Free form string - Follow IPv4 IP address formatting	
port	Integer	Port to connect to or listen on	Default: 54004
topic	String	Placeholder	
wait_every_n_messages	Integer	Placeholder	
send_pending_on_exit	Integer	Placeholder	
start_time	String	Start date and time to begin reading messages from the database	UTC Timestamp example: 2021-09-27T10:45:200285Z

end_time	String	End date and time to stop reading messages from the database	UTC Timestamp example: 2021-09-27T10:46:04.400288Z
speed_multiplier	Integer	Increases the replay speed of messages	Default: 2
format	String	Defines the format of the data used by the port. Defaults to PROTO	PROTO XML
icd_version	String	The SAPIENT version used by this port. Defaults to the latest version supported by Apex.	BSI Flex 335 v2.0 BSI Flex 335 v1.0 VERSION6

Table 12 - Replay Script Config

To run the script:

1. Update the replay_config.json file with the configuration required.
2. Run the replay.py script:

```
python sapient_apex_replay\replay.py
```

3. Connect your remote device to the port specified in the config file.
4. Messages should begin to be replayed.

13 Known Limitations

Below is a list of any Known limitations within Apex:

- **Only supports validation against BSI Flex ICDs:** Validation in Apex requires an annotated version of the Protobuf files to be present. Currently the only annotated version available is the BSI Flex 335 Protobuf files supplied with Apex. The SAPIENT message validation is performed at the corresponding version of the SAPIENT standard as defined by the configuration of the apex port on which the message arrived.
- **Validation against XML/SAPIENT Version:** Validation of XML messages (SAPIENT Version6) is limited in scope and messages are not strictly validated against the SAPIENT version 6 ICD. Errors are only logged, but the messages will still be attempted to be routed by Apex.

14 User Advice

Below are tips and advice for when using Apex.

14.1 Virtual Environment

It's advised to use a virtual environment for development and deployment of Apex to ensure you're developing in an isolated and self-contained environment. This will prevent conflicts with any global python installations and other projects.

14.2 Configuration Backups

It's advised to always make backups of the configuration file before editing. This is to ensure there is always a rollback option in the event the new configuration does not work.

14.3 Run GUI with Apex

Apex will function as usual without the Apex GUI open, however it's recommended to always have the Apex GUI running alongside it, or at the very least, ensure the system running Apex is capable of running the GUI if required. This is because having access to the GUI provides access to messages and connections, improving users debugging and investigation capabilities in the event of any issues.

15 Licensing

The Apex middleware and GUI software require a number of third-party dependencies, obtained through the Python ‘pip’ packaging system. Apex is not provided in a packaged format, however scripts to build and package it with its dependencies is included in the source code. An acknowledgements file is provided alongside the build scripts as guidance on what licensing acknowledgements should be included within any built version. It is the responsibility of the party building and distributing the packaged version of Apex to review the acknowledgements and ensure all licensing requirements are being met.

To ensure there are no invasive licensing requirements, Apex will aim to use dependencies with permissive licenses such as BSD, MIT, or Apache. Dependencies licensed under the GPL will specifically not be used, due to the invasive nature of the license. A limited number of dependencies licensed under the LGPL are currently used and may be used going forward; Each of these will be considered carefully and used in a manner which does not extend the LGPL license to the full Apex codebase.

Currently, the only LGPL dependency used is the Qt framework, used as a dependency of the Apex GUI. This dependency is implemented ensuring that the Qt framework is included as a replaceable shared library component, rather than being linked statically. In this manner the Apex software may be packaged into a binary distribution which includes the Qt libraries, without causing the LGPL license to apply to the full Apex codebase.

16 Intellectual property

Refer to “Serapis C67 ST2.3 Commercial Letter” (28348) for full details surrounding the IP of the deliverable D2.3b (Roke’s Apex Software).

DEFCON 703 applies to deliverables O1, O2, O3, O4, D2.3a, D2.3c and D2.2d as described in the Deliverables section of the Technical Proposal.

Roke will retain ownership of its background IP and nothing in this proposal shall restrict Roke’s use of its background IP. Roke has already delivered its Apex Software in Deliverable D2.2b (SAPIENT Middleware – Source Code) of ST2.2, and as part of this proposal for ST2.3 Roke is proposing to deliver Deliverable D2.3b which is a further drop of the Apex Software: Roke’s background IP. A training session on the Apex software is also proposed in Deliverable D2.3e.

The following terms (which applied to Deliverable D2.2b in ST2.2) shall apply to Deliverable D2.3b (Roke’s Apex software) and D2.3e (Apex software training session). For the avoidance of doubt, a licence for the use of Roke’s Apex software will be granted as part of ST2.3:

- Roke grants Dstl a non-exclusive, royalty-free, perpetual worldwide, irrevocable, unrestricted license to use, develop, modify, enhance and use for any other purpose Roke’s Apex software (including source code and documentation). This license shall include an unrestricted right to sub-license. For the avoidance of doubt, this license will allow Dstl to open-source the Apex software as-is or in modified form under any open-source licensing terms.
- THE SOFTWARE IS PROVIDED "AS IS", WITHOUT CONDITIONS, WARRANTIES OR OTHER TERMS OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE CONDITIONS, WARRANTIES OR OTHER TERMS AS TO SATISFACTORY QUALITY, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, THE USE OF REASONABLE SKILL AND CARE OR NONINFRINGEMENT.
- IN NO EVENT SHALL ROKE BE LIABLE FOR ANY CLAIM, LOSSES OR DAMAGES OR OTHER LIABILITY, WHETHER SUFFERED DIRECTLY OR INDIRECTLY OR ARE IMMEDIATE OR CONSEQUENTIAL, AND WHETHER ARISING IN CONTRACT, TORT (INCLUDING NEGLIGENCE) OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH DSTL’S EXERCISE OF THE RIGHTS GRANTED IN RESPECT OF THE SOFTWARE.

