HLA Installation Guide

Revision C
Version 3
June 2025
Document DOC-TEL-HLA-IG-C-3

Advanced Simulation Technology inc.
500A Huntmar Park Drive • Herndon, Virginia 20170 USA
(703) 471-2104 • asti-usa.com

Product Name: Telestra

HLA Installation Guide

ASTi
500A Huntmar Park Drive
Herndon, Virginia 20170 USA

Red Hat Enterprise Linux (RHEL) Subscriptions

ASTi is an official Red Hat Embedded Partner. ASTi-provided products based on RHEL include Red Hat software integrated with ASTi's installation. ASTi includes a Red Hat subscription with every purchase of our Software and Information Assurance (SW/IA) maintenance products. Systems with active maintenance receive Red Hat software updates and support directly from ASTi.

Export Restriction

Countries other than the United States may restrict the import, use, or export of software that contains encryption technology. By installing this software, you agree that you shall be solely responsible for compliance with any such import, use, or export restrictions. For full details on Red Hat export restrictions, go to the following:

www.redhat.com/en/about/export-control-product-matrix

Revision history

Date	Revision	Version	Comments
2/6/2020	A	0	Combined HLA setup instructions from *ASTi HLA RTI Compatibility*, *Studio Technical User Guide*, and *Telestra Server Operations & Maintenance Manual* into one document.
6/24/2020	B	0	Added description of the three-way TCP handshake process to "ASTi HLA Host Control " and edited document for grammar and clarity.
1/11/2021	C	0	Updated the guide with Pitch RTI information. Made minor edits to grammar and style.
9/27/2022	C	1	Removed "license" references from the Red Hat Enterprise Linux export statement in the front matter.
3/8/2023	C	2	Updated the Red Hat Enterprise Linux subscription statement in the front matter.
6/20/2025	C	3	Fixed a broken link in "Add an HLA domain."

Contents

1.0 Introduction **1**

2.0 Choose a compatible HLA RTI file **2**

2.1 Supported HLA RTIs for Telestra 7.X 2

3.0 Install and activate an RTI file in the Telestra web interface **4**

3.1 Install an RTI file 4

3.2 Activate an RTI file 7

3.3 Set the HLA standard 8

4.0 Set up the HLA federation in Studio **9**

4.1 Add an HLA domain 10

4.2 Add an HLA configuration file 12

4.3 Add a DIS gateway 14

4.4 Connect the layout to the domain and gateway 15

4.5 Set up HLA radios 17

5.0 Join the federation at HLA run time **20**

5.1 ASTi HLA Host Control 20

5.1.1 Host emulator 21

5.2 ASTi HLA Host Control commands and responses 22

5.2.1 Status 22

5.2.2 Name 23

5.2.3 Activity 24

5.2.4 Join 25

5.2.5 Resign 26

5.2.6 Openlog 27

5.2.7 Closelog 29

5.2.8 Bye 29

5.2.9 Quit 29

5.2.10 Help 30

5.3 RTI Executive 30

5.3.1 (Optional) MAK RTI Executive 30

5.3.2 Pitch Central RTI Component (CRC) 31

Appendix A: External FED/FDD, RID, and mapping files 32

1.0 Introduction

Unlike many other solutions, ASTi's high-level architecture (HLA) implementation takes full advantage of the flexibility and interoperability that the DoD for Modeling & Simulation (DMSO) envisioned for HLA 1.3. ASTi has since expanded Telestra to support HLA Evolved (HLAe). The Telestra server's industrial, off-the-shelf technology also provides increased HLA performance and reliability.

HLA software is highly flexible, with capabilities that support HLA operation for radio and communication models:

- Support for multiple run-time infrastructure (RTI) vendors
- An established and published ASTi radio simulation object model (SOM) and Real-time Platform Reference (RPR) Federation Object Model (FOM) support
- Agile FOM capabilities
- Back-channel communication option
- Debug monitoring and log capabilities
- Dynamic run-time configuration changes (e.g., change HLA-associated files without reinstalling the project)

Unfortunately, this flexibility comes with added complexity. HLA is both a standard and an architecture. Unlike DIS, where Transmitter and Signal PDU formatting is fixed, HLA can define each object's structure and interaction on the network. There are many ways to set up audio-related objects and interactions. HLA is riddled with complex parts, such as the RTI version, RTI Initialization Data (RID) file, mapping file, and federation file. These pieces all affect the final, on-wire data structure. As a result, HLA implementation seldom allows for a plug-and-play system.

The following chapters discuss how to:

- Choose a compatible HLA RTI file
- Install an RTI file
- Set up the HLA federation in Studio
- Set the default project and layout
- Join the federation at HLA run time

To download a sample HLA project, go to “Telestra App Note #126: Sample HLA Project for Telestra 7.3.0 and Later” at support.asti-usa.com/appnotes/126.html.

2.0 Choose a compatible HLA RTI file

Run-time infrastructure (RTI) is a middleware required for HLA implementation. This software coordinates federate operations and data exchange during a run-time execution. Each RTI has an associated RTI Initialization Data (.rid) or equivalent (e.g., .mtl, .settings) file that configures RTI-specific initialization parameters.

ASTi is involved with an ever-growing number of high-level architecture (HLA)-based communication simulations for a variety of programs throughout the U.S. and internationally. From experience, each HLA program has its own set of unique problems dependent on the equipment, software, and simulation deployed. Therefore, ASTi cannot guarantee there is zero risk of any issues arising with the currently supported set of run-time infrastructures (RTIs). No set of tests can verify all aspects of operation.

ASTi must test a complex set of variables together in their target environment under operational conditions:

- RTI operation, vendor, and software version
- RTI Initialization Data (.rid), MAK (.mtl), or .settings file configurations
- Network operation and conditions
- Simulation software
- HLA use in a network environment

2.1 Supported HLA RTIs for Telestra 7.*X*

Choose the proper RTI based on GCC compatibility and the requirements of your operating system(s) or other HLA-related software. If you are interested in an unlisted RTI, contact ASTi to discuss the possibility of expanding our HLA software. Minor version changes from the RTI vendor do not typically impact our software's ability to run the RTI major version.

Requirements: RTIs must be 64-bit, GCC 4.8, and HLA 1.3 or HLA Evolved (HLAe) compatible.

The current HLA functionality is based on HLA 1.3 and HLAe testing. The Telestra 7.*X* software series supports the following HLA RTIs:

Vendor	Version	File Name	Telestra Software
HLA 1.3 RTIs			
MAK	4.5c	makRti4.5c-linux64-rhe7.tar.gz	7.3.0 or later
RTI-s	D35G, GCC 4.8	rtis_D35G_x86_64_g++-4.8.tar.gz	7.3.0 or later
HLAe RTIs			
MAK	4.5c	makRti4.5c-linux64-rhe7.tar.gz	7.3.0 or later
RTI-s	D36C, GCC 4.8	rtis_D36C_x86_64_g++-4.8.tar.gz	7.3.0 or later
Pitch	5.5.0.0	prti1516e_5_5_0_0_linux64_b189.rpm ***Note:*** *ASTi tested GCC 4.1 because GCC 4.8 was unavailable; however; these versions are compatible.*	7.8.0 or later

Table 1: Telestra 7.X HLA RTI compatibility

3.0 Install and activate an RTI file in the Telestra web interface

This section discusses how to:

- Install an HLA RTI file
- Activate an RTI file

3.1 Install an RTI file

***Important**: Telestra web interface access is disabled on hardened Telestra systems. Therefore, you must manually upload and install the RTI from the Linux command line. Alternatively, install the RTI from the Telestra web interface before hardening your system.*

To install the RTI file in the Telestra web interface, follow these steps:

1. If you have not done so already, insert a USB License Key with an active HLA license file into the Telestra server. To learn more about licensing, go to "LicensingLicenses" in the *Telestra Web Interface User Guide*.
2. On the left, go to **Simulation** > **Protocols**.

