Simulation Driver and Radar Recorder (SDRR)

User Reference Guide

March 8, 2024

JVN Tool Suite v13.4.19 Volume 4

JVN Communications Inc.
2511 Fire Rd., Suite A-4
Egg Harbor Township, NJ 08234
609-569-9255
www.jvncomm.com

TABLE OF CONTENTS

TABLE OF FIGURES

1. SDRR Overview

The Simulation Driver and Radar Recorder (SDRR) is a versatile tool that can be configured to provide injection of local, interfacility, and surveillance messages to a variety of National Air Space (NAS) Air Traffic Control (ATC) systems; connect and relay data between physical systems; and record incoming surveillance data. SDRR can also be configured to emulate and respond to messages from En Route, Terminal, and other flight and surveillance data systems. SDRR can be used to replay recorded surveillance files, inject custom static simulation scenarios, or for dynamic simulation. The SDRR Graphical User Interface (GUI) provides displays of the status and exchanged message for physically connected and simulated systems.

SDRR was designed by JVN Communications Inc. to provide flight and surveillance data communications to En Route, Terminal, and other ATC systems. Flight data support includes NAS messages through Simulation Services (SSRV) keystroke injection into the En Route Automation Modernization (ERAM) system, interfacility messages into the Standard Terminal Automation Replacement System (STARS), as well as Common Message Set (CMS) message into the En Route Data Distribution System (EDDS). The surveillance data types that SDRR can provide include Airport Surveillance Radar (ASR) Model-8 (ASR-8), ASR-9, ASR-9/Mode Select Beacon System (Mode S), ASR-11, Air Route Surveillance Radar (ARSR), Automatic Dependent Surveillance – Broadcast (ADS-B), Multilateration (MLAT), Wide Area Multilateration (WAM), Digital Altimeter Setting Indication (DASI) System, All Purpose Structured Eurocontrol Radar Information Exchange (ASTERIX), and Enhanced Traffic Management System (ETMS).

In order to inject interfacility and radar data, dedicated SDRR processors are connected directly to En Route External Communications Gateway (ECG) and STARS. In the En Route installation, interfacility and surveillance cards in the SDRR's slave processors are directly connected to the ECG modem splitters. For an SDRR in the STARS Interfacility and Radar Simulation (SIRS) installation, interfacility and surveillance cards in the SDRR's slave processors are directly connected to the STARS Line Shares and Radar Splitters. Additional SDRR processor installations can also access these physical devices via network connections and SDRR Connector relay configuration files.

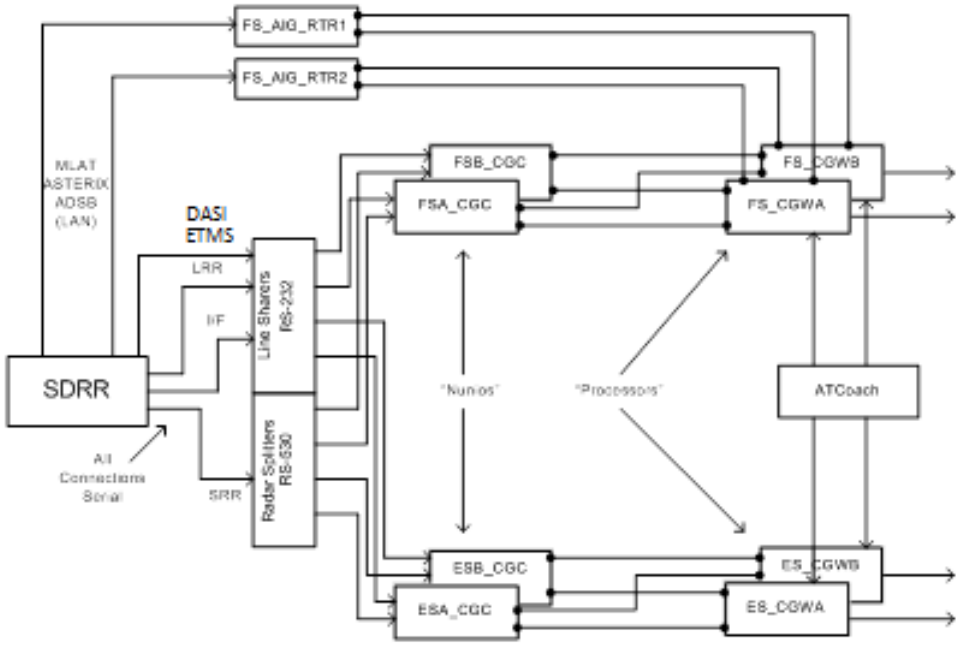

Figure 1. STARS Interfacility and Radar Simulator (SIRS) SDRR Installation

2. Getting Started

The processor with SDRR installed is configured to boot up to a user login screen. Users can enter a username and password, then click the login button or press **Enter** on the keyboard.

Figure 2. SDRR Processor Login

NOTE: *The 'root' user does not have access to the SDRR commands and utilities. If root access is needed while logged in, the user should either log out and log in again as 'root' or open a terminal window, type* ***su*** *and enter the 'root' user password.*

After a short loading period the KDE desktop will appear.

Figure 3. SDRR Processor Desktop

3. System Configuration

3.1. Environment Variables

SDRR uses several environment variables that set the locations of configuration files, scenarios, recordings, and log files.

Table 1. Environment Variables

Variable Name	Description	Default Location
SDRR_CONFIG_PATH	Location of SDRR configuration files.	/usr/local/cfg
SDRR_SCENARIO_PATH	Location of SDRR scenario files.	/usr/local/scenarios
RECORD_PATH	Location of recording files.	/usr/local/recordings
SDRR_LOG_PATH	Location of SDRR system log files.	/usr/local/log

3.2. Starting SDRR

SDRR can be started either by left clicking on the SDRR icon in the system task bar on the lower right side of the display (see figure below) or by typing **sdrr** at the command line in a terminal window:

```
> sdrr
```

Figure 4. SDRR Icon

3.2.1. SDRR Startup Wizard

Once SDRR is launched, the Select Configuration window appears. This window shows expandable directories in black and selectable configuration files in green. For configuration files to appear selectable green, they need to end in '.xml'. Users may need to expand the Name bar to the right to see the entire filename. The Comments are displayed from the comments section of the configuration file. Users can add or change these comments. Configuration files are located in the directory specified by the environment variable ${SDRR_CONFIG_PATH}. Once a configuration file is selected, the Next button becomes available.

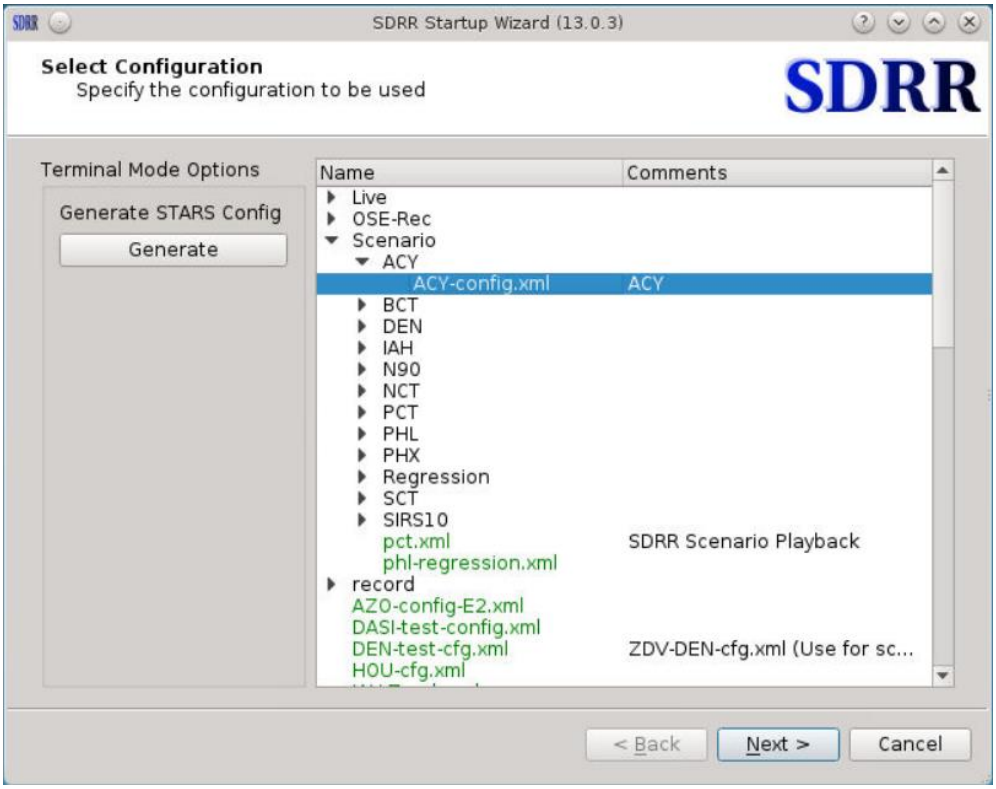

Figure 5. Select Configuration Window

After a configuration file is selected and the Next button is pressed, the Select Scenario window appears displaying directories in black and scenarios in green. The Location button on the left side allows users to select a scenario from the Local machine, Media (CD/USB/Tape), or Recording. On the Local machine, scenarios are located in the directory specified by the environment variable ${SDRR_SCENARIO_PATH}. The Recording option is for playback of recorded scenarios from the directory specified by the environment variable ${RECORD_PATH}.

Selecting a scenario is optional. The Skip Scenario button can be pressed when the window is first displayed and before any files or directories are selected. After pressing the Skip Scenario button, the

Next button becomes available and allows users to continue without specifying a scenario. Running SDRR without selecting a scenario can be used to test a configuration file or interfacility connections.

The example below illustrates the ACY directory expanded and the ACY-ARRDEP scenario selected.

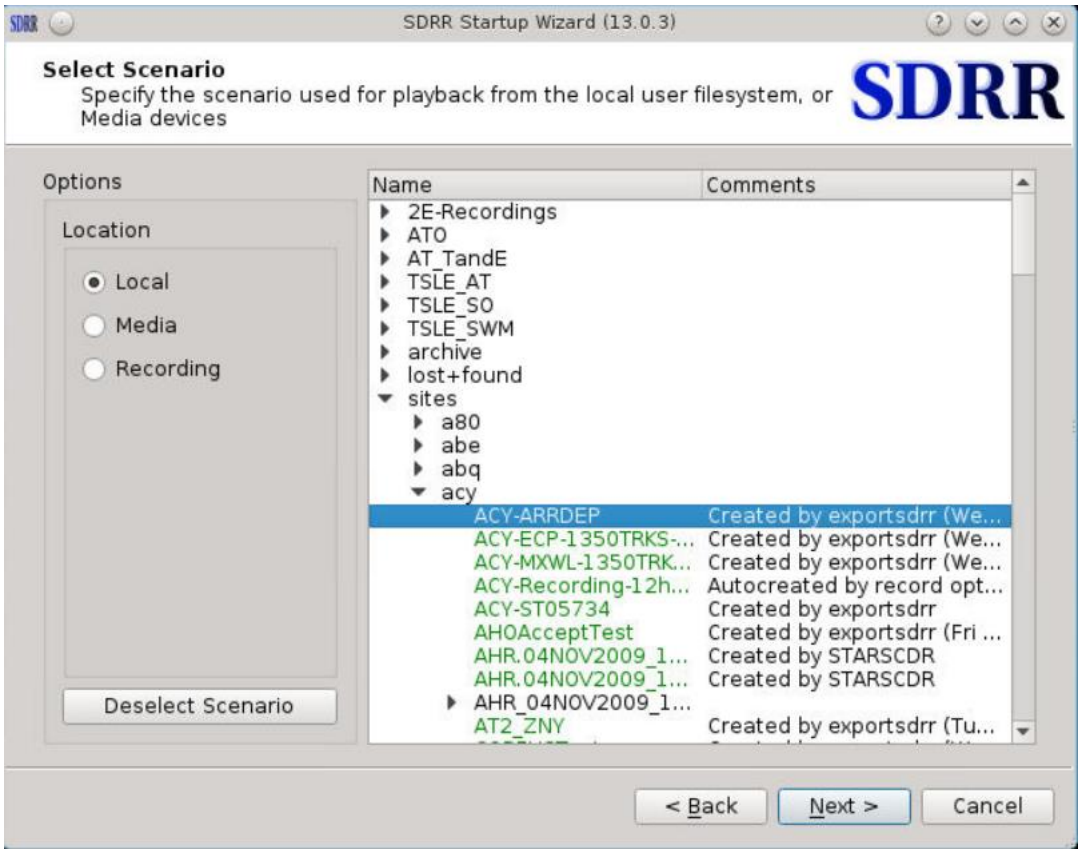

Figure 6. Select Scenario Window

After the Next button is pressed in the Select Scenario window, the Select Options window appears. If a scenario was selected in the previous window, the options will be populated from the scenario file (sdrr.xml) found in the scenario directory. This file can be modified to predefine some of the options.

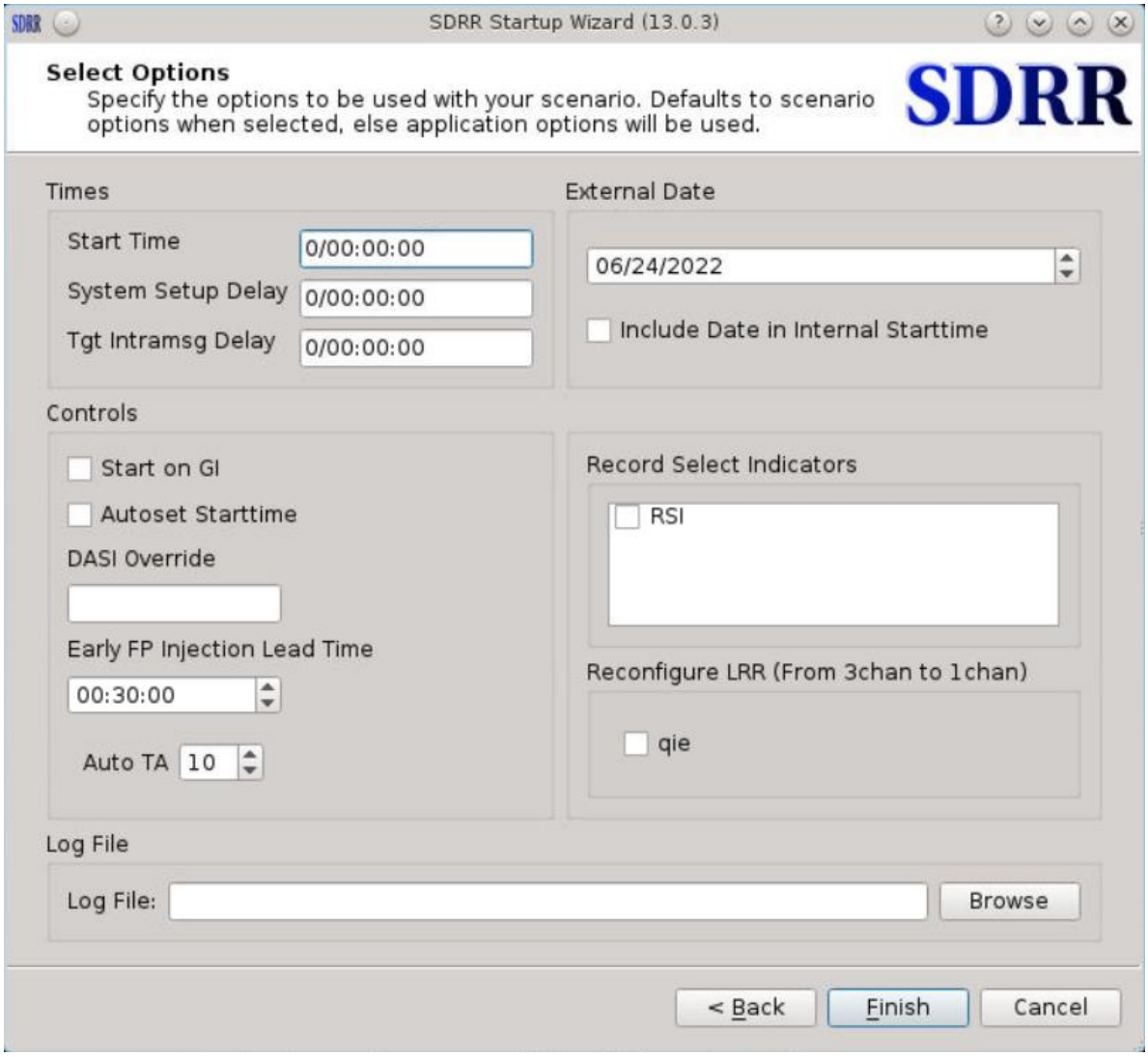

Figure 7. Select Options Window

Times

Start Time – Starting point of the scenario. The pre-populated time is read from the scenario file, sdrr.xml.

System Setup Delay – The amount of time to delay scenario injections after the start button is pressed.

Tgt Intramsg Delay – The amount of time between consecutive messages for the same target. This option is used to ensure that flight messages for a flight are not injected out of order.

External Date

Scenario date. The pre-populated date is the current system date and cannot be modified.

Include Date in internal Starttime – When this box is checked, the date is included in the SDRR internal start time.

Controls

Start on GI – Start upon receiving GI message from ERAM. ERAM sends GI messages to attached devices when it transitions to an operational state. When this option is selected, SDRR will start running upon receipt of this message.

Autoset Starttime – SDRR will determine start time based on first radar message (time stamp from the .srv files). This option is usually used for playback of live radar recordings.

DASI Override – Sets the default DASI value.

Early FP Injection Lead Time – The default amount of time prior to a target start time to inject the flight plan message. For targets starting within the window of "Scenario Start Time" to "Early FP Injection Lead Time", the flight plans will be immediately injected upon pressing Start.

Auto TA – The default time (in seconds) for SDRR to send a TA message after receiving a TI.

Record Select Indicators

Only targets and messages tagged with the selected RSIs will be injected.

Reconfigure LRR (From 3chan to 1chan)

SDRR will search the configuration file for any long range radars that are set for three channels. Checking the box of the long range radar will change it to a single channel radar. If a channel has been previously set to 1 channel in the configuration file, it will show up checked by default.

Log File

Location of the SDRR log file. The log file can be renamed to a scenario related name for easier tracking.

Once all the desired options are specified and the Finish button is pressed, the SDRR GUI is launched with the selected configuration and, optionally, a scenario loaded.