17 Support

As the original authors of the Apex software, Roke is well-placed to offer support and consultancy for Apex, and related technologies. If you encounter any issues or have any queries, please first contact your Roke account manager. They will liaise internally to identify the most appropriate team to address the issue and provide support.

Please ensure a comprehensive description of the issue or inquiry is provided. For issues, if possible, please provide:

- Any errors messages seen.
- Console outputs.
- A copy of the Apex SQLite database (If sensitive information is not contained).
- Steps for issue replication.
- Description of test setup and configuration.
- Contact information.

18 Abbreviations & Definitions

18.1 List of Abbreviations

API	Application Programming Interface
BSI	British Standards Institution
Edge node	Node equipped with sensors, a.k.a. Autonomous Sensor Module
Fusion node	Node equipped with data fusion and tasking capabilities, a.k.a. Decision Making Module
GUI	Graphical User Interface
ICD	Interface Control Document
PMO	Programme/Project Management Office
REST	Representational State Transfer
SAPIENT	Sensing for Asset Protection with Integrated Electronic Networked Technology
ULID	Universally unique Lexicographically sortable Identifier
UUID	Universally unique identifier
XML	Extensible Markup Language

19 Distribution List

Copy	Position	Organisation	Name/Contact
1.	PM	PMO	Luke Tyzack
2.	PM	Dstl	Arthur Stratford
3.	Technical Lead	Dstl	Stuart Colley
4.	Technical Lead	Dstl	Alyn Laughlin
5.	PSO	Dstl	Rosina Henderson
6.	PM	Roke	Nam-Taan Woods Mestre
7.	Technical Oversight	Roke	Simon Islip
8.	Software Engineer	Roke	Prashant Sharma
9.	Commercial	Roke	Jenny Bolton
-	Serapis Lot 1 PMO (for Official reports)		SerapisLot1PMO@roke.co.uk

20 Report Documentation Page v1.0

* Denotes a mandatory field

1a. Report number: *	C67-02 D2.3d	1b. Version number:	02.005
2 Date of publication: *	03/06/2024	3. Number of pages:	47
4a. Report UK protective marking: *	UK OFFICIAL		
4b. Report national caveats: *	None		
4c. Report descriptor: *	None		
5a. Title: *	D2.3d – User Manual		
5b. Title UK protective marking: *	UK OFFICIAL		
5c. Title national caveats: *	None		
5d. Title descriptor: *	None		
6. Authors: *	Sharma, P., d’Avezac, M.		
7a. Abstract: *	A user manual for operating the Apex middleware software		
7b. Abstract UK protective marking: *	UK OFFICIAL		
7c. Abstract national caveats: *	None		
7d. Abstract descriptor: *	None		
8. Keywords:	user manual, D2.3d, Apex, Roke		

Please note: Unclassified, Restricted and Confidential markings can only be used where the report is prepared on behalf of an international defence organisation and the appropriate prefix (e.g. NATO) included in the marking.

* Denotes a mandatory field

9. Name and address of publisher: * Roke Manor Research Limited Old Salisbury Lane Romsey SO51 0ZN		10. Name and address of funding source: Dstl Porton Down Salisbury SP4 0JQ	
11. Funding source contract:		DSTL-0000010627	
12. Dstl project number:		C67-02	
13. Programme:		Serapis	
14. Other report numbers:		72-23-N-350-U	
15a. Contract start date:		15b. Contract end date:	08/04/2024
16. IP conditions for report: *		DEFCON 703	
17a. Patents:		No	
17b. Application number:		N/A	
18. Release authority role:		Roke PM	

Guidance on completing the report documentation page can be found on the [Gov.UK website](https://www.gov.uk).



20150701_Guidance
_Document_Vers_5_

OFFICIAL

This page is intentionally blank

OFFICIAL

OFFICIAL



Serapis Framework



© Crown Owned Copyright 2024.

THIS DOCUMENT IS THE PROPERTY OF HIS BRITANNIC MAJESTY'S GOVERNMENT, and is issued for the information of those who need to know its contents in the course of their official duties. Outside Government service, this document is issued on a personal basis; each recipient is personally responsible for its safe custody and for ensuring that its contents are disclosed only to authorised persons. Anyone finding this document should hand it to a British Forces unit or to a police station for safe return to the Chief Security Officer, DEFENCE SCIENCE AND TECHNOLOGY LABORATORY, Porton Down, Wiltshire SP4 0JQ, with particulars of how and where found. THE UNAUTHORISED RETENTION OR DESTRUCTION OF THE DOCUMENT IS AN OFFENCE UNDER THE OFFICIAL SECRETS ACTS OF 1911 - 1989.

OFFICIAL