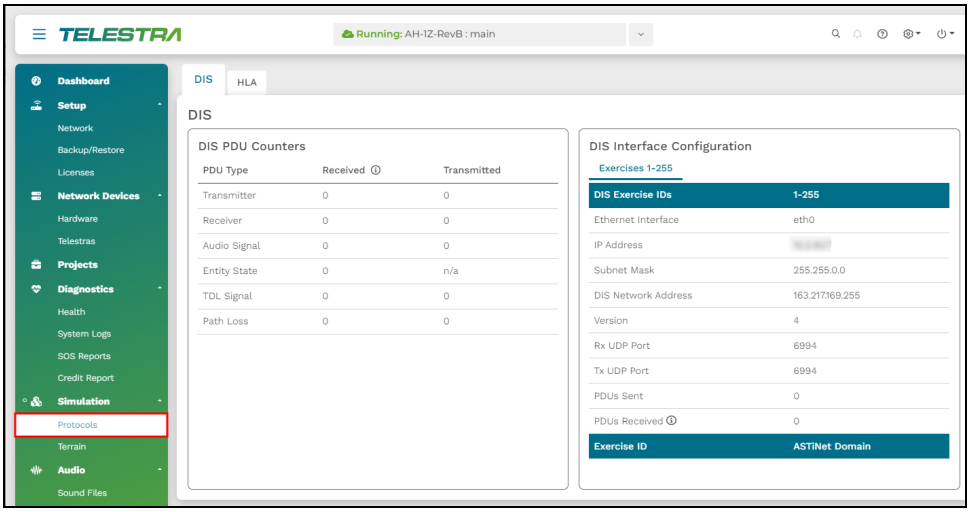

Figure 1: Protocols navigation

3. Go to the **HLA** tab.
4. Log in with the following default credentials:

Username	Password
admin	astirules

5. *(Optional)* To view the hidden password, select **Show Password** (👁).

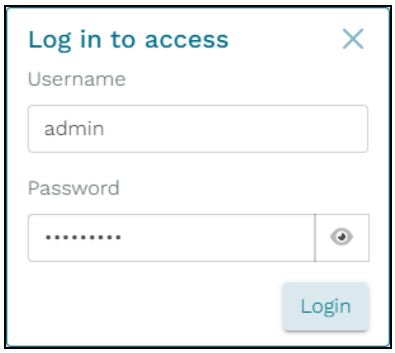

Figure 2: Login pop-up window

6. Select Login.

7. On **HLA Management**, select Upload an RTI.

8. In **Upload RTIs**, select Browse Files, and find up to five RTI files on your local system. Accepted file types include tape archive GNU ZIP (.tar.gz, .tgz, .gz), shell script (.sh), and Red Hat Packet Manager (.rpm).

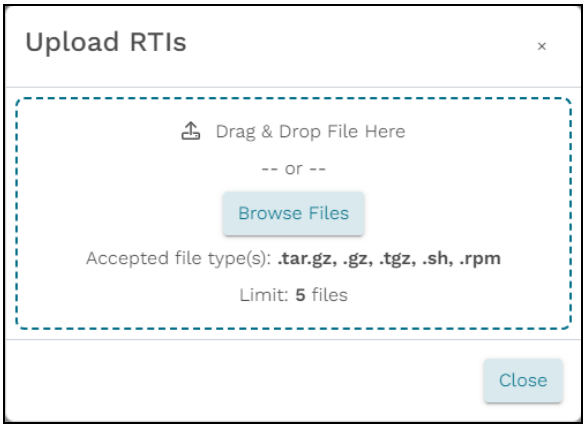

Figure 3: Browse RTI files

9. Review the RTI files to upload. To clear a file, select the trash can icon (🗑).

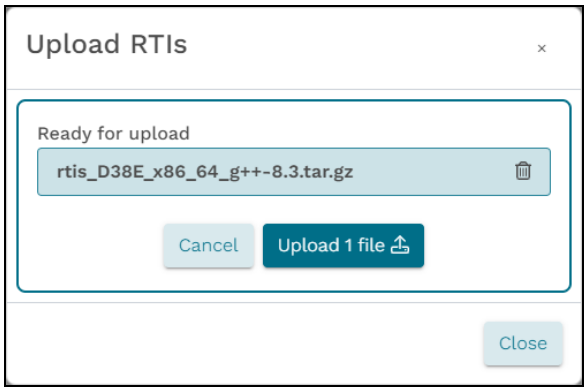

Figure 4: Upload RTI file(s)

Select Upload 1 file. The number of files on the button changes depending on your selection.

10. When the files are finished uploading, an "Upload is complete!" message displays, and a check (✓) appears next to each green status bar. Close the pop-up window, and confirm the new RTI files appear in the **RTIs** table:

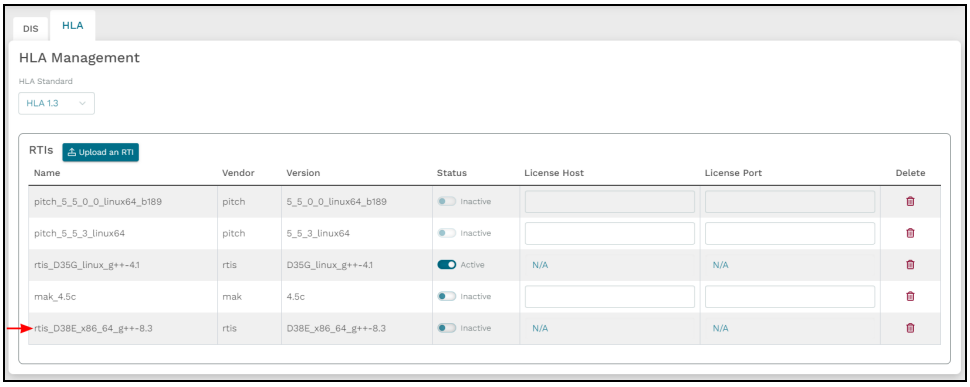

Figure 5: New RTI file in the RTIs table

***Note**: Telestra supports multiple RTI versions and vendors; however, only one may be active at a time.*

11. *(Optional)* To delete an RTI from the **RTIs** table, select the RTI's corresponding trash can icon (🗑).

3.2 Activate an RTI file

To activate a high-level architecture (HLA) run-time infrastructure (RTI) file, follow these steps:

1. On **HLA Management**, choose an RTI file from the **RTIs** table. In the **Status** column, turn on on the RTI's corresponding toggle switch, which now displays “Active:”

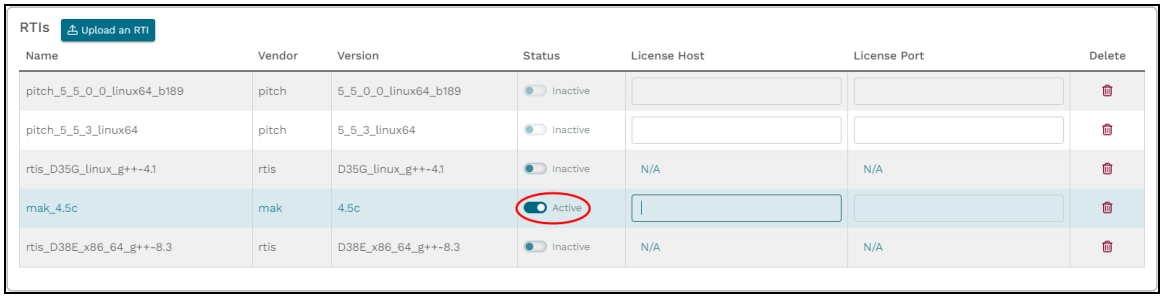

Figure 6: Active RTI file

2. *(Optional)* If your RTI vendor requires the RTI to withdraw a license from an external license server, do the following:

   a. In **License Host**, enter *xxx.xxx.xxx.xxx*, where *xxx.xxx.xxx.xxx* is the license server's IP address.

   b. In **License Port**, enter the license server's port number (e.g., 27001).