3.2.2. Command Line Startup

To bypass the Startup Wizard, SDRR can be started from the command line of a terminal window with a configuration file, scenario, and other optional parameters specified. To launch SDRR, enter:

```
> sdrr cfgFile.xml -s sdrrScenFile.xml [options]
```

To launch a version of SDRR that is not the default version, enter:

```
> /usr/local/jvn.x.x.x/bin/sdrr cfgFile.xml -s sdrrScenFile.xml [options]
```

3.2.3. Dynamic Simulation

For dynamic simulation, SDRR must be started from the command line in order to set the parameters for message exchange with the DYSIM executable, simDriver. The simDriver executable must also be started with the corresponding parameters. To launch SDRR, enter:

```
> sdrr cfgFile.xml --start –noscenario --cmdDev=tcps://<address>:<port#>?serverMode=1
--tgtDev=tcps://<address>:<port#>?serverMode=1
```

Note that there may be a need to multiplex the cmdDev and tgtDev definitions to a second device. For example, one SDRR instance may be connected to two instances of simDriver:

```
> sdrr cfgFile.xml --start --noscenario
--cmdDev=(tcps://<address1>:<port1>?serverMode=1+tcps://<address2>:<port2>?serverMode=1)
--tgtDev=(tcps://<address1>:<port3>?serverMode=1+tcps://<address2>:<port4>?serverMode=1)
```

Or the output could also be multiplexed to a file:

```
> sdrr cfgFile.xml --start --noscenario
--cmdDev=(tcps://<address1>:<port1>?serverMode=1+/tmp/commands.jvn)
--tgtDev=(tcps://<address1>:<port3>?serverMode=1+/tmp/targets.jvn)
```

Also note that the cmdDev option is allowed multiple times on the command line:

```
> sdrr cfgFile.xml --start --noscenario
--cmdDev=tcps://<address1>:<port1>?serverMode=1
--cmdDev=tcps://<address2>:<port2>?serverMode=1
--tgtDev=tcps://<address1>:<port3>?serverMode=1
```

3.2.4. Command Line Options

For a list of the command line options and parameters available, the **sdrr** command can be entered with the **help** parameter:

```
> sdrr --help
```

Figure 8. SDRR Command Line Options

The most common program parameters are described in Table 2below. For a complete list of options, refer to Figure 8 above.

Table 2. Program Parameters

Parameter	Description
`cfgFile.xml`	At least one XML configuration file is required to start SDRR. A relative path from ${SDRR_CONFIG_PATH} is assumed, unless an explicit path is given. Multiple files may be specified. See section 8 for further details.
`--cmdDev=tcp://<address>:<port#>?serverMode=1`	Directs SDRR to listen for simulated flight data messages from DYSIM over a TCP connection on the named address and port. The DYSIM executable, simDriver, must be started with the corresponding parameter. This parameter may be specified multiple times.
`--tgtDev=tcp://<address>:<port#>?serverMode=1`	Directs SDRR to listen for simulated target position data from DYSIM over a TCP connection on the named address & port. The DYSIM executable, simDriver, must be started with the corresponding parameter.
`--precipDev=tcp://<address>:<port#>?serverMode=1`	Directs SDRR to listen for simulated precipitation data from DYSIM over a TCP connection on the named address & port. The DYSIM executable, simDriver, must be started with the corresponding parameter.
`--connectionFile=sdrrconnector.xml`	File that defines a relay between facility interfaces in different physical test beds or lab strings.
`--deviceFile=deviceFile`	File that defines devices used in variable configurations.

Parameter	Description
`--recordingDir=directory`	Directs SDRR to read in recorded files from the specified directory.
`--scriptDefinitions=file`	Selects the scriptDefinitions file.
`-s sdrrScenFile.xml`	Start SDRR with an exported scenario. A relative path from ${SDRR_SCENARIO_PATH} is assumed, unless an explicit path is given.
`-n or --noscenario`	Start SDRR without specifying a scenario. Injections can come from a non-scenario source, such as DYSIM.
`--disableFormattedLogging`	Removes extra white space and makes SDRR log files easier to read.
`--start`	Directs SDRR to begin running immediately upon launch.
`--norappi`	Directs SDRR to not display a RAPPI tab.
`--nofullscreen`	Not full screen mode.
`--headless`	Runs SDRR without launching the GUI.
`--version`	Displays SDRR version.
`--help`	Displays application parameters.

4. Error Status

When starting SDRR, there may be possible errors that occur while the configuration or the scenario is loading. Dialog boxes will be displayed to indicate the cause of the errors. Below are some possible errors that may pop up while loading SDRR.

The following error will be displayed when there is a process already running and using the same instance on the SDRR machine.

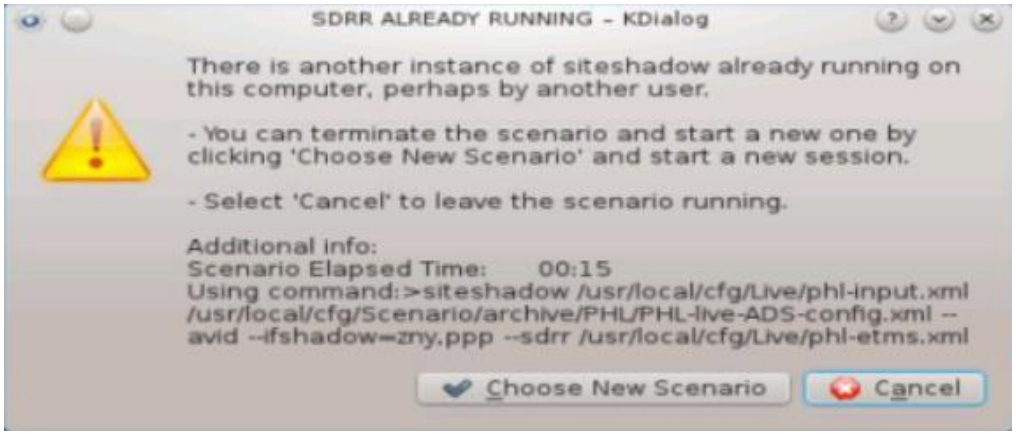

Figure 9. SDRR Already Running Error

The following error is displayed when a physical device is unavailable. Ensure that each physical device in assigned to a single source in the configuration file and that no other instances of SDRR are connected to the device.

Figure 10. Device Busy Error

5. Scenario Playback

5.1. Simulation Modes

SDRR can be configured in multiple ways depending on which systems will be physically connected and which systems will be simulated by SDRR. This is defined in one or more configuration files. The configuration must include either a simulated ERAM or an ERAM interface. In the case of a simulated ERAM, SDRR is configured to act as an En Route center. SDRR typically emulates an ERAM system generating messages and responses to interfacing systems accordingly; however, in Terminal environment testing, SDRR can be configured as a legacy Host system. This can provide a simpler En Route center simulation when connections to other external systems are not required.

5.1.1. Direct Mode

When SDRR is configured to simulate all En Route and Terminal systems, the configuration is referred to as direct mode. In addition to the standard interfacility and surveillance data, SDRR can generate other data formats. For En Route data, SDRR generates and responds to CMS messages, emulating the interface that would be provided by ERAM to an EDDS. For Terminal data, SDRR generates and responds to AIG messages, emulating the interface provided by STARS. This simulation mode can be used to test systems such as TBFM when an ERAM test bed and STARS string are not available.

5.1.2. Mixed Mode

In mixed mode, SDRR is configured to emulate En Route systems while driving one or more Terminal systems. For the En Route data, SDRR generates CMS messages for injections into an EDDS, emulating the feed that would be provided by ERAM. For terminal data, SDRR sends IFDT messages and radar data to a STARS system through a directly connected SDRR processor, also known as SIRS.

5.1.3. Indirect Mode

In indirect mode, SDRR is configured to drive En Route and Terminal systems. For the En Route data, SDRR uses SSRV command injection, interfacility messages, and surveillance data to drive an ERAM system. The interfacility and radar data are sent through an ECG to an ERAM test bed or with an ECG emulation to an ERAM virtual lab or ERAM-in-a-Box (EIB). For terminal data, SDRR sends surveillance data to a STARS system while the IFDT messages are sent to STARS by the ERAM system.

5.2. Graphical User Interface

Once launched successfully, the main SDRR Graphical User Interface (GUI) appears. The GUI is made up of a main menu bar, date and time clock, and display tabs.

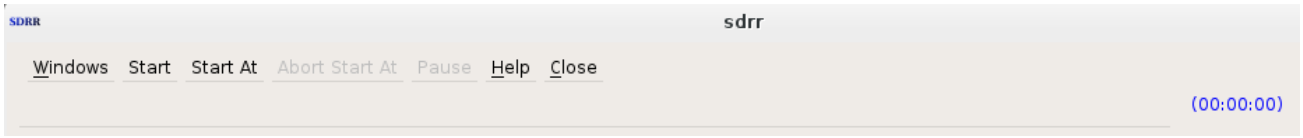

Figure 11. SDRR Menu Bar

The menu bar includes the following options:

Windows

When multiple windows are present on a particular display tab, those windows can be arranged using the options Cascade or Tile.

Start

Start the scenario execution immediately.

Start At

Start the scenario execution at the specified time.

Abort Start At

Interrupt the 'Start At' countdown.

Pause

Pause the scenario execution. The Start option becomes available.

Help

The Help menu provides an option to select **About**. The **About** option displays the "About SDRR" dialog which shows the version of SDRR, and the date and time that the SDRR executable was created.

Close

Stop the scenario execution and close the SDRR GUI.

Figure 12. About SDRR

When SDRR is running, a scenario runtime clock (displayed in blue and in parentheses) and the current date and external system time are displayed in the upper right corner. Note that this runtime clock is not synced to the start of the scenario running in the DYSIM executable, if running in dynamic mode.

Figure 13. Runtime Clock

The display tabs include a Status tab, and various other tabs determined by the SDRR configuration file. Right clicking in the message log areas of each display tab launches a pop-up dialog with the following options:

Copy

Place any selected text into the copy buffer.

Select All

Select all the text in the current display tab message log area.

Find

Open a search bar at the bottom of the current display tab message log area.

In addition to accessing the Find function as described above, it can also be initiated by pressing the Ctrl and F keys while in a message log are of any tab. The figure below shows the search bar.

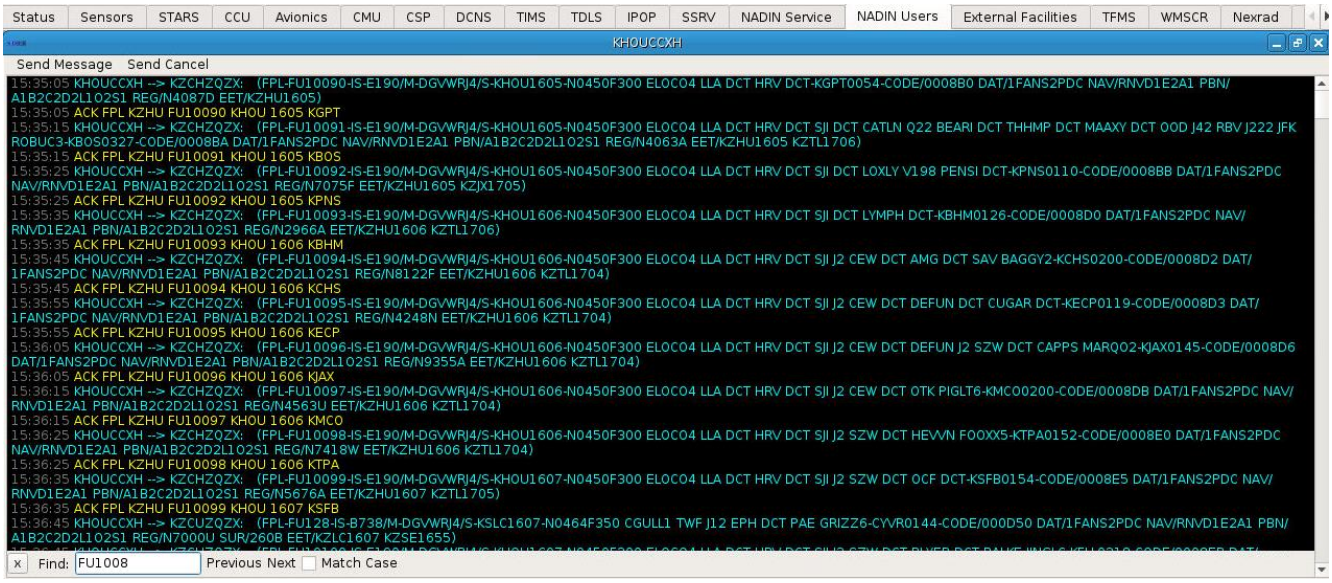

Figure 14. Find Function Search Bar

5.2.1. Status Tab

Upon startup, the SDRR GUI displays the Status Tab. The left side of the status tab shows Device Status. This is a direct reflection of the contents within the SDRR configuration file(s). The device or facility types are labeled in white. The physical or simulated devices are displayed in green. The prefix "pipe" indicates an internal simulated device. In the figure below, the Radar and SVOL list the defined surveillance sources, STARS and ZNY list the defined interfacility sources, and DASI and ETMS list additional non-surveillance sources. The device to which each source is assigned is listed to its right.

To further illustrate, the ZNY:AAA source under the STARS heading is shown with a pipe device. The same device is listed for the AAA source under the ZNY heading. This indicates that ZNY ARTCC and AAA STARS are configured to communicate via an SDRR simulated device. In the case of PPP, the live site, the configured device is a physical interface card, /dev/if0. This physical device connects a simulated En Route center, ZNY, to a live STARS string configured as PPP. A terminal controller at PPP could initiate a handoff of a flight through ZNY to AAA and AAA could send an accept (DA) or a reject (DR) response back to PPP. When SDRR is started, the link turns green as soon as a clock signal is detected. If the device is red, it is an indication of a down link.

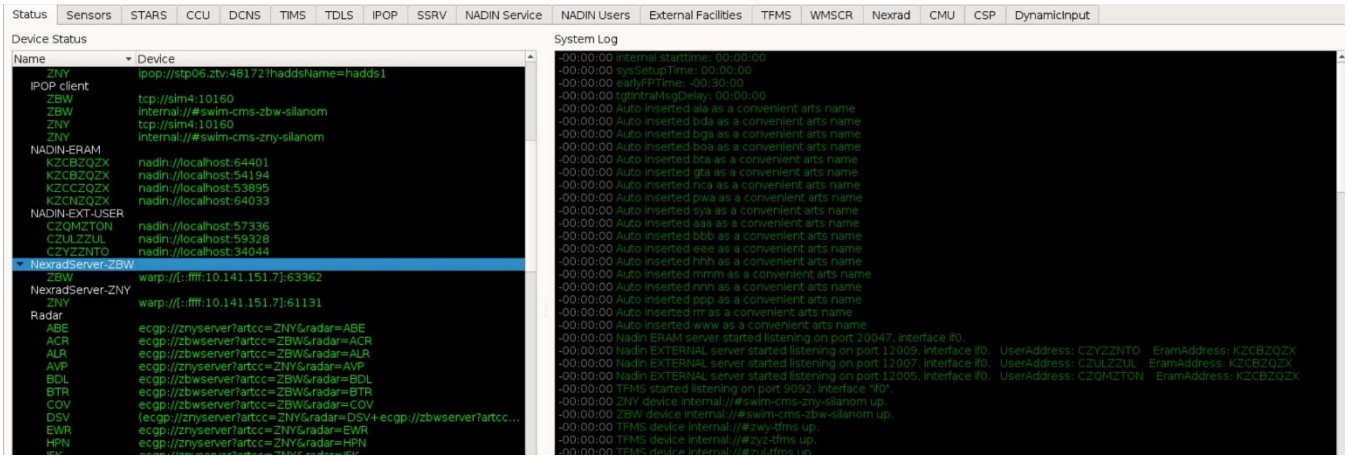

Figure 15. SDRR Status Tab

The right side of the status tab shows the System Log**.** The log displays error and warning messages about the scenario. Green messages indicate success; yellow and red messages indicate a problem. Not all problems will affect the success of the scenario but should be noted and may need to be investigated. These messages are also written to the SDRR log file. The log file of each run can be found in the directory specified by the environment variable ${SDRR_LOG_PATH} and will include a timestamp in the filename.

5.2.2. Sensors Tab

The Sensors tab is displayed when SDRR is configured with any surveillance devices. This tab displays a window for each radar and service volume defined in the configuration file. The window for each surveillance device shows details for radar channels, counts, message types, and errors. The **ResetStats** button is available to reset the channel counts to zero. This does not affect the output data.

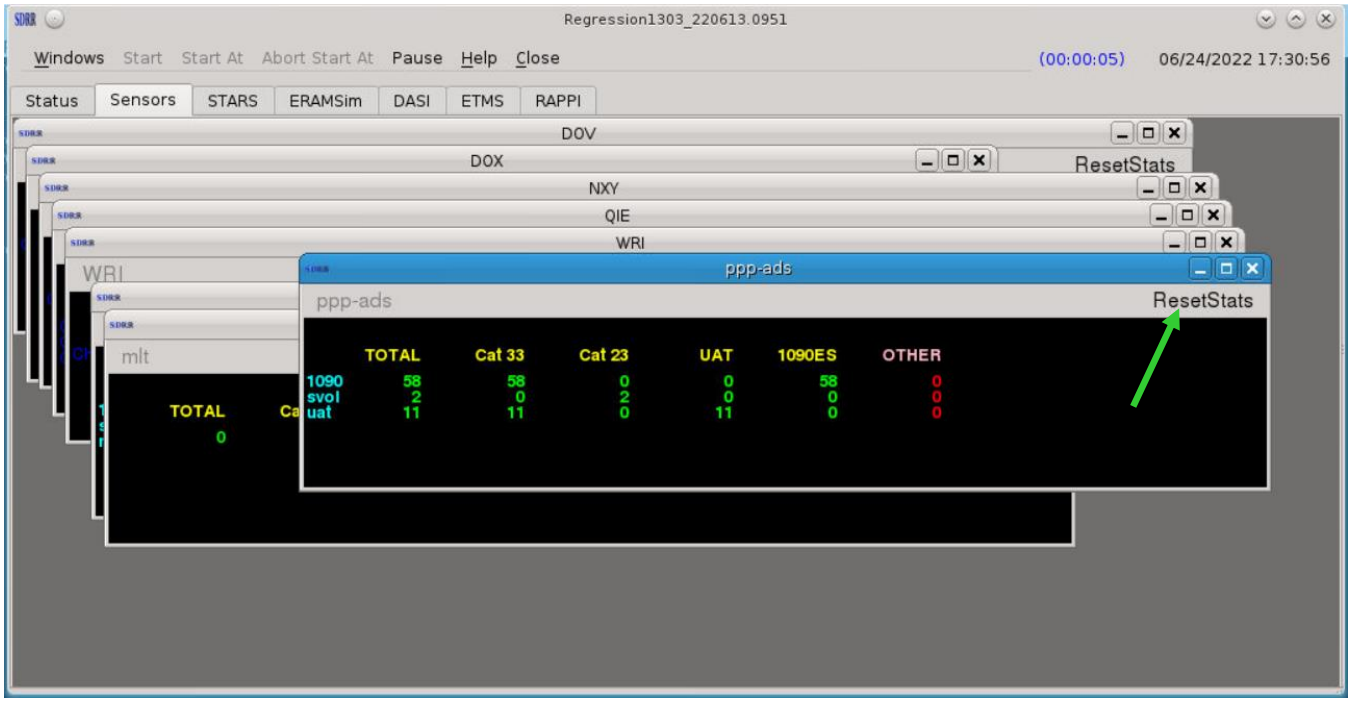

Figure 16. Sensors Tab

The figure below shows the window for radar QIE. The column on the left, in blue, lists the channels that are adapted for this radar device. The top row, in yellow, lists the message types. The numbers displayed in green are message counts for each of the feeds with good data. The three columns at the far right displayed in red indicate errors in the data.