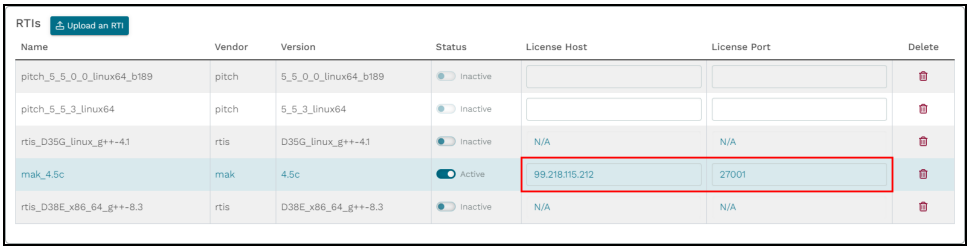

Figure 7: RTI License Host and License Port

***Note**: Alternatively, you may configure a Pitch RTI license in the Local RTI Component (LRC) .settings file. Go to Pitch documentation for more information.*

3.3 Set the HLA standard

To set the high-level architecture (HLA) standard, follow these steps:

1. From **Simulation** > **Protocols**, go to the **HLA** tab.
2. Log in with the following default credentials:

Username	Password
admin	astirules

3. *(Optional)* To view the hidden password, select **Show Password** (👁).

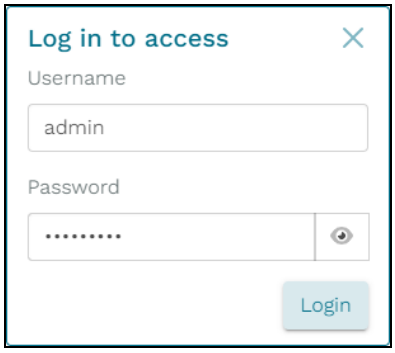

Figure 8: Login pop-up window

4. Select Login.
5. On **HLA Management**, select **HLA Standard**, and choose **IEEE 1516e** for HLAe or **HLA 1.3** for HLA 1.3.

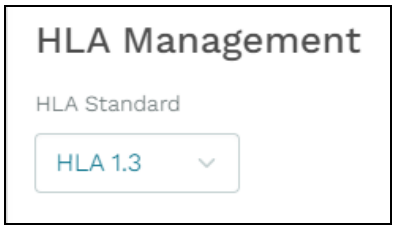

Figure 9: HLA Standard

4.0 Set up the HLA federation in Studio

The federation (.fed for HLA 1.3, .xml for HLAe) and mapping (.map.json) files on the Telestra server define the audio communication interface to the HLA environment. The federation file provides the RTI with new federation execution details and potential routing spaces, objects, and interactions. The mapping file tells the Telestra which objects and interactions are of interest to the ASTi federate and where to find that data. The mapping file also contains optional features related to object-naming conventions, Data Distribution Management (DDM), time stamp format, and more.

On the Telestra server, the mapping file reads the ASTi HLA radio environment class names and hierarchies. The code does not compile this information, which allows the mapping file to switch from one federation object model (FOM) to another. The mapping file provides object class attribute and interaction class names to the Telestra server federate. It then directs the Telestra server federate to find specific data in these attribute updates and interactions.

When designing your simulation object model (SOM) and FOM, consider units, data efficiency, reliability, and coherence. You can then incorporate ASTi's SOM into your federation object model, allowing the federation file and ASTi's mapping file to communicate. Alternatively, you can use ASTi's Real-time Platform Reference (RPR) FOM mapping file to leverage the RPR radio audio objects and interactions for radio communications.

The Telestra server can run HLA software on the same platform as the communication models (e.g., radios, network intercoms, engines). Figure 10, "Example HLA setup" below shows an example configuration:

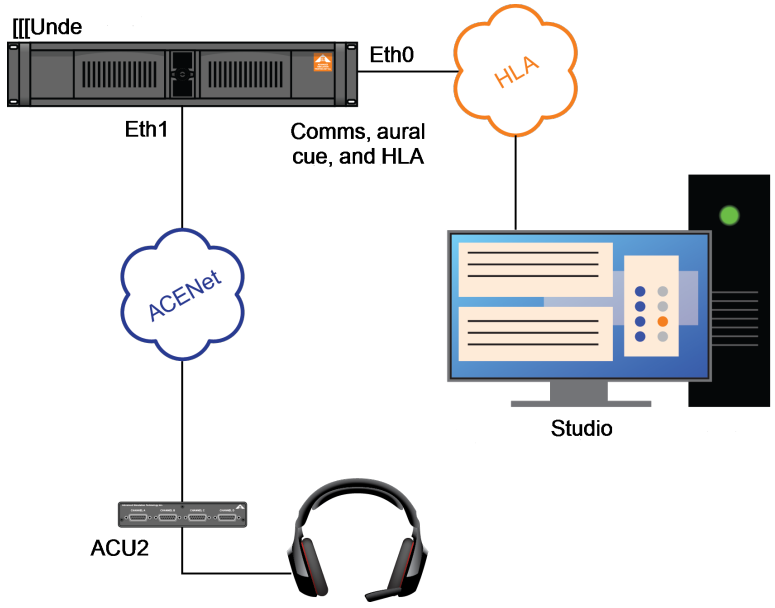

Figure 10: Example HLA setup

This chapter discusses how to:

- Add an HLA domain
- Add an HLA configuration file
- Add a DIS gateway
- Connect the layout to the domain and gateway
- Set up HLA radios

4.1 Add an HLA domain

To add an HLA domain in Studio, follow these steps:

1. On an Studio development workstation, from the taskbar, go to **Applications** > **ASTi** > **Studio**.

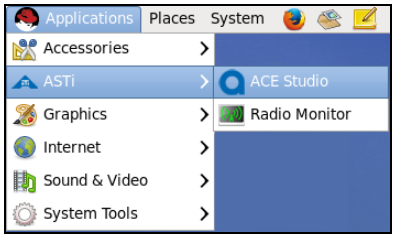

Figure 11: Studio navigation

2. From the top menu, go to **Project** > **Open**.
3. In **Open Project**, choose a project, and select Open.

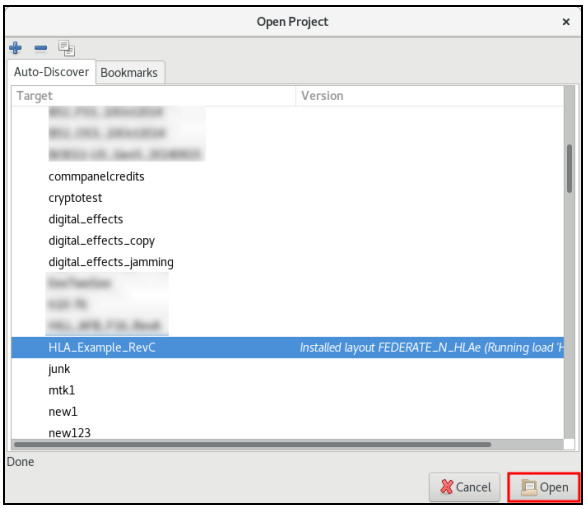

Figure 12: Open Project

4. From the left menu, go to **domains**.

5. Right-click the **Icon View** canvas, and select **Add...**.

6. In **New Domain plan**, enter a unique name for the domain. Select OK.

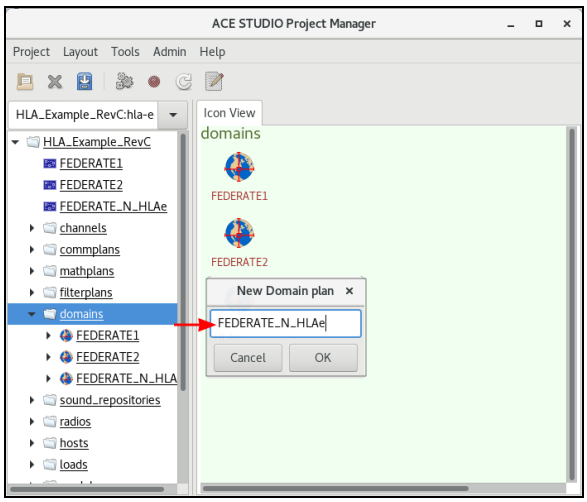

Figure 13: Add a domain

7. On **Names**, select **Add Domain** ( ).

8. In **Set Domain Name**, enter a unique name for the domain (e.g., **HLA_Domain**). Write down this name for future reference.