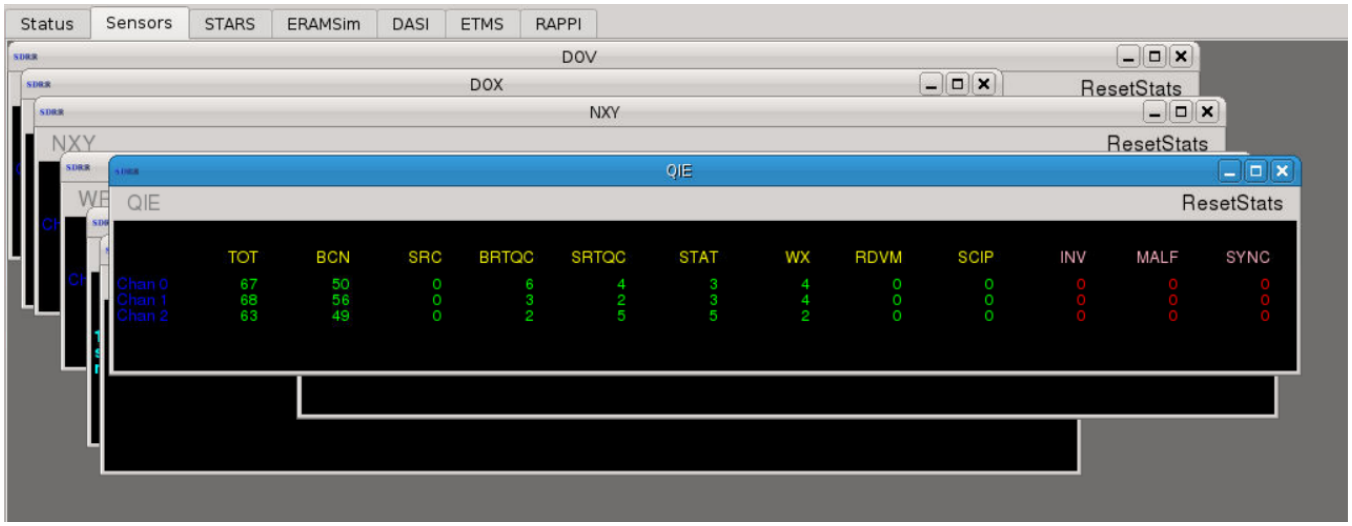

Figure 17. Surveillance Device Window

5.2.3. ERAMSim Tab

The ERAMSim tab is displayed if SDRR is configured for En Route simulation for testing where an ERAM connection is not required, such as Terminal or TBFM testing. This tab provides the user with all of the messages that are exchanged between a simulated ERAM and its neighbors. Within the ERAMSim tab, a window is displayed for each ERAM facility included in the SDRR configuration file. Each ERAM window displays tabs for an internal log and the configured terminal facilities and adjacent En Route centers. If the simulated ERAM facility stanza in the SDRR configuration file includes an external eddserver, then the ERAM facility window will also include a tab labeled EDDS (see section 5.2.3.2 EDDS Tab).

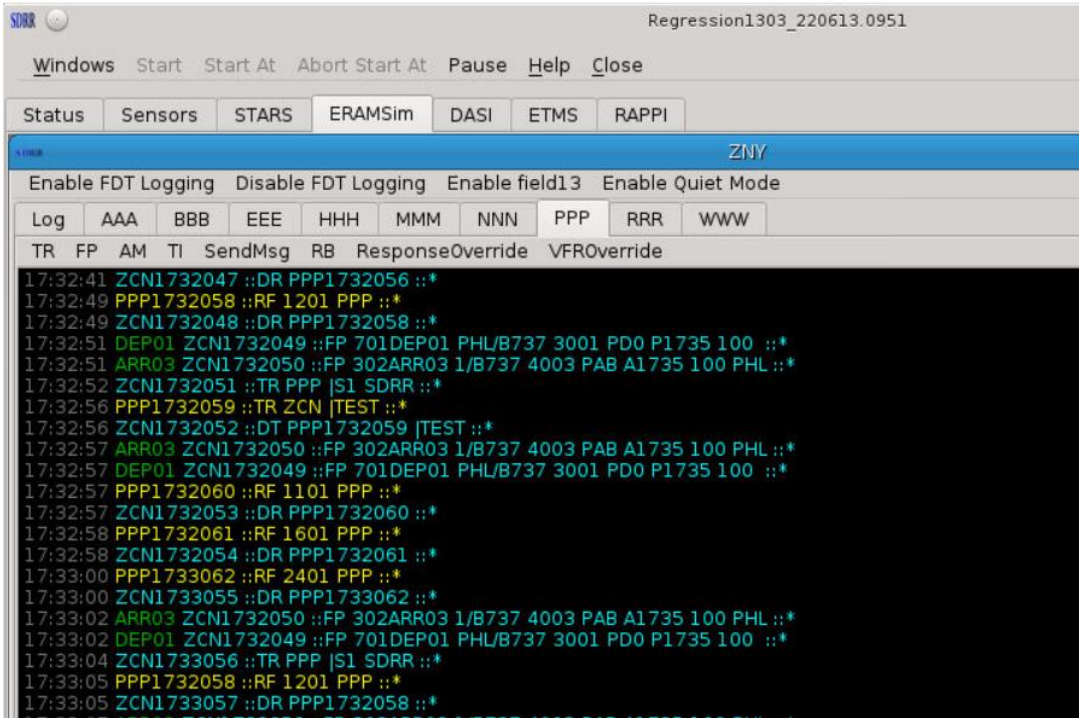

Figure 18. ERAMSim Tab

In each ERAM facility window, the following buttons are available: Enable FDT Logging, Disable FDT Logging, Enable field13, and Enable Quiet Mode.

Figure 19. ERAM Facility Window

Enable FDT Logging

Enables Flight Data Table logging in the Log Tab.

Disable FDT Logging

Disables Flight Data Table logging in the Log Tab.

Enable field13

Enables Field13 on FP Messages.

Enable Quiet Mode

Disables TR / DT Logging.

Within each facilities tab, the following buttons are available specific for the facility:

TR

Sends Test Message, TR, to the facility.

FP

Sends a custom flight plan message for an aircraft.

AM

Sends a custom amendment message for an aircraft.

TI

Sends a custom Initiate Transfer message, TI, for an aircraft.

SendMsg

Sends a custom IFDT message, such as FP, AM, DR, TA, or any other valid message.

RB

Restore Base. Resends all previously sent flight plan messages.

ResponseOverride

Overrides the automatic message response function. The default function is for the simulated facility to send responses of acceptance, DA. This override allows reject, DR; retransmit, DX; or no responses to be sent upon receipt of a message requiring a response.

VFROverride

Controls the automatic response to VFR requests; including beacon code, fix, or STARS destination.

5.2.3.1. Terminal Facility Tab

In the figure below, ZNY is configured as the ERAM facility. The terminal facilities within ZNY airspace are configured to communicate with ZNY. This is defined in the SDRR configuration file based on ERAM and STARS adaptation.

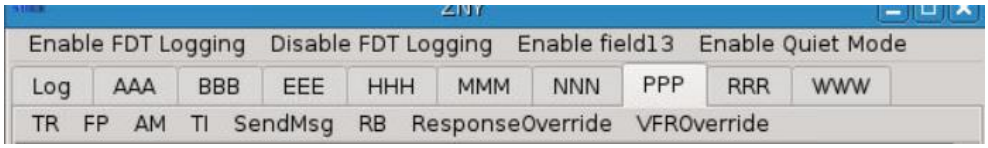

Figure 20. Terminal Facility Tabs

In each terminal facility tab, the following buttons are available: TR, FP, AM, TI, SendMsg, RB, ResponseOverride, and VFROverride. These buttons give users the ability to send any interfacility messages as well as the ability to control how the simulated facilities respond. The messages are sent in real time, providing many advantages during testing.

TR

Sends Test Message, TR, to the facility.

FP

Sends a custom flight plan message for an aircraft.

AM

Sends a custom amendment message for an aircraft.

TI

Sends a custom Initiate Transfer message, TI, for an aircraft.

SendMsg

Sends a custom IFDT message, such as FP, AM, DR, TA, or any other valid message.

RB

Restore Base. Resends all previously sent flight plan messages.

ResponseOverride

Overrides the automatic message response function. The default function is for the simulated facility to send responses of acceptance, DA. This override allows reject, DR; retransmit, DX; or no responses to be sent upon receipt of a message requiring a response.

VFROverride

Controls the automatic response to VFR requests; including beacon code, fix, or STARS destination.

5.2.3.1.1.Error Messages

Messages in red text indicate a message processing error. In the figure below, flight plan messages sent from ZNY to PPP were rejected by STARS. This could indicate an adaptation mismatch between the simulation and the live STARS, duplicate flights in the STARS database, etc. When the STARS system is populated with surveillance targets without the accompanying flight plans, PPP sends an RF (request flight plan) message to ZNY. Since the flight plans were rejected, SDRR does not have the mapping of aircraft identification to the requested beacon code and the RF message is rejected.

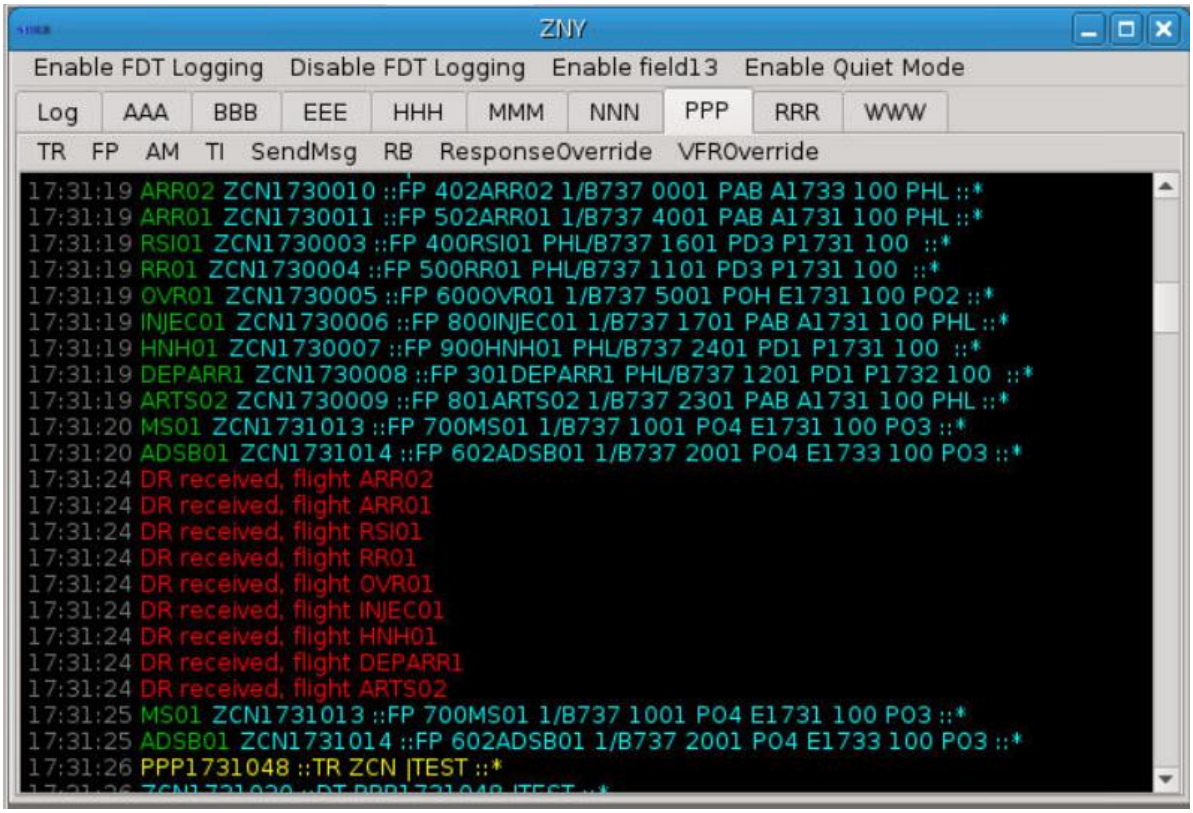

Figure 21. Error Messages

5.2.3.1.2.Test Message

The TR button can be pressed to send a test message from the Local Host. In the figure below, the TR messages in blue text are outgoing from ZNY and the DT responses in yellow text are incoming from NNN. Receiving DT responses indicates that the interface between the facilities is configured correctly.

Figure 22. TR Message

5.2.3.1.3.Send Message

The Send Message button allows the user to inject messages while the scenario is running. This provides the ability to create different situations without modification of the scenario. From a facility tab within the Local Host tab, users can click the **SendMsg** button. The Send IF Message dialog is displayed with the destination pre-populated with the name of the facility of the current tab.

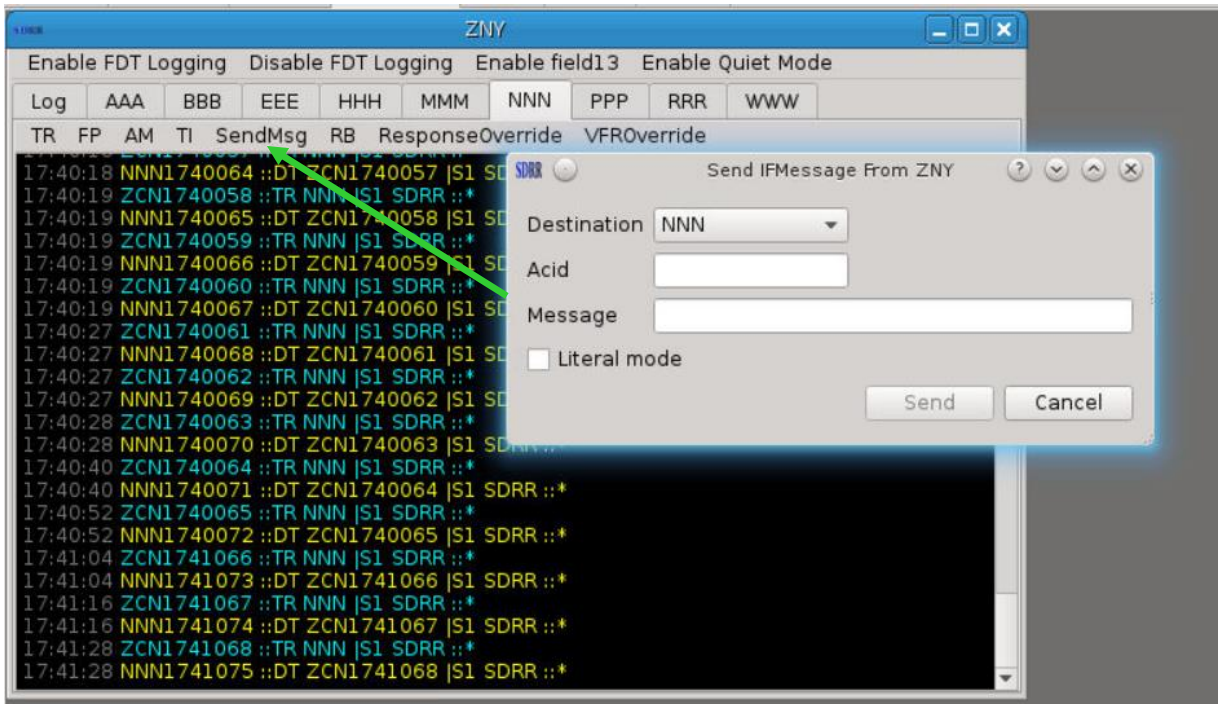

Figure 23. Send Interfacility Message

In the figure below, the **SendMsg** button is pressed in the NNN tab. The destination is set to NNN, indicating that the message is to be sent from the Local Host ZNY to NNN. Next, the aircraft identification of a scenario target needs to be entered for any flight related messages. Finally the desired message text can be entered and the Send button pressed.

In this example, a flight plan message is being sent from ZNY to NNN. After providing the aircraft ID, a flight message can be entered including substitution tokens. SDRR recognizes ### as a substitution token for the En Route computer ID (ECID) and will automatically assign a unique value. The @@@ is a substitution token for the terminal computer ID (TCID), which SDRR will also automatically assign for simulated terminal facilities. For a live STARS facility, the actual TCID will be inserted in the message upon injection. The user may also enter values for the ECID and TCID.

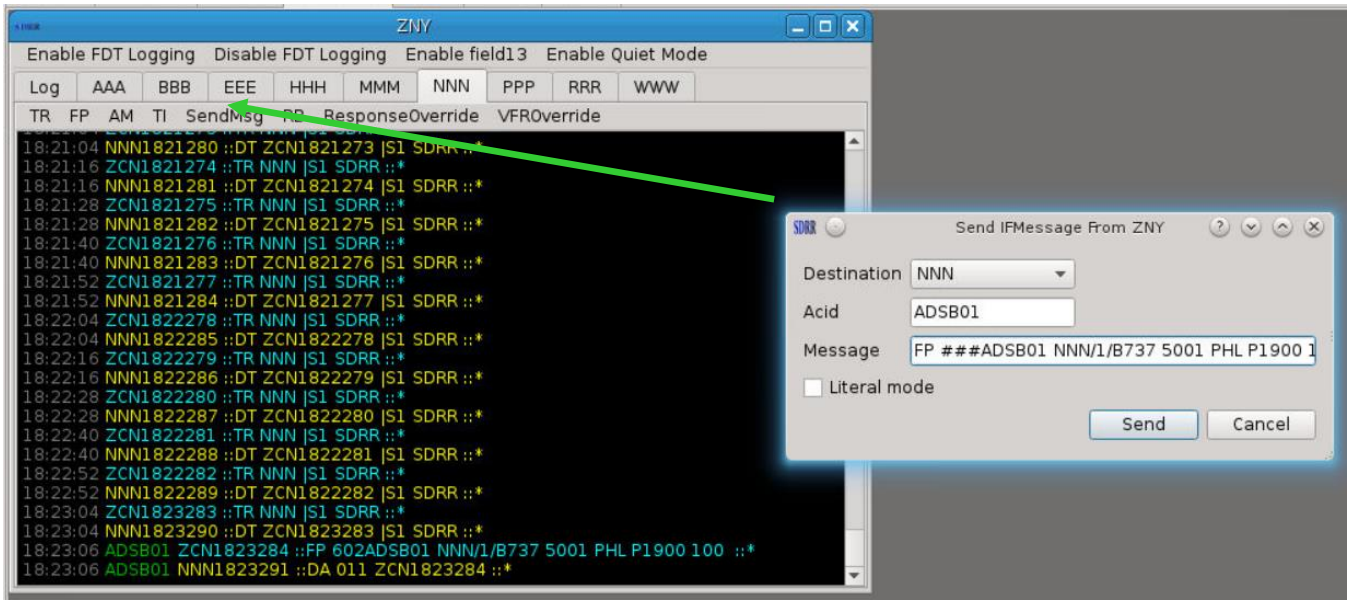

Figure 24. Send Message

SDRR also recognizes % at the beginning and end of the coordination time as an offset of the current time. For example, an arrival time of A%0010% is processed as current system time plus ten minutes.

NOTE: *The % coordination time substitution should only be used when the scenario start time is set to 00:00:00. For scenarios with start times other than 00:00:00, the exact coordination time (e.g., A1900) should be entered without the % substitution.*

The Literal mode check box allows symbols to be sent without any substitution. This may be helpful for sending custom error conditions and invalid characters.

5.2.3.1.4.Send Flight Plan

To create a new flight while running a scenario, users can input a terminal flight plan by clicking on the **FP** button. This button has a drop down menu allowing the user to select the type of flight plan to send – arrival, departure, or overflight. The figure below shows a dialog box for each flight plan type.

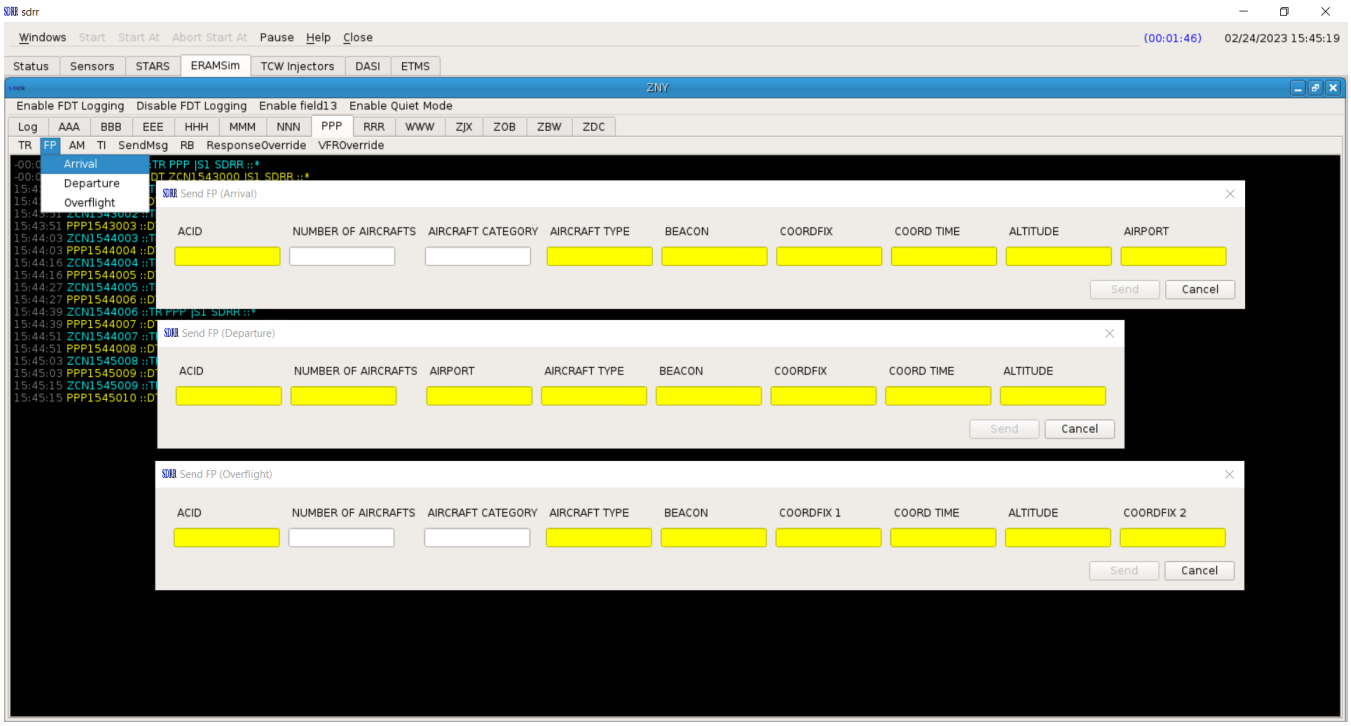

Figure 25. Send Flight Plan Types

The yellow textboxes are required fields. Once the fields are populated, the Send button becomes available. When the Send button is pressed, the flight plan is injected and added to the message log.

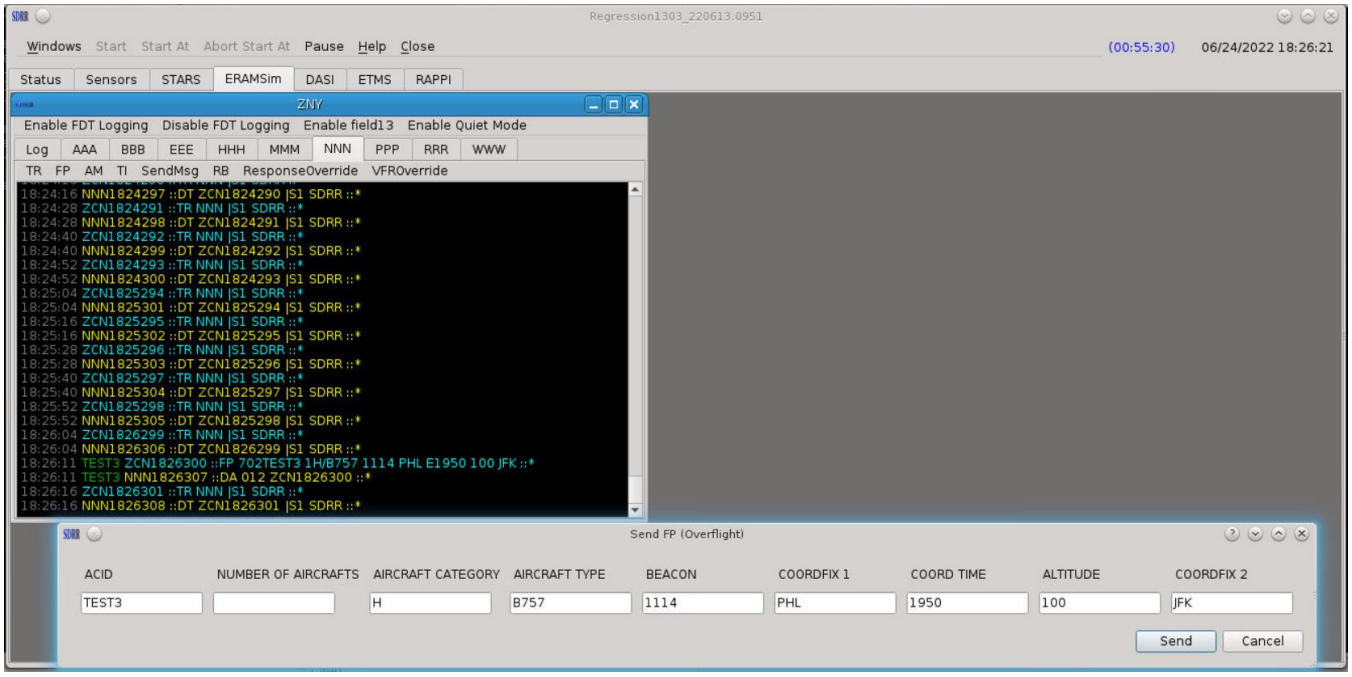

Figure 26. Send Flight Plan Example

5.2.3.1.5.Send Amendment

Flight plan amendments can be injected by clicking on the **AM** button. When the button is pressed a dialog box is displayed listing the flight plan fields that may be amended. Once the new values are entered in the flight plan fields to be amended and the Send button is pressed, an amendment message is injected and added to the message log. In the figure below, the beacon code in field four is amended from 1114 to 1111 for flight TEST3.

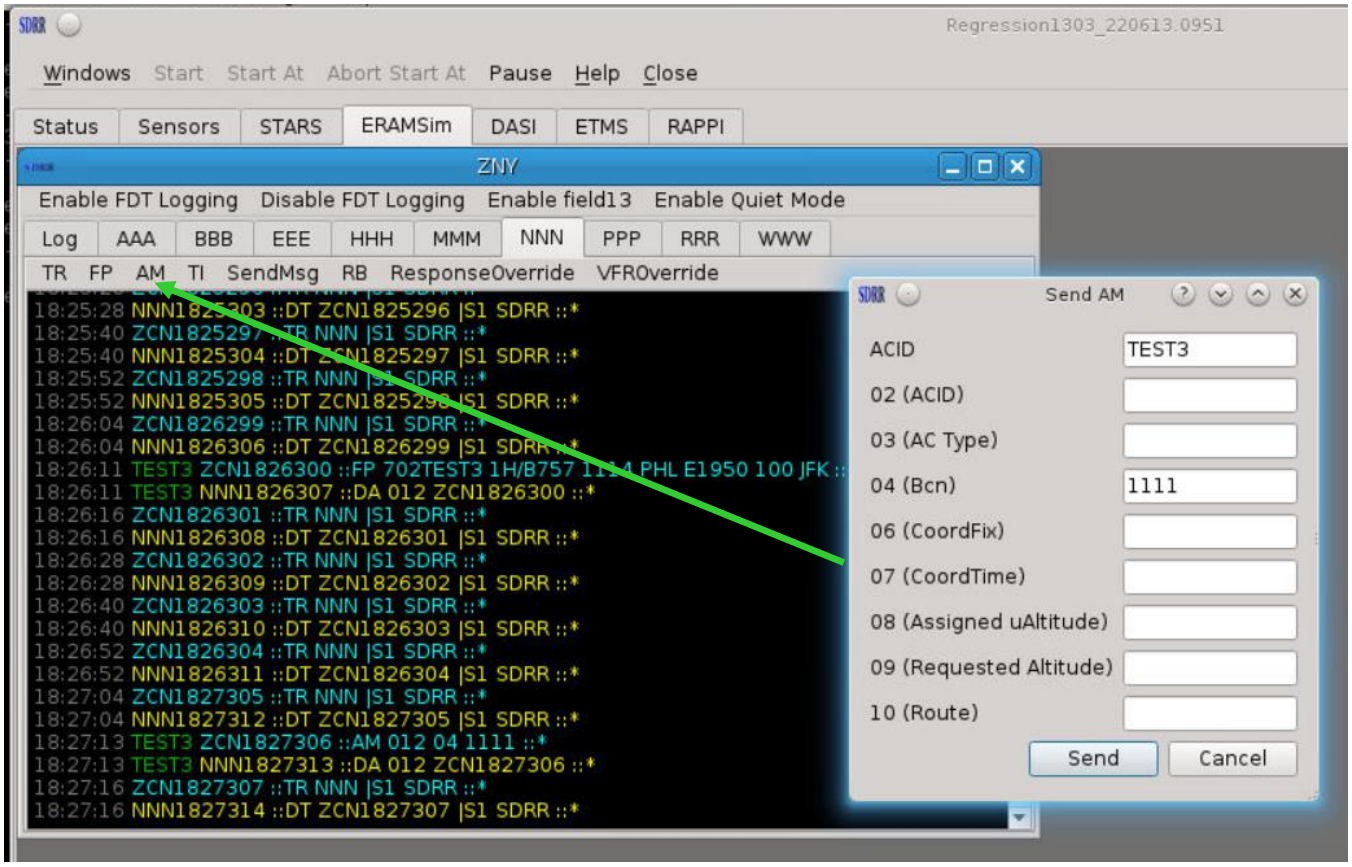

Figure 27. Send Amendment

5.2.3.1.6.Initiate Transfer of Control

To initiate a transfer of control for a flight, a TI message can be manually injected. When the **TI** button is pressed, a dialog is displayed with an ACID textbox. In the textbox, once the aircraft identification is entered the Send button becomes available. When the Send button is pressed, the TI message is injected and added to the messages log. The figure below shows a TI message for flight TEST3 sent from ZNY to NNN.

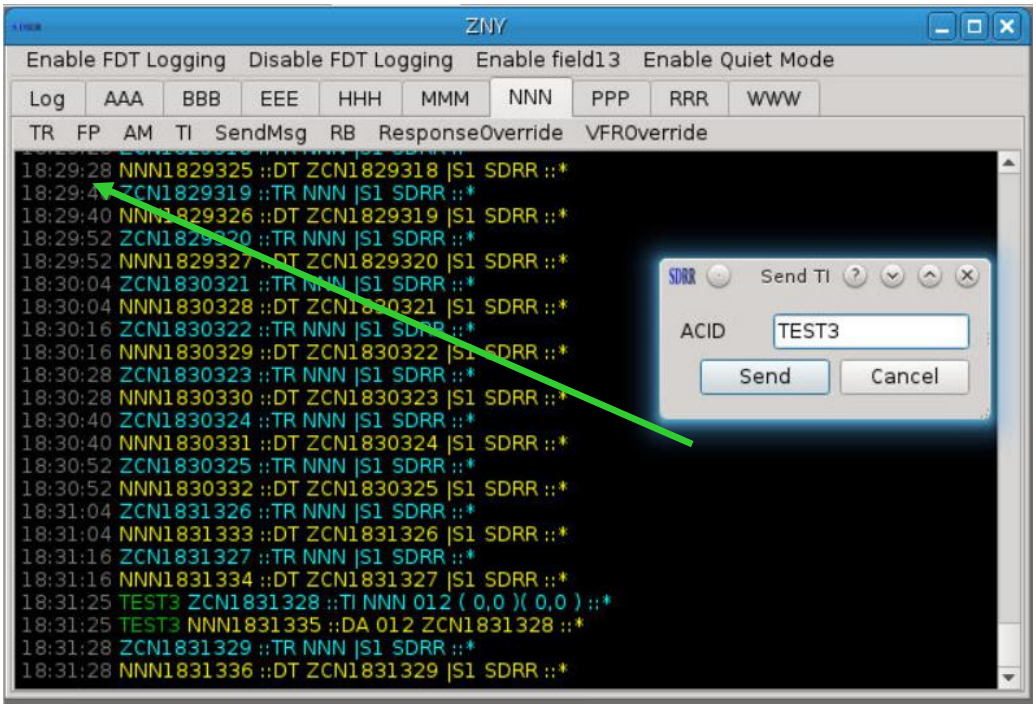

Figure 28. Send Transfer Initiate

5.2.3.1.7.Response Override

By default, SDRR sends acceptance responses from all simulated facilities. However, users can control how an individual facility responds to interfacility messages. The **ResponseOverride** button allows the user to override how the selected facility (in the current tab) will respond to messages for an individual flight. The STARS facility can be set to respond with DA, DX, DR, or NONE for a specific aircraft. Changes in response control are indicated in the message log by green status messages.

NOTE: *The response override will only affect the next response. After the selected response is sent one time, the responses will return to normal, default processing.*

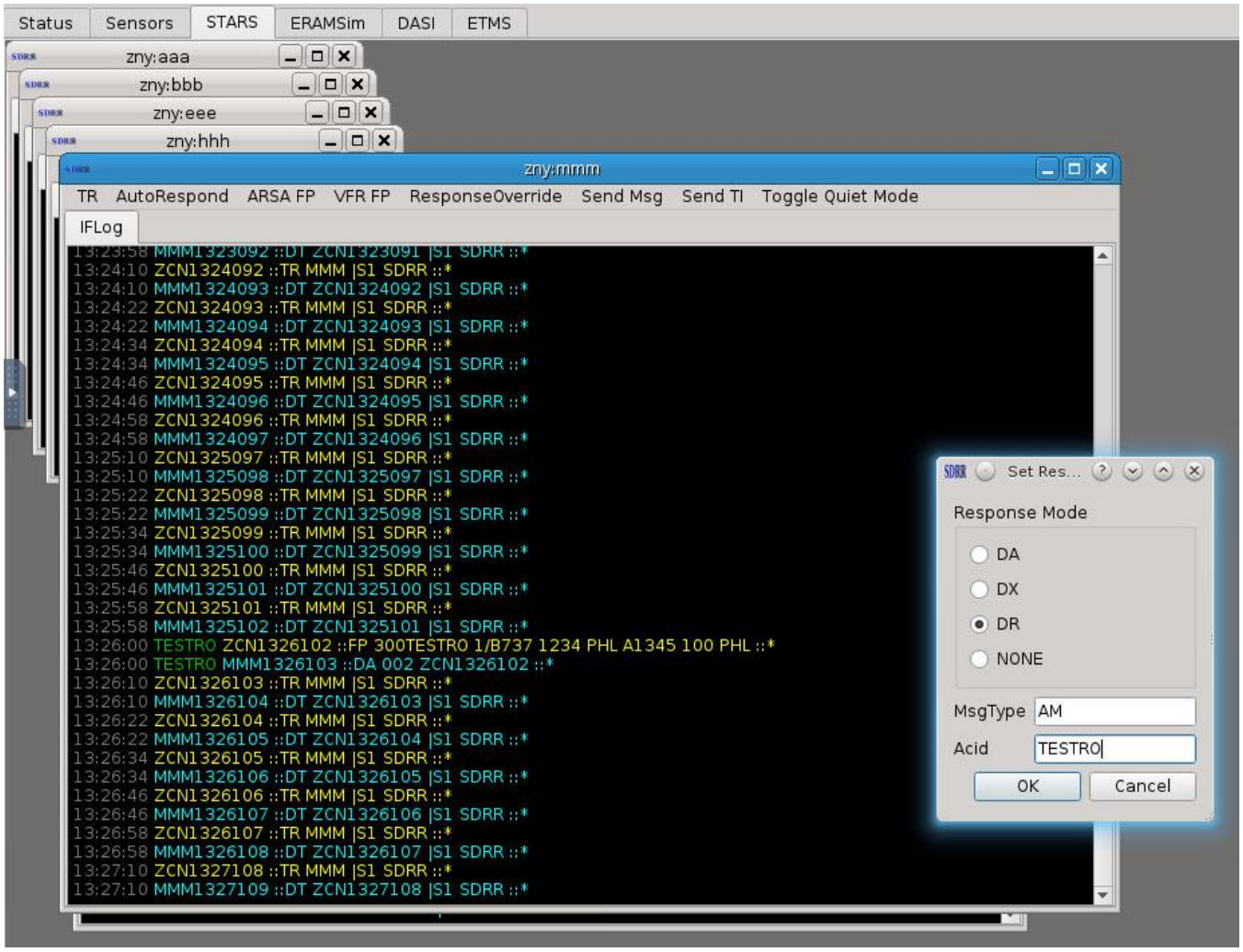

Figure 29. Response Override

In the figure below, MMM is set to respond with a DR to the next message from ZNY for flight TESTRO.