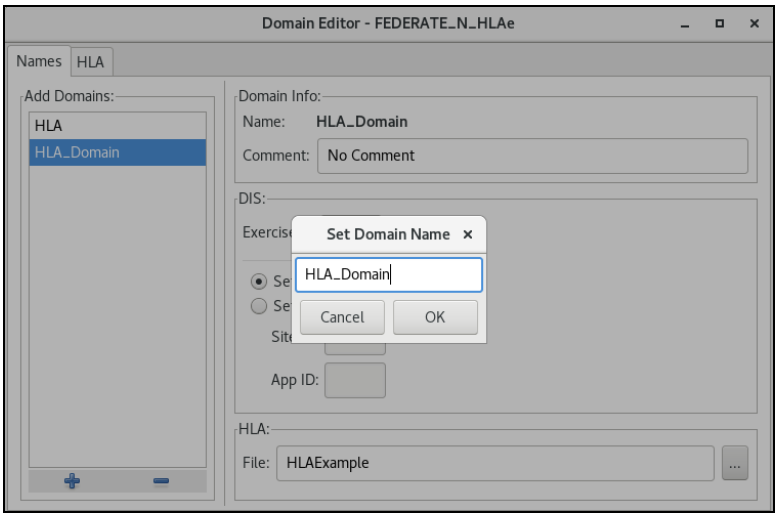

Figure 14: Set Domain Name

9. Under **DIS**, in **Exercise ID**, enter **1**.

10. *(Optional)* To automatically set IDs, select **Set IDs to Last Two IP Octets**. To set IDs manually, select **Set IDs Manually**. In **Site ID** and **App ID**, enter numbers defining the first two parts of the HLA 64-bit stream tag identifier.

    To learn more about stream tags, go to "REAL-TIME PLATFORM REFERENCE FEDERATION OBJECT MODEL" in SISO-STD-001.1-2015. To view stream tags in ASTi's simulation object model, go to the Example HLA Project in App Note #126: Sample Project for 7.3.0 and Later.

11. Select Apply and OK.

4.2 Add an HLA configuration file

The HLA configuration file defines the default federation name, federate name, RID file, mapping file, and federation file. Because the Telestra server can define and use multiple configuration files, you can create multiple HLA configurations and switch between them.

To add an HLA configuration file, follow these steps:

1. In **Domain Editor**, on the left, go to the **HLA** tab.
2. Select **Add HLA File** ( ).
3. In **Create HLA File**, enter a unique name for the file. Select OK.

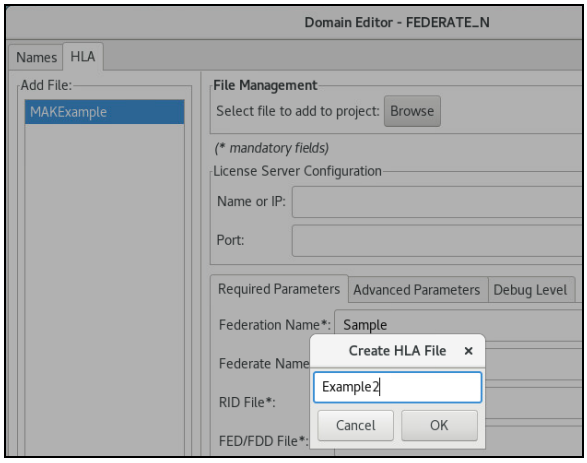

Figure 15: Create HLA File

4. To add one or more HLA file(s) to a project, under **File Management**, select **Browse**, and find the desired .rid, .map.json, .fed, .settings, or .xml files on your local system. For a list of common federation and mapping files, go to Appendix A, "External FED/FDD, RID, and mapping files" on page 32.

5. On **Required Parameters**, in **Federation Name**, enter the name of the default federation the Telestra server will join.

Important: *In Studio 7.3 and later,* ***Advanced Parameters****,* ***Debug Level****, and* ***License Server Configuration*** *on* ***Required Parameters*** *are not operational. You can set up this functionality in the mapping file or the Telestra web interface. For more information about mapping files, go to Appendix A, "External FED/FDD, RID, and mapping files" on page 32.*

6. In **Federate Name**, enter a unique federate name that the federation will see. Each Telestra server appears as a federate in the federation.

7. In **RID File**, **FED/FDD File**, and **Mapping File**, select the files you uploaded in Step 4.

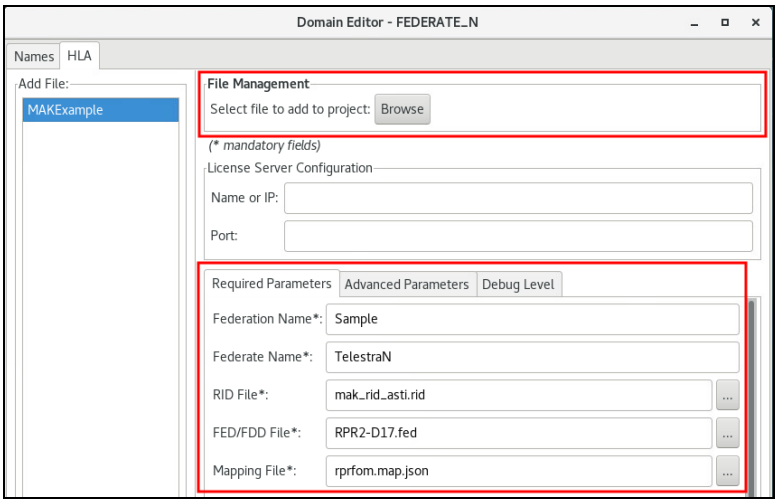

Figure 16: Required Parameters

8. Scroll down to **ASTi HLA Network**, and enter the following:

   a. ***IP****:* **127.255.255.255**

   b. ***Port****:* **54001**

   c. ***Int****:* **eth0**

9. Under **ASTi HLA Host Control**, in **Port**, enter the port number the host platform will use to control the ASTi federate. For more information, go to Section 5.0, "Join the federation at HLA run time" on page 20.

10. Select **Interface**, and choose an Ethernet interface (e.g., **eth0**).

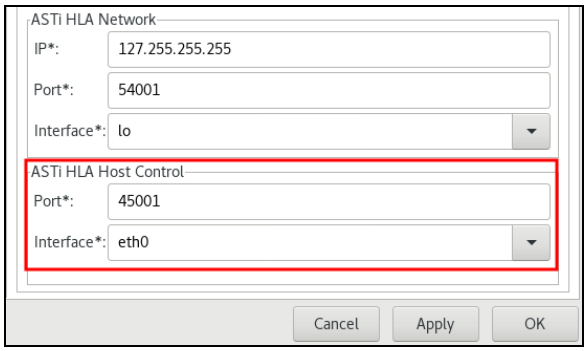

Figure 17: ASTi HLA Host Control

11. Select Apply and OK.

4.3 Add a DIS gateway

To add a DIS gateway, follow these steps:

1. From the left menu, go to **servers**.

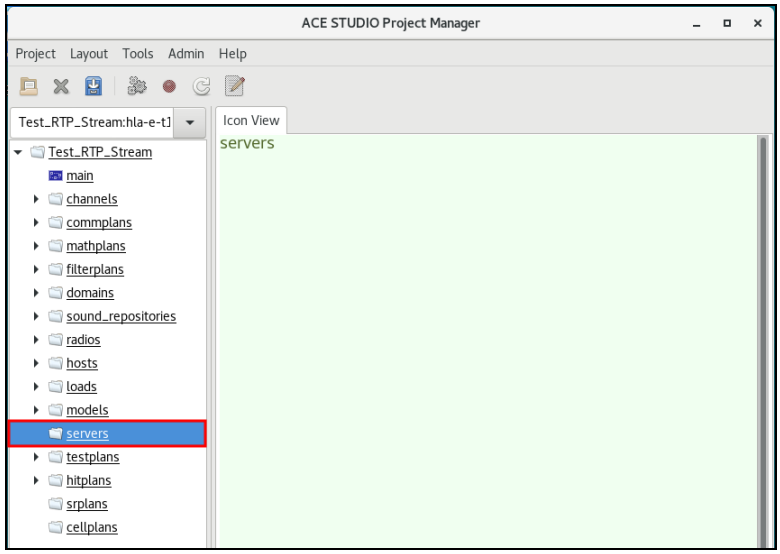

Figure 18: servers navigation

2. Right-click in the **Icon View** canvas, and select **Add**.
3. In **New Item**, choose **DIS Gateway**.

4. Enter a unique name for the gateway, and select OK.

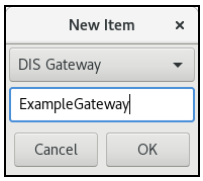

Figure 19: New gateway settings

5. On **Icon View**, open the new gateway. **DIS Gateway** opens.
6. Under **General**, in **version**, enter **7**.
7. Under **Interfaces**, select **DIS Interface**, and choose **lo**.
8. In **port**, enter **54001**.
9. In **main**, enter **127.255.255.255**.