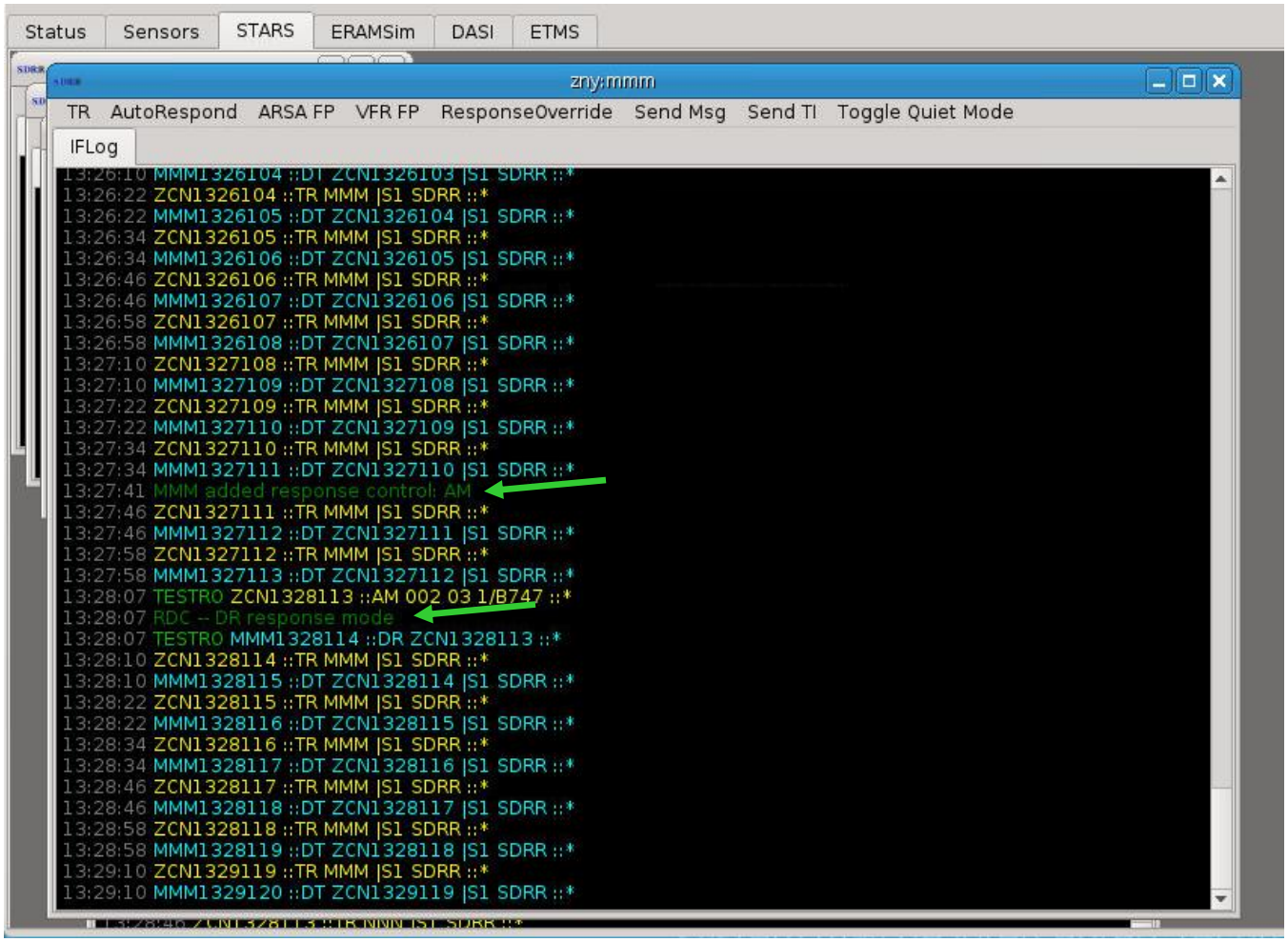

Figure 30. Response Override for TESTRO in STARS Tab

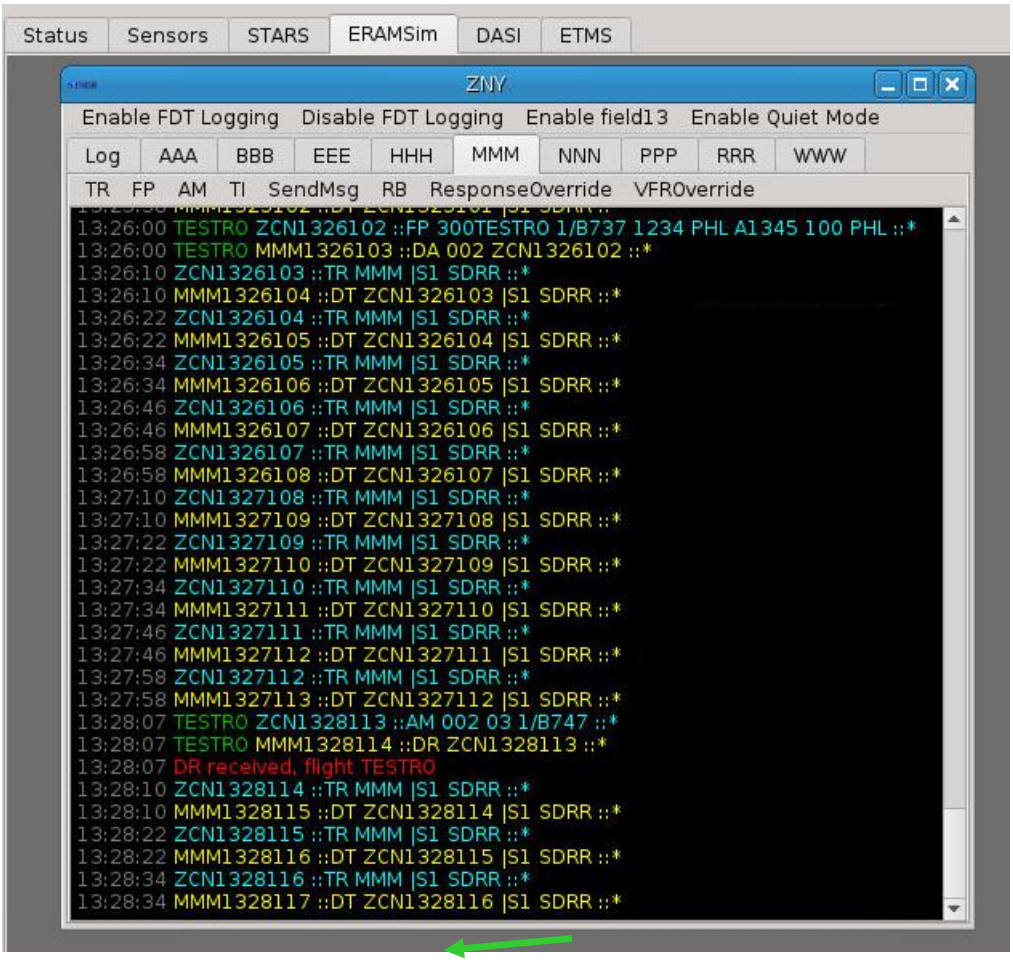

Figure 31. DR from MMM for Flight TESTRO

NOTE: *For a live facility, ResponseOverride can be set to override responses sent from TCW Controller units.*

5.2.3.2. EDDS Tab

If the simulated ERAM facility stanza in the SDRR configuration file includes an external eddserver, then the ERAM facility window will also include a tab labeled EDDS. The EDDS tab shows messages sent to and received from a connected EDDS.

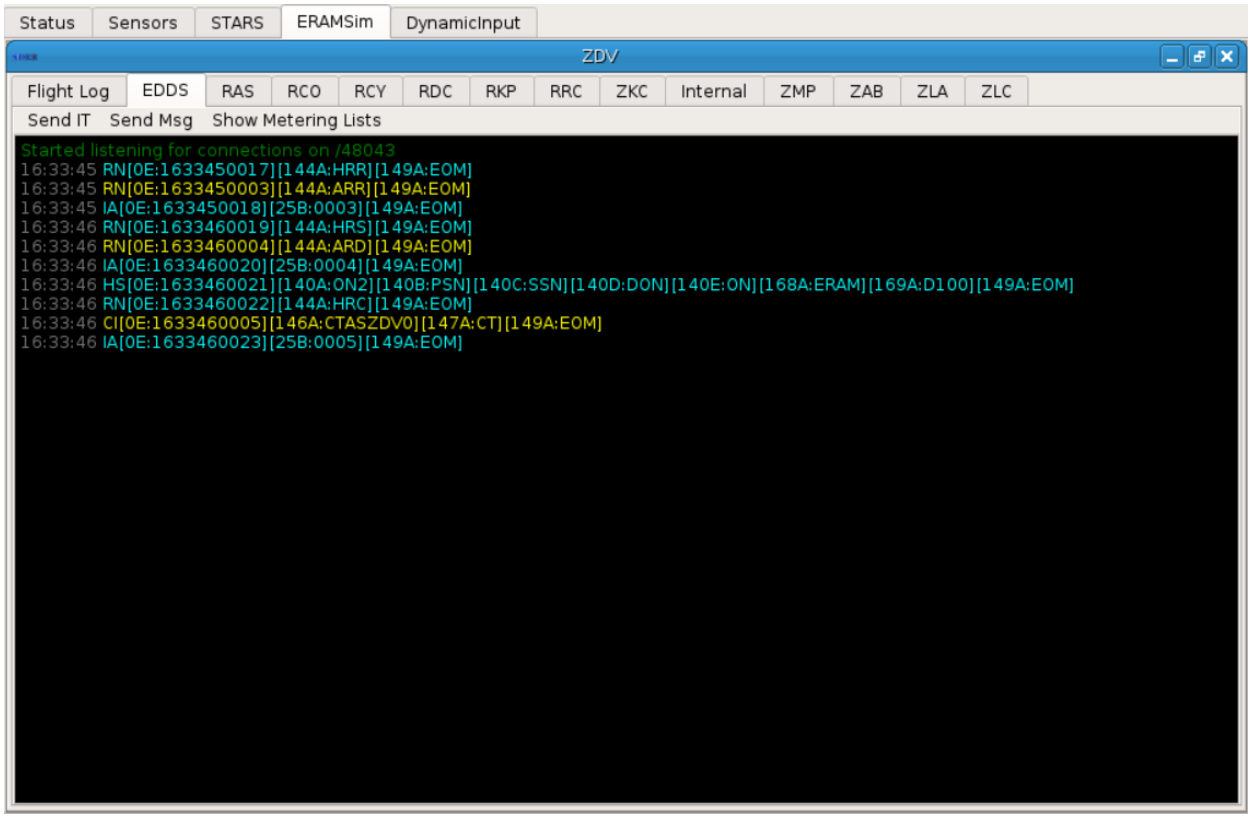

Figure 32. ERAMSim Tab

The EDDS Tab of each simulated ERAM facility window has the following buttons:

Send IT

Clicking on this button causes an IT message to be sent to EDDS.

Send Msg

This button launches a dialog box where any freeform CMS message can be entered and sent to EDDS.

Show Metering Lists

This button displays the Meter Entry Viewer including meter fixes, aircraft IDs, meter times, delays and

speed advisories sent by TBFM. Note that this Viewer is not updated dynamically; it must be closed and re-opened to view the most current entries.

5.2.4. STARS Tab

The STARS Tab is displayed if SDRR is configured to simulate one or more Terminal facilities and shows messages sent to and received from the host En Route facility. A window is displayed for each STARS facility included in the SDRR configuration file. Each STARS facility window includes the following buttons: TR, AutoRespond, ARSA FP, VFR FP, ResponseOverride, Send Msg, Send TI, Toggle Quiet Mode.

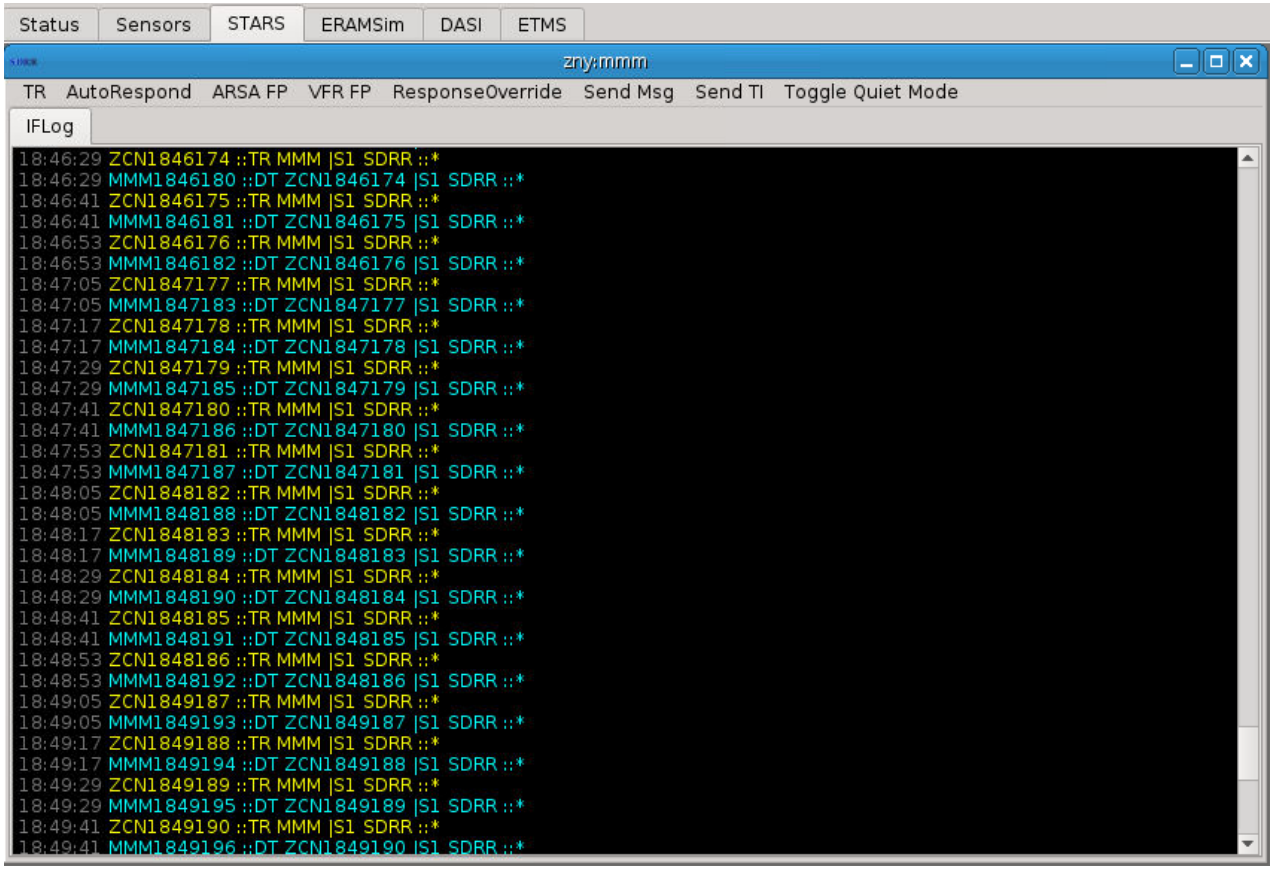

Figure 33. STARS Tab

5.2.5. TCW Injectors Tab

The TCW Injectors tab is displayed if the TCW Injector element is present in the configuration file. This allows SDRR to send scripted or injected TCW messages from the TCW Injector tab.

Example cfg.xml file:

```
<tcwInjector fac="ppp" user="ATBTEAM">
        <position name="*">tcp://${TCW_URL}</position>
</tcwInjector>
```

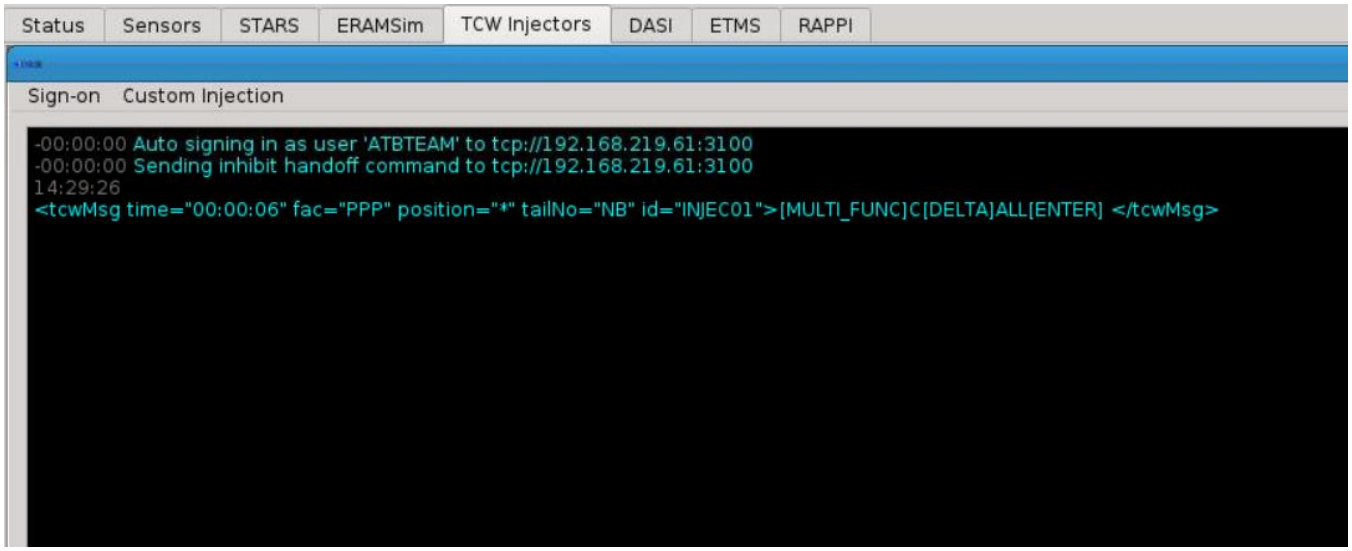

Figure 34. TCW Injectors Tab

Sign-on

Clicking on this button will auto sign-on as the user specified in the configuration file. A message will appear to indicate which user was signed in on the TCW machine, e.g. ATBTEAM.

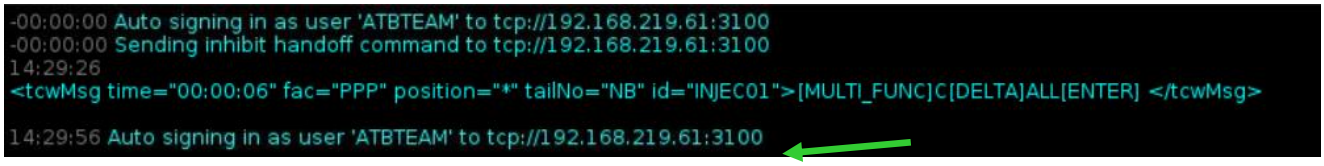

Figure 35. TCW Sign-on

Custom Injection

This button displays a Command dialog box where the user can inject a TCW message.

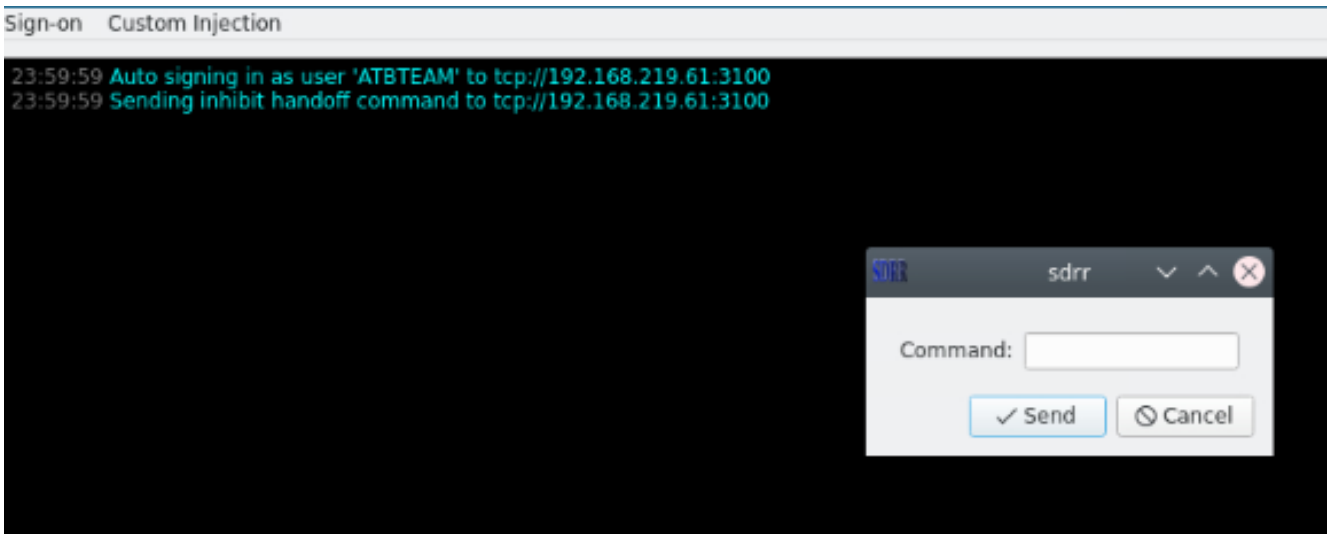

Figure 36. Custom TCW Injection

5.2.6. ETMS Tab

The ETMS tab is displayed if an ETMS data source is defined in the SDRR configuration file. This allows SDRR to provide an ETMS link to a live Terminal string via a DB9 serial port. Like the interfacility data, SDRR sends blue TR / DT messages and receives yellow TR / DT messages from the Terminal.

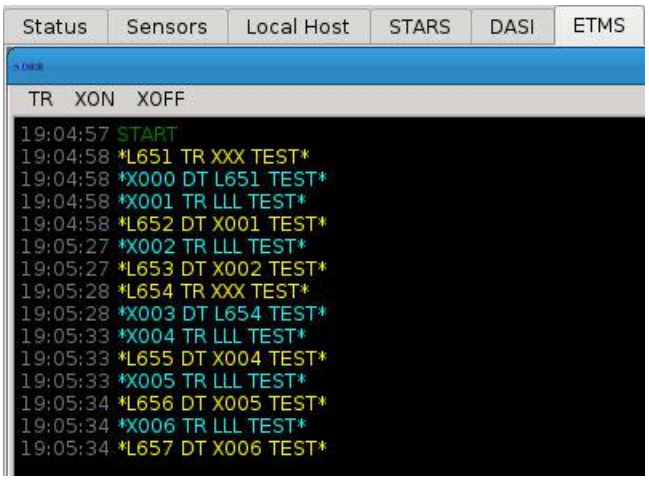

Figure 37. ETMS Tab

5.2.7. DASI Tab

The DASI tab is displayed if a DASI device is defined in the SDRR configuration file. Within the DASI tab, a window is displayed for each configured device. Buttons are available in each window to change DASI settings such as the update period and DASI value and to send generic messages. The SDRR configuration file contains the default values in DASI setting. Changes to the DASI values can also be a part of the SDRR scenario, from CDR extraction or recording, and can be injected into the terminal system to recreate recorded data.

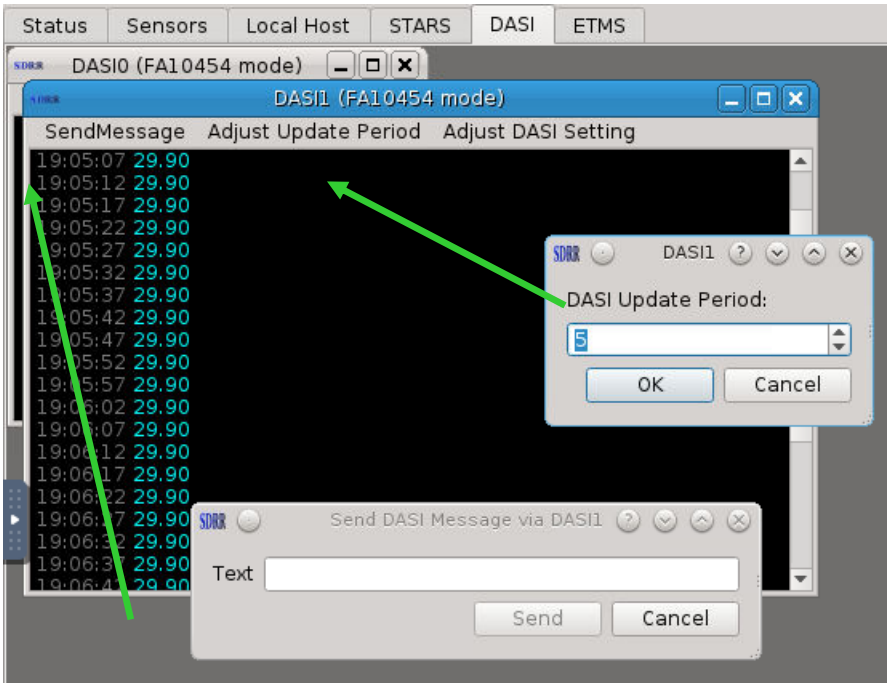

Figure 38. DASI Tab (SendMessage and Adjust Update Period)

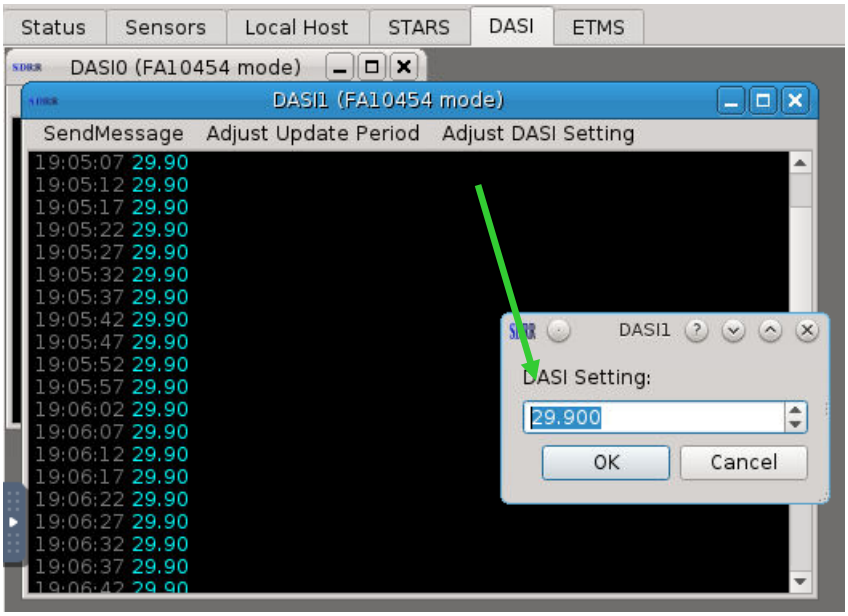

Figure 39. DASI Tab (Adjust DASI Setting)

5.2.8. RAPPI Tab

The RAPPI tab is displayed if the SDRR configuration file includes surveillance devices. Within the RAPPI tab, an individual tab will be available for each radar and service volume found in the configuration file. These tabs give a visual representation of the targets that are sent from each surveillance source.

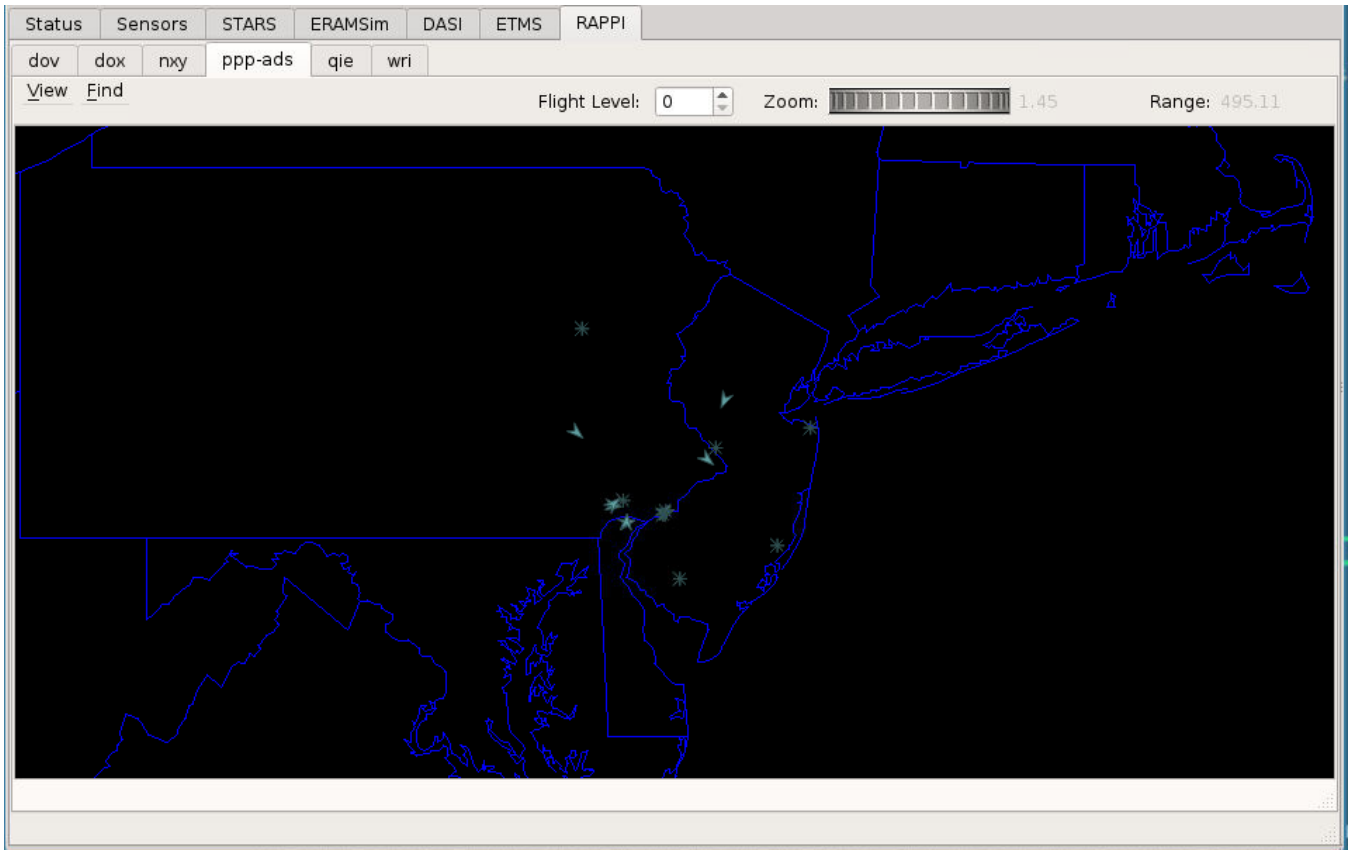

Figure 40. RAPPI Tab

The tab for each surveillance source includes a map display and a menu/tool bar which includes:

View

The View menu is used to select maps for display and to set options for radar sources.

Find

The Find menu is used to search for a specific target by beacon code or ICAO address. Enter the three letter radar name followed by the beacon code or ICAO address and click on “OK”. If the track exists, a dialog box will appear with real-time track information.

Flight Level

The Flight Level tool allows an altitude to be set either by typing the value into the box or by clicking the up and down arrows. Changing the altitude also changes the surveillance coverage area.

Zoom

The Zoom tool allows the range of the display to be adjusted using a wheel selector.

Range

The Range tool displays the horizontal size (in nmi) of the airspace showing in the map display.

The figure below shows the Find dialog along with the search results for a target with beacon code 5001 in the QIE radar. The search results consist of a target information box containing radar details. This target information box can also be displayed by right clicking on a track in the RAPPI map display. Once the information box is displayed, right clicking the track again toggles the box off.

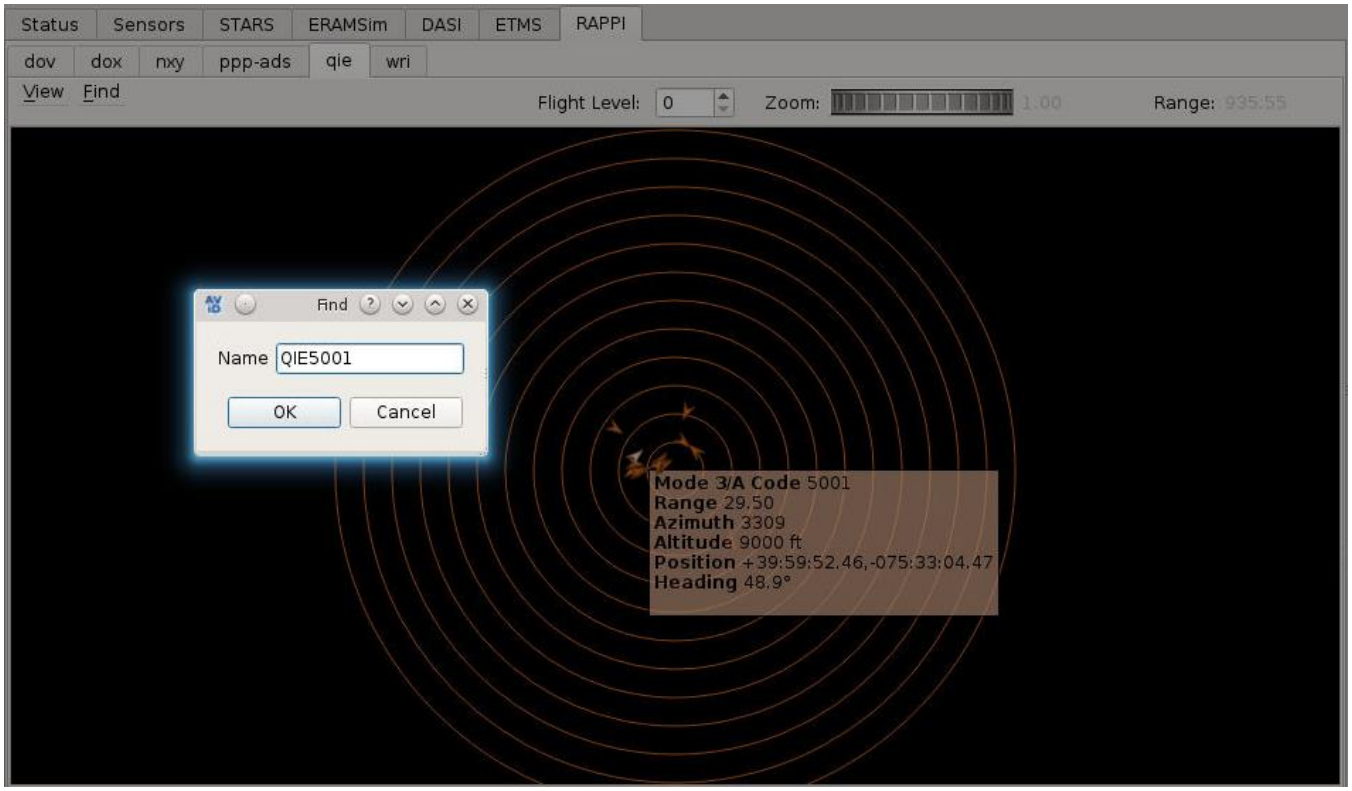

Figure 41. Target Details

Clicking on the **View** button and selecting the **Sources** option displays the radars found in the SDRR configuration file. For each radar source, options are available to control the data that are displayed. Text that is grayed out indicates options that are listed for information only and cannot be modified. The options in black text allow users to modify the presentation of the data from the radar. Checking an option enables the display of the data; un-checking disables the display. In the figure below, the weather (wx) option for radar PHL is checked and weather data are added to the RAPPI map display. The correct radar tab must be selected to see the effect of the change.