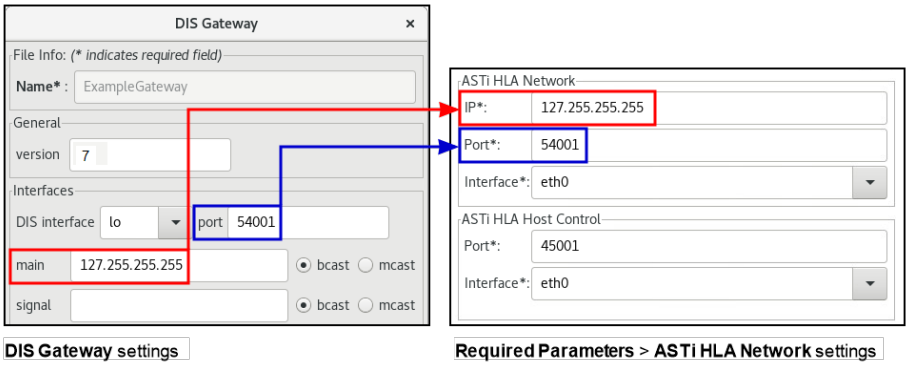

Figure 20: DIS Gateway settings

***Important**: Ensure **main** and **port** match the **ASTi HLA Network IP** and **Port** values that you set in Section 4.2, "Add an HLA configuration file" on page 12.*

10. Select OK.

4.4 Connect the layout to the domain and gateway

To set the project layout to use the new HLA domain and DIS gateway configuration file, follow these steps:

1. From the left menu, expand the project, and choose a layout.
2. Right-click the Telestra icon (💻), and select **Edit**.

3. On **CORE**, select **Domain**, and choose the domain you created in "Add an HLA domain."

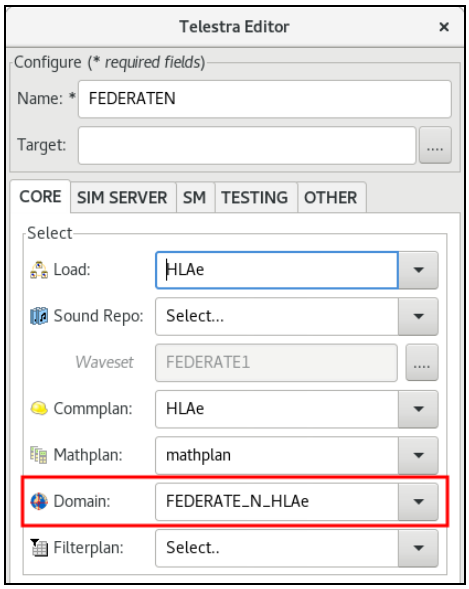

Figure 21: Choose a domain

4. Go to **SIM SERVER**.

5. Select **DIS Gateway**, and choose the DIS gateway you created in "Add a DIS gateway."

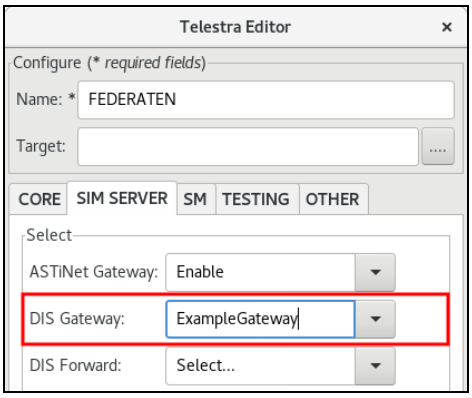

Figure 22: DIS Gateway

6. Select 

. The Telestra server can now output HLA network traffic.

7. From the toolbar, select **Project** > **Save**.

4.5 Set up HLA radios

Now that the Telestra server layout supports HLA traffic, you must choose which radios in the layout will use HLA. These instructions assume the Telestra server already has a model with radios. For more information about radio setup, go to the *Studio Components Reference Guide* at support.asti-usa.com/telestra/index.html.

To set up Telestra radios for HLA, follow these steps:

1. From the toolbar, select **Install Layout** ( ).

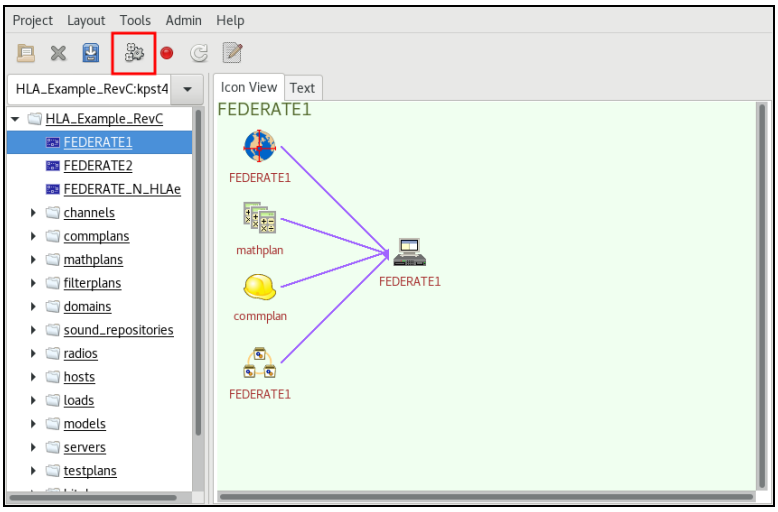

Figure 23: Install Layout

2. When the confirmation message appears, select **Yes**. Wait about 30 seconds for the layout to install.
3. In **Icon View**, open the Telestra server ( ).
4. In **System Load**, find a simulation model with radios.
5. Choose a radio in the model.

6. On **Views**, open a radio **Transceiver**.

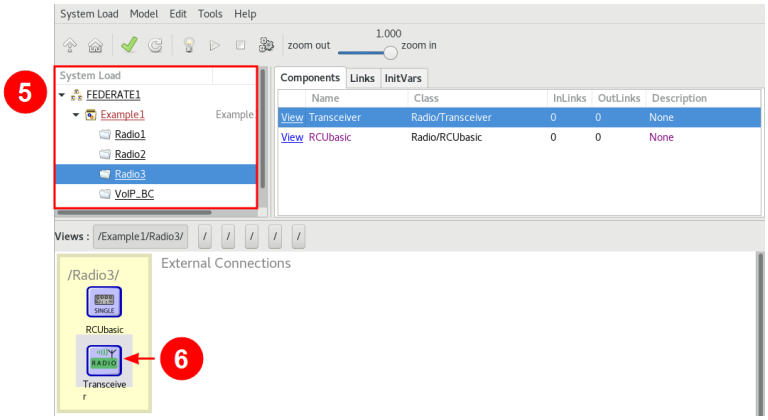

Figure 24: Open a Transceiver

7. To set the HLA domain name, find *DomainName*, and double-click **Value**.
8. In **Set Value**, enter the domain name you created in Section 4.1, "Add an HLA domain" on page 10. Select OK.