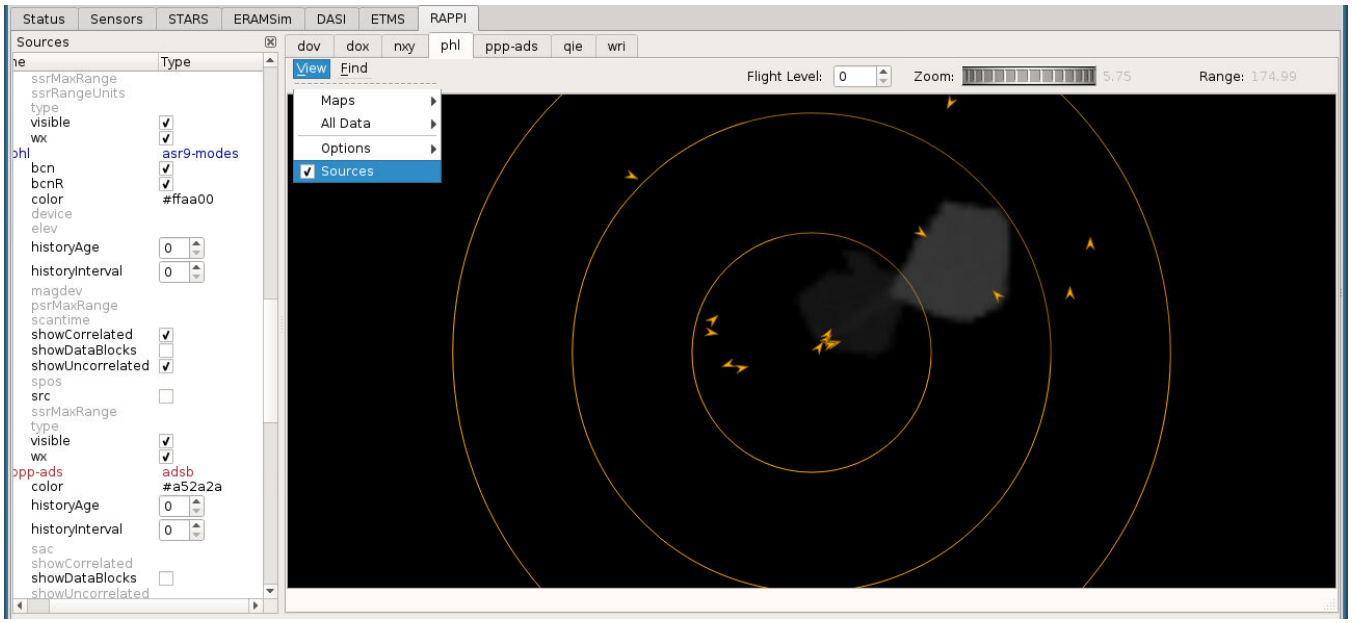

Figure 42. Sources Option

The RAPPI tab is a function of the Airspace Visualization Display (AViD) software and can be used for radar recording, displaying and analyzing data. For more information on AViD, please refer to the AViD user manual.

5.2.9. CCU Tab

The CCU tab is displayed if the SDRR configuration file includes CCU devices. A window is displayed for each configured CCU and shows a log of all messages exchanged with that device. Each window also includes menu buttons **SendMessage** and **QuietMode**.

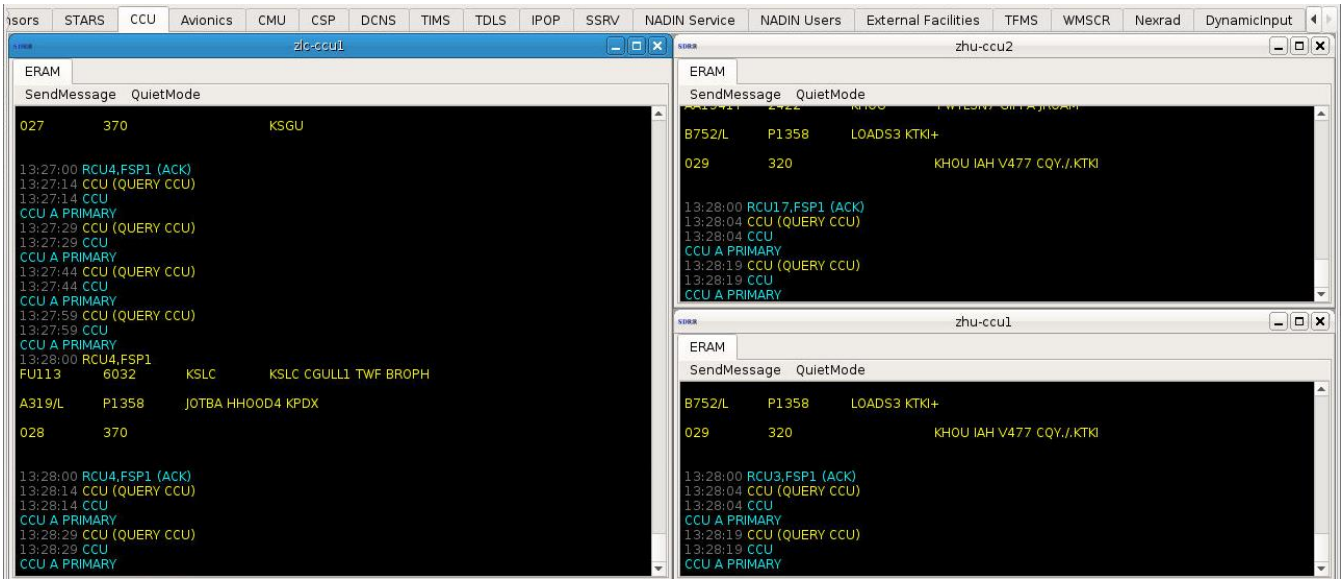

Figure 43. CCU Tab

When the **SendMessage** button is clicked, a dialog is opened allowing users to enter a specific device and message to send. The **QuietMode** button disables display of health check messages.

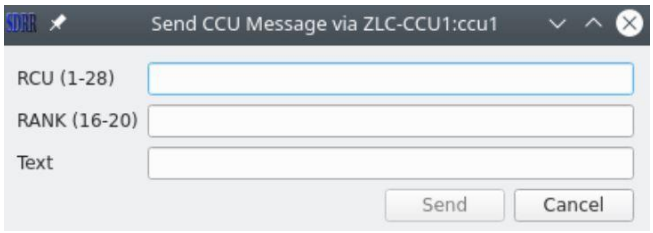

Figure 44. Send CCU Message

5.2.10. Avionics Tab

The Avionics tab displays logon messages for a flight. Green text indicates a successful logon, while red text indicates that the logon has failed.

Figure 45. Avionics Tab

The Avionics tab includes buttons for **AFNLogon** and **SendMessage**. The **AFNLogon** button allows users to manually send an AFN logon if the flight is not scripted to log on automatically. The **SendMessage** button allows users to manually send any CPDLC downlink message.

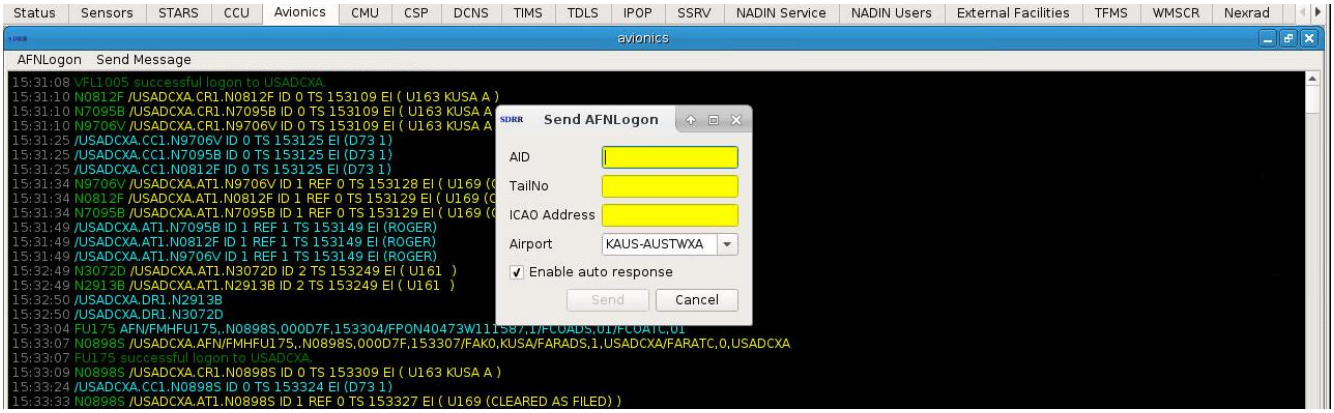

Figure 46. Send AFN Logon

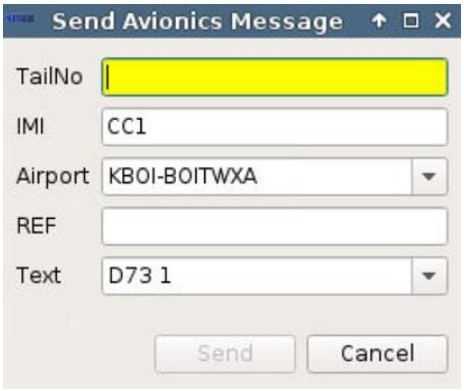

Figure 47. Send Avionics Message

5.2.11. CMU Tab

The CMU tab includes sub tabs for Avionics and CSP. The Avionics tab displays messages that include AFN messages and CPDLC messages. The CSP tab displays CSP messages.

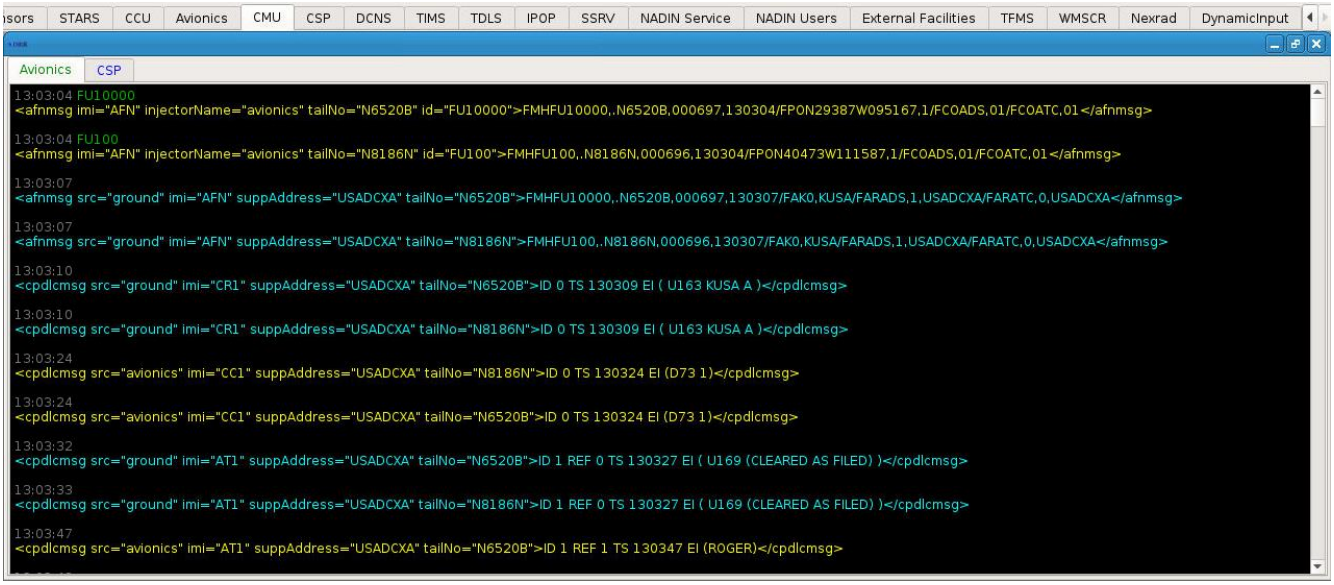

Figure 48. CMU Tab - Avionics

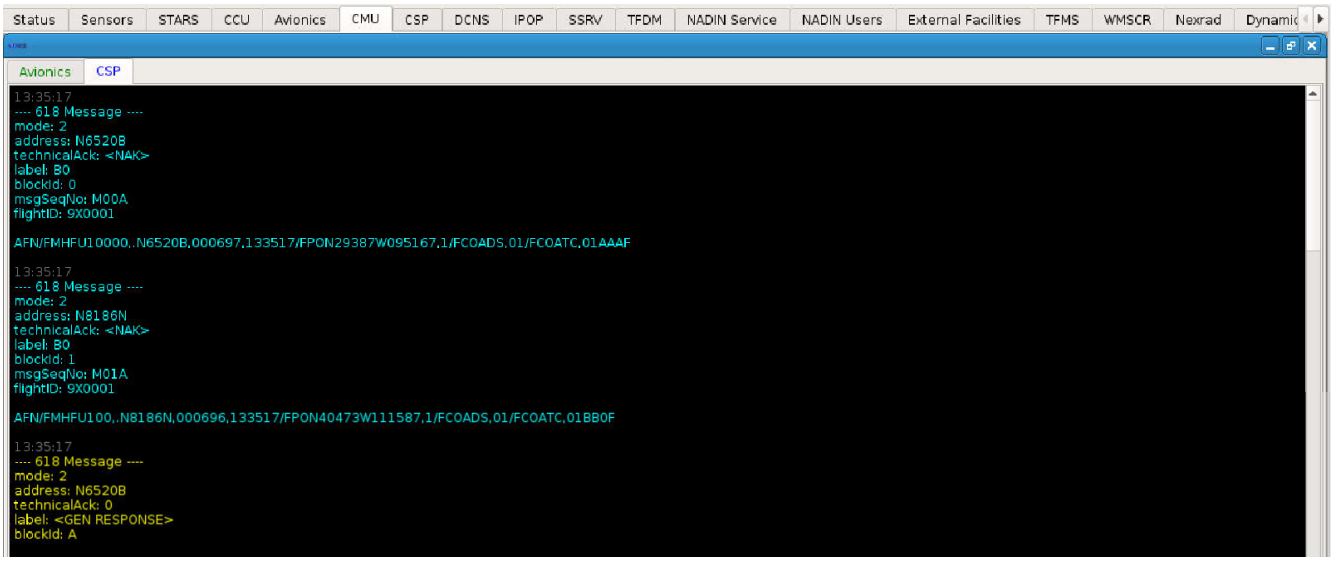

Figure 49. CMU Tab - CSP

5.2.12. CSP Tab

The CSP tab displays SDR messages and DCNS messages. The user has the ability to send MAS Response Override messages. These messages will override the message assurance (MAS) responses sent for received uplinks. These messages can include the following error codes:

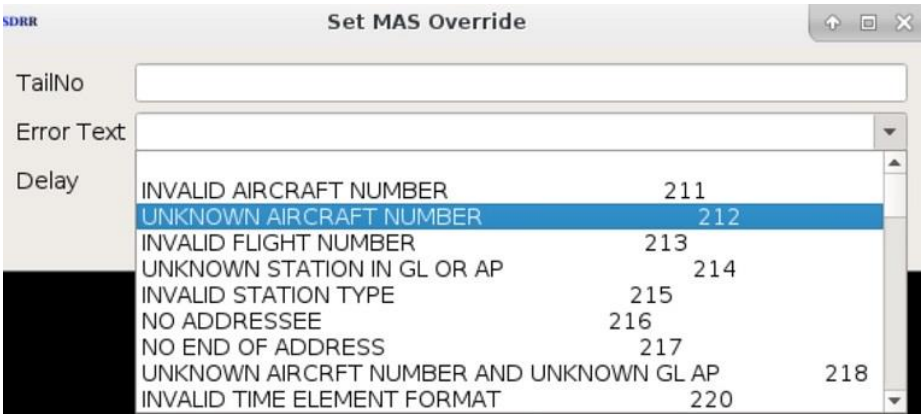

Figure 50. CSP Error Codes

To stop the MAS response overrides, click the **Clear MAS Response Override** button.

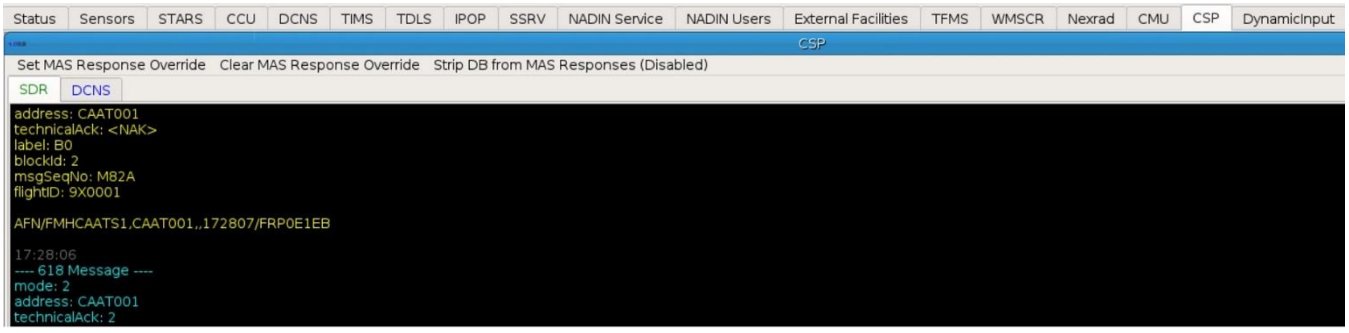

Figure 51. CSP Tab - SDR

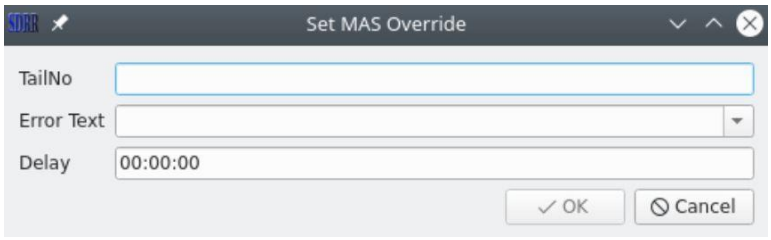

Figure 52. CSP Tab - DCNS with MAS Override Dialog

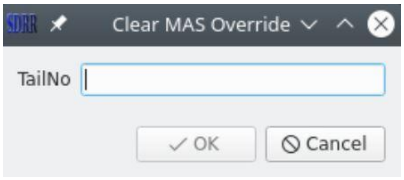

Figure 53. CSP Tab - DCNS with Clear MAS Override Dialog

Strip DB from MAS Responses (Disabled/Enabled) - The MAS response includes a copy of the uplink it's in reference too. Strip DB removes that field from the copy of the uplink.