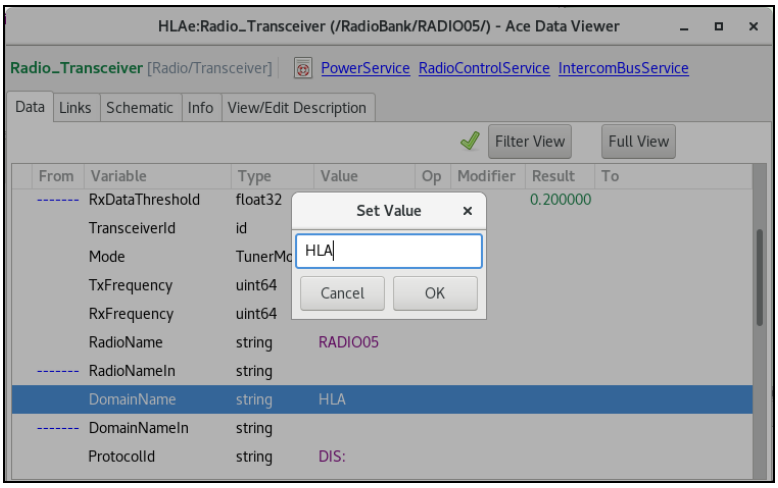

Figure 25: DomainName value

9. To set the HLA stream tag, find *ProtocolId*, and double-click **Value**.

10. In **Set Value**, enter a DIS: syntax command (e.g., **DIS:1.2.3.4**). If a specific stream tag isn't required, enter **DIS:**, which ensures the radio stream tag is unique within the federate. Select OK.

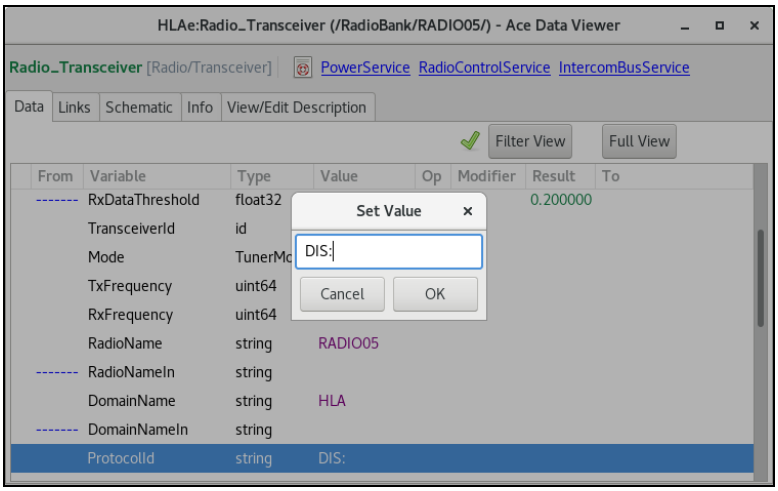

Figure 26: ProtocolID value

11. From the top toolbar, select **Apply changes to Project** (✔). When the confirmation message appears, select OK.

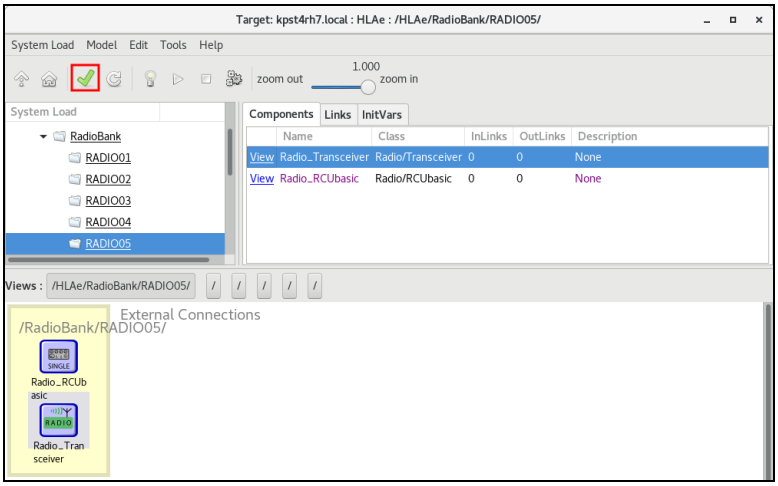

Figure 27: Apply changes to Project

12. Close the window. From the toolbar, select **Project** > **Save**.

5.0 Join the federation at HLA run time

After installing the run-time infrastructure (RTI) software in the Telestra web interface and setting HLA options in Studio, you must join the simulation federation via the Telestra server. Telestra's built-in ASTi HLA Host Control feature allows you to join the federation, view network activity, and complete other HLA tasks. ASTi also supports a host emulator that acts as a remote control interface but can control a local or remote host.

This chapter discusses the following topics:

- ASTi HLA Host Control
- Host emulator
- ASTi HLA Host Control commands and responses
- RTI Executive

5.1 ASTi HLA Host Control

The ASTi HLA Host Control feature allows you to control the Telestra server from a host server, instructing the federate to join or resign from a federation. Access HLA Host Control through a TCP/IP connection to the appropriate control port. Each federate has its own control port, as described in Section 5.1, "ASTi HLA Host Control " above. A control port can only support one TCP connection at a time.

After gaining control of the federate, the host can command the federate to:

- View current federate settings (e.g., federate name, federation name, .rid file)
- Change federate settings (e.g., federate name from FED_A to FED_B)
- Join or resign a federation
- Check run-time infrastructure (RTI) network activity
- Enable HLA interface ingress logging

For a full list of commands, go to Section 5.2, "ASTi HLA Host Control commands and responses" on page 22.

Changes that you make in HLA Host Control are temporary. Reinstalling the Studio project or layout overwrites your changes. To make permanent changes, modify the **HLA** tab in the **Domain Editor**. For more information about HLA settings in the **Domain Editor**, go to Section 4.2, "Add an HLA configuration file" on page 12.

The Telestra server interacts with the host computer using a three-way, TCP handshake process. As part of this process, a single host computer connects to a Telestra server acting as a TCP server. The Telestra server listens on the HLA Host Control TCP port for a SYN message from the host computer. When it receives the message, the Telestra server responds with a SYN-ACK message. In turn, the host responds with an ACK message.

After completing the handshake, the host and Telestra server can exchange data: the host computer sends command request(s), and the Telestra server responds accordingly. The host may terminate or reestablish the connection as needed. The host must reestablish a connection when you install an Studio project.

Figure 28, "HLA Host Control interaction" below shows the TCP/IP-based HLA Host Control interaction:

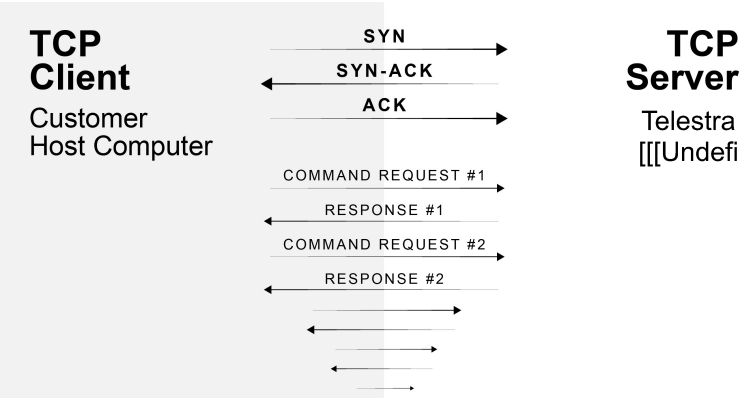

Figure 28: HLA Host Control interaction

5.1.1 Host emulator

The Telestra server and Studio development workstation include a host emulator called **hostemu**. This program acts as a console attached to the federate on a local or remote system. You can use **hostemu** to test the remote interface and verify that it's working correctly.

To run the host emulator, follow these steps:

1. Open a terminal.
2. Log in using one of the following default credentials:

   Telestra server:

Username	Password
admin	admin

Studio:

Username	Password
aceuser	aceuser

***Note**: If your organization changed these default credentials, check with your IT administrator for an updated username and password.*

3. To allow a Telestra server or Studio to act as a remote host to another Telestra server, do one of the following:

   - To connect to a local Telestra server, enter **hostemu** *yyyyy*, where *yyyyy* is the Telestra server's port number (e.g., 45001) from the Studio project.
   - To connect to a remote Telestra server, enter **hostemu** *yyyyy xxx.xxx.xxx.xxx*, where *xxx.xxx.xxx.xxx* is the remote Telestra server's IP address. This command allows a local Telestra server or Studio to act as a remote host to another Telestra server.

Once connected, **hostemu** acts as a console that sends user input to ASTi's HLA Host Control feature. If the host emulator's connection terminates, the socket automatically reopens and listens for a new connection.

5.2 ASTi HLA Host Control commands and responses

Commands and responses in the HLA Host Control feature are text-based, readable messages. While commands are not case sensitive, some data for the commands may be case sensitive (e.g., the federation file name).

The Telestra federate expects all command lines to terminate with a new line character. When the host sends two commands simultaneously, the federate executes them in order. The federate waits for the final new line before executing a command. To yield a new line, press the Spacebar, and then press Enter.

5.2.1 Status

Status displays the state of the federate. It shows if the federate is **JOINED** to a federation, **RESIGNED** from a federation, or **READY** to join a new federation. A **DOWN** response means that an error has occurred. Go to **/var/log/messages** on the Telestra server for more information.

Request

```
status
```

Response(s)

```
JOINED

RESIGNED

READY

DOWN
```

5.2.2 Name

When called with no parameters, **name** returns the names of the federate, the federation, the federation file, the RID file, and the mapping file.

Request

```
name
```

Response

```
FEDERATION ExampleFederation

FEDERATE ExampleFederate

FEDFILE /tmp/aced-
repoKwg3FX/repo/domains/FEDERATE1/hla/example.fed

RIDFILE /tmp/aced-
repoKwg3FX/repo/domains/FEDERATE1/hla/example.rid

CONVERT /tmp/aced-
repoKwg3FX/repo/domains/FEDERATE1/hla/example.map.json
```

***Note**: **CONVERT** displays the mapping file name.*

These variables receive their initial values from the **Project** domain configuration on project/layout install. After installation, you can temporarily rename the federation and federate or substitute a federate, RID, or mapping file for testing purposes. Reinstalling the Studio project or layout resets these changes to the default.

Request


```
name FEDERATION {New_Federation_Name}

name FEDERATE {New_Federate_Name}

name FEDFILE {New_Fed_File_Name}

name RIDFILE {New_Rid_File_Name}

name CONVERT {New_Mapping_File_Name}
```

The **name FEDFILE**, **name RIDFILE**, and **name CONVERT** parameters only define the file name. These file(s) must exist in the hla subfolder of the currently installed Telestra project domain. You do not need to enter the full path.

***Note**: To reference files (e.g., federation, RID) outside of the project repository, go toAppendix A, "External FED/FDD, RID, and mapping files" on page 32.*

5.2.3 Activity

Activity returns activity counters from the Telestra server. Use this command to verify the federate joined the federation or view troubleshooting information. On the Telestra server's host emulator, you can also press Enter to initiate **activity**.

Request

```
activity
```

Activity responds with the following:

- *Life count:* a general indicator of federate health. The counter increments once a second while the entity is joined to a federation. A fixed number indicates an error.
- *Attribute updates:* report the cumulative number of attribute updates that the federate sends and receives. The attributes include radio object parameters, such as radio power, world location, and frequency. An attribute update occurs whenever a field changes.
- *Interactions:* displays the number of interactions that the federate sends and receives.
- *Objects:* displays the number of transmitter objects on the HLA network. A transmitter's local counter reflects the number of local transmitter objects, while the RTI counter reflects the number of remote objects.

Response

```
Life count: 10
<Attribute Updates>
rx: 50 | tx: 25
<Interactions>
rx: 382 | tx: 378
<Objects>
Transmitters | local: 13 | rti: 13
```

5.2.4 Join

Join directs the Telestra server federate to join the federation and optionally specifies the name. You do not need to resign to join a different federation. Issuing a new **Join** command resigns the Telestra server federate from its current federation before attempting to join the new one. If the federate is already part of a federation, it automatically resigns and attempts to rejoin, even if the federation name is the same.

Two federates must use the same federation name to exchange information. If the federate has no name, the Telestra server uses the federation name from the Telestra server configuration file. To learn more about Telestra server configuration file settings, go to Section 4.2, "Add an HLA configuration file" on page 12.

Request

```
join {optional federation name}
```

Response(s)

```
join {federation name} OK

join {federation name} FAIL

join {federation name} FAIL {federation file path}

Error joining federation, see log
```

Join may return four types of responses:

- **join** *{federation name}* **OK**: the federate successfully joined the federation. The system is running, and you may now query the federate's **STATUS** and **ACTIVITY**.
- **join** *{federation name}* **FAIL**: the federate failed to join the federation. The following issues usually cause a generic failure to join:

  - The RTI Executive is not running.
  - The Telestra server's project does not specify a default federation name.
  - The RTI does not have a license.
  - A networking error occurs.
  - The federation file does not contain the names in the ASTi Radio SOM.
  - The Telestra server could not find the federation file in the specified location.
- **join** *{federation name}* **FAIL** *{federation file path}*: the federation can't retrieve handles for all of the object class and attribute names it requested from the RTI. Usually, this error occurs because the object and attribute names in the federation file are different in the mapping file. Contact ASTi at support@asti-usa.com for more information.
- **Error joining federation, see log**: For more information, go to **/var/log/messages** on the Telestra server. If the error is RTI-specific, contact support@asti-usa.com and/or the applicable COTS/GOTS RTI vendor.

5.2.5 Resign

Resign prompts the Telestra server federate to leave the federation. The federate can't transmit data, but it is still available on the network.

Request

```
resign
```

A successful resign responds with **RESIGN OK**. HLA-related errors (e.g., the system can't communicate with **rtiexec**) usually cause a **RESIGN FAIL**. Check the Telestra server's system log for more information.

Response(s)

```
RESIGN OK

RESIGN FAIL
```

5.2.6 Openlog

This command logs all HLA networking objects and interactions that an attribute or parameter receives. When enabled, the log writes to a file located at /var/log/hla_debug on the Telestra server. You might use **openlog** to debug, test, or verify networking operations. ASTi does not recommend leaving **openlog** enabled for normal operations.

***Important**: This command logs all ingress HLA data; do not enable under heavy networking conditions.*

Request

```
openlog
```

The following response shows an example of network logs for the **Radio Transmitter**, **Radio Receiver**, and **EncodedAudio Interaction** attributes from the ASTi SOM.

Response

```
6/20/2025 7:59:51 Reflect Attributes: Com-
municationSystem.RadioTransmitter

AntennaPatternData (7)[0]:

Encryption (1)[4]: 00 00 00 00 *....*

Frequency (2)[12]: 00 00 00 00 04 C4 B4 00 46 C3 50 00
*........F.P.*

RFModulation (3)[6]: 00 03 00 01 00 01 *......*

SpreadSpectrum (4)[0]:

TransmitState (5)[16]: 01 00 00 00 41 F0 00 00 01 00 00 01 E1 D4
9E 2B *....A..........+*

WorldLocation (6)[24]:

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
*................*

00 00 00 00 00 00 00 00 *........*

11/15/2019 8:02:51 Reflect Attributes: Com-
municationSystem.RadioReceiver

ReceivedPower (1)[4]: C3 87 00 00 *....*
```



```
ReceivedTransmitterIdentifier (2)[0]:

ReceiverOperationalStatus (3)[16]: 00 01 00 00 00 00 00 00 01 00
00 01 E1 D4 9E 2B *..............+*

11/15/2019 8:06:38 Receive Interaction: Radi-
oTransmission.EncodedAudio

StreamTag (2)[8]: 01 00 00 01 E1 D4 9E 2B *.......+*

EncodedParam (1)[208]:

00 01 1F 40 06 40 00 C8 7A ED 5A B6 91 86 81 80
*...@.@..z.Z.....*

83 8B 9D 34 11 07 01 00 03 0B 1D B4 91 87 81 80
*...4...........*

83 8B 9D 34 11 07 01 00 03 0B 1D B4 91 87 81 80
*...4...........*

83 8B 9D 34 11 07 01 00 03 0B 1D B4 91 87 81 80
*...4...........*

83 8B 9D 34 11 07 01 00 03 0B 1D B4 91 87 81 80
*...4...........*

83 8B 9D 34 11 07 01 00 03 0B 1D B4 91 87 81 80
*...4...........*

83 8B 9D 34 11 07 01 00 03 0B 1D B4 91 87 81 80
*...4...........*

83 8B 9D 34 11 07 01 00 03 0B 1D B4 91 87 81 80
*...4...........*

83 8B 9D 34 11 07 01 00 03 0B 1D B4 91 87 81 80
*...4...........*

83 8B 9D 34 11 07 01 00 03 0B 1D B4 91 87 81 80
*...4...........*

83 8B 9D 34 11 07 01 00 03 0B 1D B4 91 87 81 80
*...4...........*

83 8B 9D 34 11 07 01 00 03 0B 1D B4 91 87 81 80
*...4...........*

83 8B 9D 34 11 07 01 00 03 0B 1D B4 91 87 81 80
*...4...........*
```

5.2.7 Closelog

Closelog disables HLA network logging.

Request

```
closelog
```

Response

```
DONE
```

5.2.8 Bye

Bye prompts the federate to exit the host emulator; however, the federate remains joined to the federation.

Request

```
Bye
```

Response

```
(Closed by remote host)
```

5.2.9 Quit

Quit resigns the Telestra server from the federation and exits the host emulator. If successful, it does not return a response.

Request

```
quit
```


5.2.10 Help

Help returns a list of HLA Host Control commands:

Request

```
help
```

Response

```
Commands: activity, bye, closelog, help, join, name, openlog,
quit, resign, status
```

5.3 RTI Executive

The RTI Executive (**rtiexec**) is a global process that typically executes on a single platform. In a large-scale distributed simulation, this process often manages a large number of computing resources or federates. The **rtiexec** creates and destroys multiple federation executions that have different names and monitor federate health. For example, the **rtiexec** might coordinate timing (e.g., pause each federation execution simultaneously).

The **rtiexec** typically requires a desktop platform to operate (e.g., Linux, Windows computer, or Mac, depending on the RTI vendor). While **rtiexec** may operate on Studio, ASTi is not responsible for any errors that occur. For more detailed information specific to your configuration, go to your vendor's HLA documentation package.

This chapter provides a top-level overview of the following RTI Executive processes:

- (Optional) MAK RTI Executive
- Pitch Central RTI Component (CRC)

5.3.1 *(Optional)* MAK RTI Executive

The Telestra federate or Local RTI Component (LRC) can operate without the **rtiexec**; however, your program's configuration and/or the federate's RID or MTL file settings may still require its use. The **rtiexec** runs in conjunction with RTI Forwarders, which link different sites over the WAN or distribute TCP traffic load over multiple machines. They can also compress or bundle network packets to optimize for throughput or processor utilization.

***Note**: The following examples pertain to MAK RTIs, version 4.5c.*

To initiate an RTI Forwarder, at the prompt, run the following:

Request

```
./rtiForwarder -M -R {.rid file path} -l {.log file path}
```

To start the **rtiexec** process, run the following:

Request

```
./rtiexec -M -R {.rid file path} -l {.log file path}
```

To set up and run the MAK GUI, RTI Executive, or RTI Forwarder, go to MAK RTI documentation at Mak.com.

5.3.2 Pitch Central RTI Component (CRC)

The CRC (i.e., **rtiexec**) coordinates the entire federation and distributes work among the Local RTI Components (LRC). When a federate wants to join a federation execution, it connects to the CRC and receives information about the federation execution (e.g., other federates on the network). The Telestra federate or LRC requires a CRC on the network when the host requests to join. The join fails if no CRC exists. Monitor execution via the CRC's GUI or command-line interface.

To set up and run the CRC, go to Pitch pRTI documentation at pitchtechnologies.com/prti.

Appendix A: External FED/FDD, RID, and mapping files

A Telestra project or model typically contains the federation (.fed, .xml), RID (.rid, .mtl, .settings) and mapping files (.map.json). However, these files may also exist outside of a project. While the file(s) can reside in any system directory, follow the structure below for ease of implementation.

You may want to manage mapping files outside of the project to:

- Share mapping file(s) with other projects
- Administer project revision control outside of a Telestra project
- Leverage a storage area network (SAN) or file share
- Host scriptable or manageable files

***Note**: Creating a custom mapping file requires an advanced understanding of HLA concepts or completion of ASTi's HLA training course. If your application requires a custom or federation-specific mapping file, contact ASTi at support@asti-usa.com for additional guidance.*

Customers receive the following files in Telestra's baseline software:

File Type	Telestra File Path	Included Files
FED/FDD	/opt/hla/feds	• asti3_0.fed • asti3_1.fed • asti3_2.fed • ASTi_SOM_v32_1.xml • RPR2-D17.fed • RPR_FOM_v2.0_1516-2010.xml
RID	**/opt/hla/rids**	None
Mapping	/var/asti/hla/maps	• astisom30_revA.map.json • astisom_revA.map.json • rprfom_revA.map.json • t4rprfom_revA.map.json

Table 2: Included mapping files

Actual file names and revisions vary depending on the software release. For more information, go to your software installation. You may add custom application or program-specific files as needed. Since the above files are not part of a project, you must enable them using the host platform controlling the ASTi federate.

Configure the files using the **"name"** command available in the ASTi HLA Host Control TCP/IP connection. For more information about **"name"**, go to Section 5.2.2, "Name" on page 23. After installing the project, run the **"name"** command for each file that requires an external project reference.

To update the FED, RID, and mapping files to external project files, follow these steps:

1. To connect to the host, at the prompt, run **hostemu** *{port_number}*, where *{port_number}* is the host's port number.

```
[root@esxitarget1-730 rids]# hostemu 45001

Trying port 45001 on host 127.0.0.1

Connected to port 45001

Hit ctrl-c <enter> to quit
```

2. To check which files are in use, run **name**.

```
name
```

   The Telestra server responds with the current federation and federate as well as the .fed file, .rid file, and .map.json file (also known as a CONVERT file):

```
FEDERATION Sample

FEDERATE TelestraN

FEDFILE /var/tmp/aced-repoL3gLdj/repo/domains/FEDERATE_
N/hla/RPR2-D17.fed

RIDFILE /var/tmp/aced-repoL3gLdj/repo/domains/FEDERATE_
N/hla/mak_rid_asti.rid

CONVERT /var/tmp/aced-repoL3gLdj/repo/domains/FEDERATE_
N/hla/rprfom.map.json
```

3. To replace the current .rid file with an external project file, run **name RIDFILE** *{file_path}*, where *{file_path}* is the name and location of the new .rid file.

```
name RIDFILE /opt/hla/rids/mak_rid_asti.rid
```


In response, the Telestra server displays the new file path next to RIDFILE:

```
FEDERATION Sample

FEDERATE TelestraN

FEDFILE /var/tmp/aced-repoL3gLdj/repo/domains/FEDERATE_
N/hla/RPR2-D17.fed

RIDFILE /opt/hla/rids/mak_rid_asti.rid

CONVERT /var/tmp/aced-repoL3gLdj/repo/domains/FEDERATE_
N/hla/rprfom.map.json
```

Important: *If using Pitch RTI, the .settings extension replaces the .rid extension.*

4. To update the .fed or .fdd file to an external project file, run **name FEDFILE** *{file_path}*, where *{file_path}* is the name and location of the external .fed or .fdd file.

```
name FEDFILE /opt/hla/feds/asti3_2.fed
```

In response, the Telestra server displays the new file path next to FEDFILE:

```
FEDERATION Sample

FEDERATE TelestraN

FEDFILE /opt/hla/feds/asti3_2.fed

RIDFILE /opt/hla/rids/mak_rid_asti.rid

CONVERT /var/tmp/aced-repoL3gLdj/repo/domains/FEDERATE_
N/hla/rprfom.map.json
```

5. To update the conversion or mapping file to an external project file, run **name CONVERT** *{file_path}*, where *{file_path}* is the name and location of the external mapping file.

```
name CONVERT /var/asti/hla/maps/astisom.map.json
```

In response, the Telestra server displays the new file path next to CONVERT:

```
FEDERATION Sample
FEDERATE TelestraN
FEDFILE /opt/hla/feds/asti3_2.fed
RIDFILE /opt/hla/rids/mak_rid_asti.rid
CONVERT /var/asti/hla/maps/astisom.map.json
```

The host uses these files to join the federation until you update them or reinstall the project.