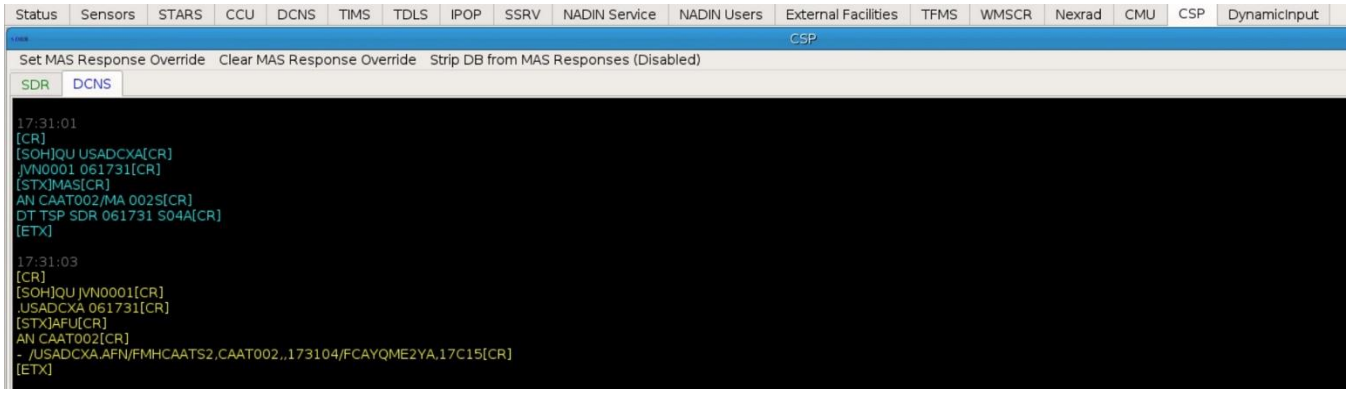

Figure 54. CSP Tab – Strip DB from MAS Responses (Disabled/Enabled)

5.2.13. DCNS Tab

The DCNS tab displays DCNS messages. Cyan color is the messages that SDRR sends and yellow is the response from ERAM.

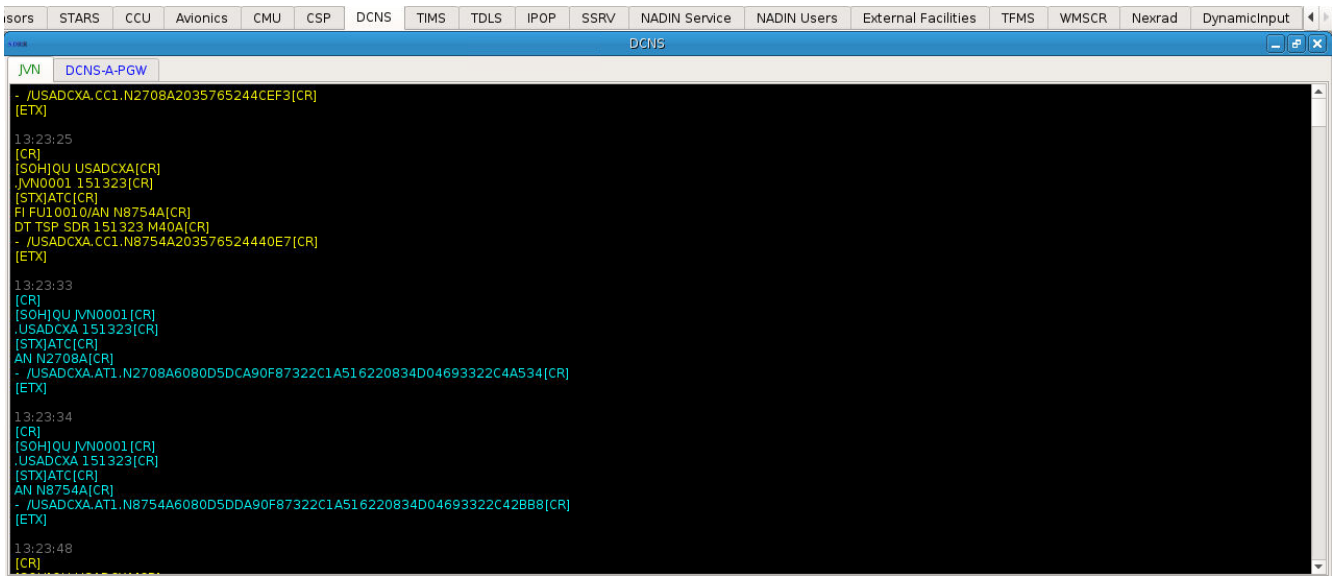

Figure 55. DCNS Tab - JVN

Figure 56. DCNS Tab - DCNS-A-PGW

5.2.14. TIMS Tab

The TIMS tab displays information about the active TIMS: WST or EST. This tab also shows information about TDLS. This tab is for simulated TIMS/TDLS.

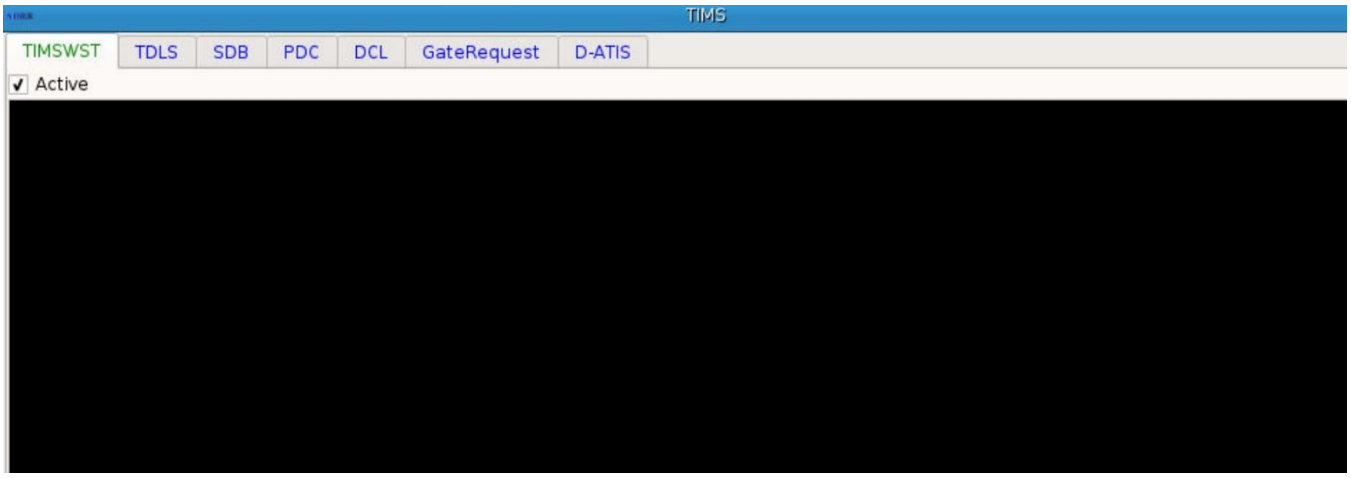

Figure 57. TIMS Tab - TIMSWST

Figure 58. TIMS Tab - TDLS

5.2.15. TDLS Tab

The TDLS tab displays a separate window for each simulated TDLS included in the configuration file. Each window includes an ERAM tab and a CPDLC tab. The ERAM tab displays a message log for messages exchanged between ERAM and the simulated TDLS. The CPDLC tab displays a message log for messages exchanged between CPDLC and the simulated TDLS.

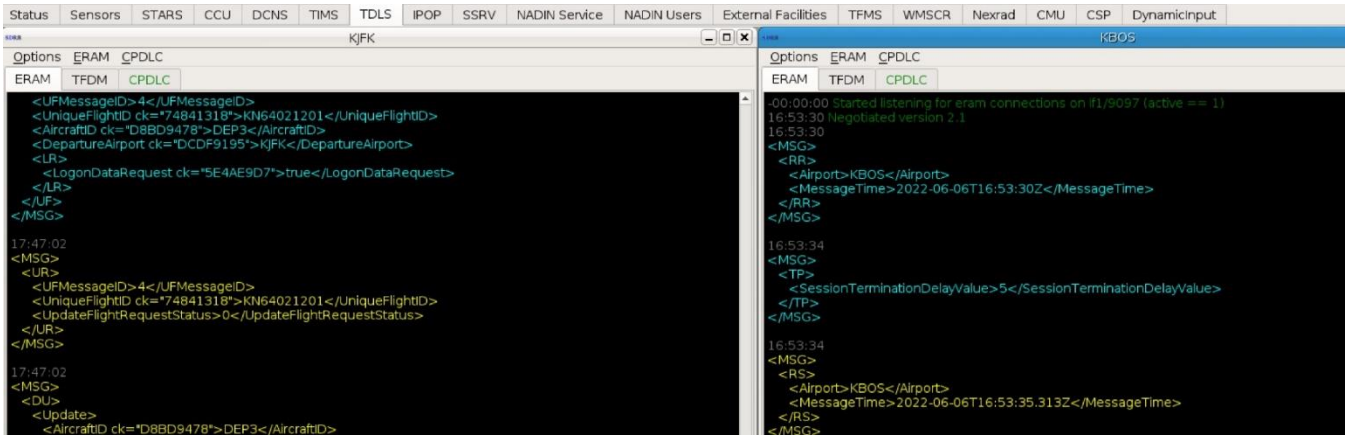

Figure 59. TDLS Tab - ERAM

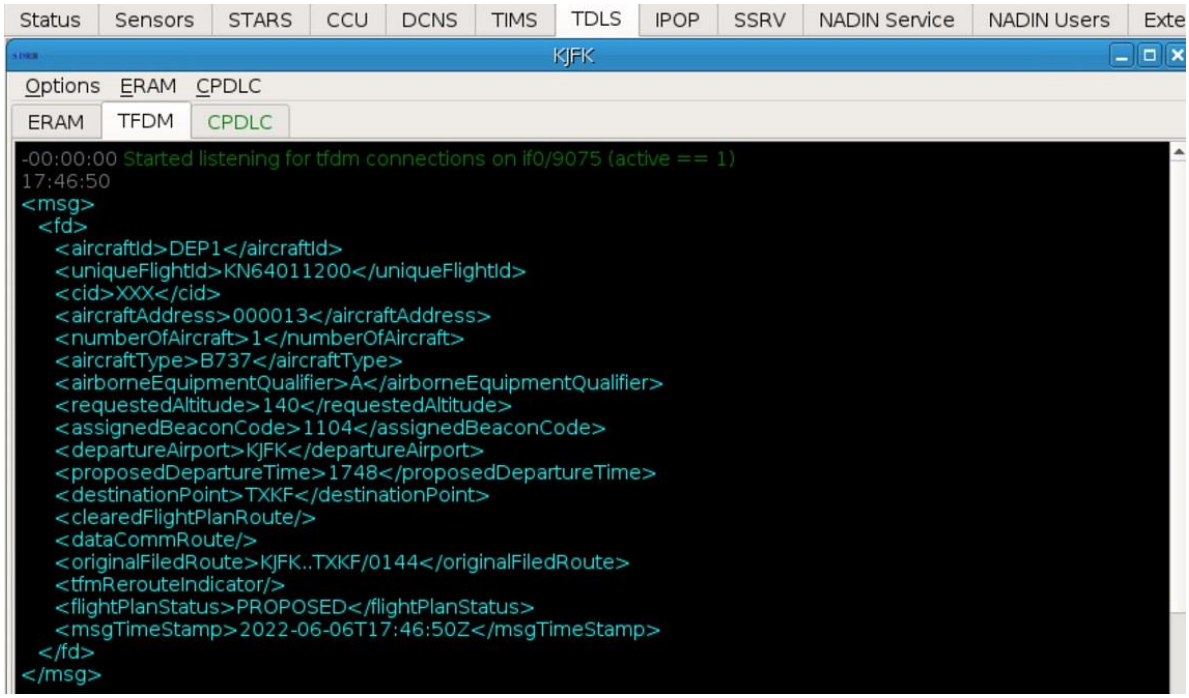

Figure 60. TDLS Tab - TFDM

Figure 61. TDLS Tab - CPDLC

Each TDLS window also includes a menu bar with the following items: **Options**, **ERAM**, and **CPDLC**. The **Options** menu item allows users to select the Autoresponse Mode option and Require MAS Response. When the Autoresponse Mode option is checked, the simulated TDLS automatically processes and responds to CPDLC messages. When Require MAS Response is checked it will require a MAS response from TDLS uplinks.

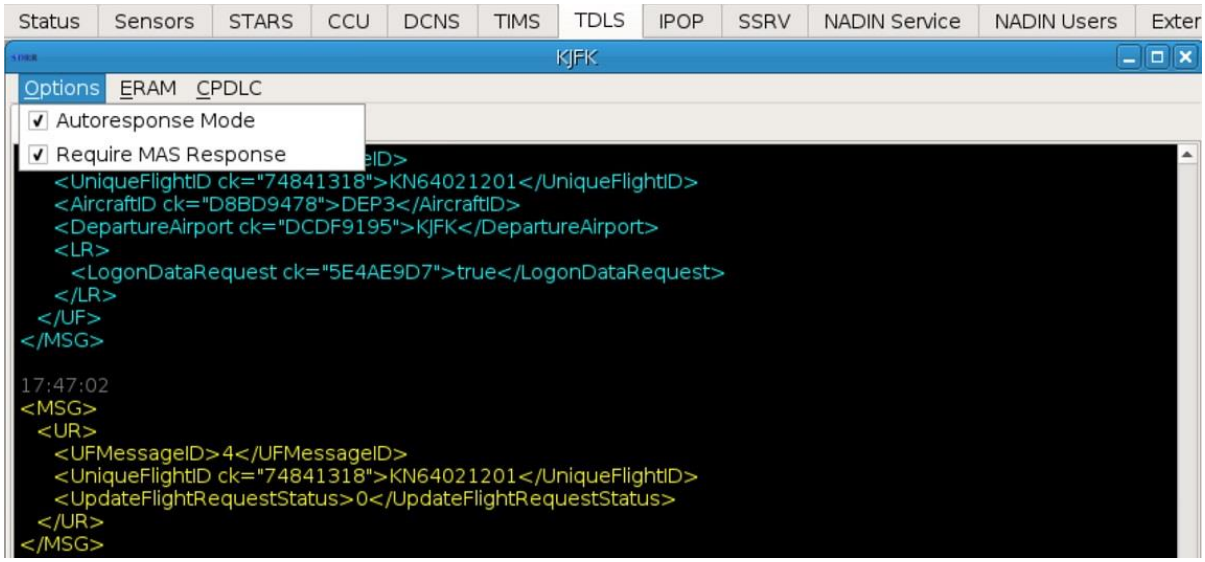

Figure 62. Autoresponse Mode & Require MAS Response Options

The **ERAM** menu item lists the following selections: **Request Reconstitution**, **Send Logon Request**, **Send Session Update**, **Send Clearance Delivered**, and **Disconnect**. Selecting **Request Reconstitution** reconstitutes the TEDC connection. The **Send Logon Request** selection sends a UF-LR message for the tower to log on. The **Send Session Update** selection sends a CC1 message to establish the connection. The **Send Clearance Delivered** selection sends the tower clearance. When the **Disconnect** option is checked, the simulated TDLS disconnects from ERAM.

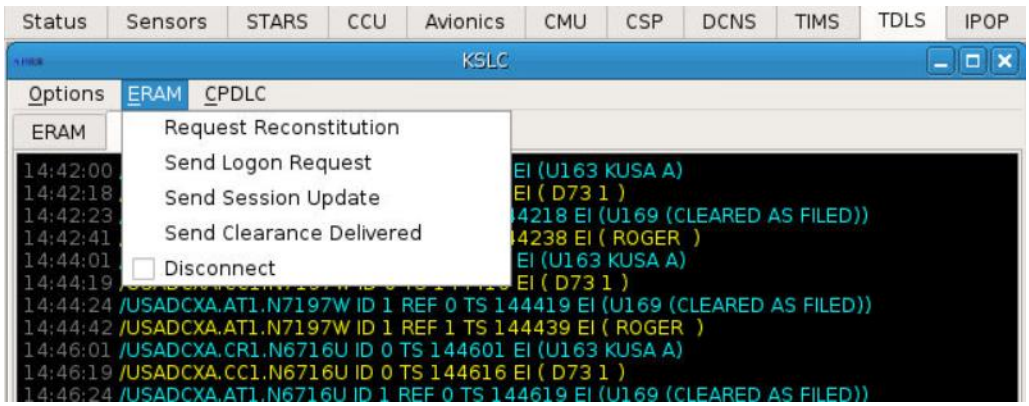

Figure 63. ERAM Session Selections

The CPDLC menu item lists the following selections: **Connection Request**, **Disconnect**, **CAF**, and **Generic Message**. Selecting the **Connection Request** sends the CR1 messages and waits for CC1 reply. Selecting **Disconnect** sends a DR1 message that makes the flight disconnect the CPDLC session. The

CAF selection is a Cleared as Filed clearance. Selecting **Generic Message** allows the user to send any AT1 message.

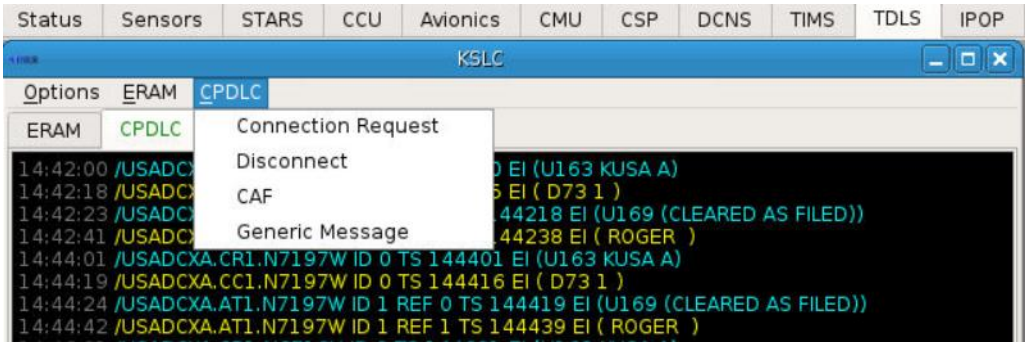

Figure 64. CPDLC Session Messages

5.2.16. IPOP Tab

The IPOP tab shows the logs for each local facility. There is an option to send CMS messages and to turn quiet mode on. Quiet mode stops displaying the IPOP heartbeat messages.

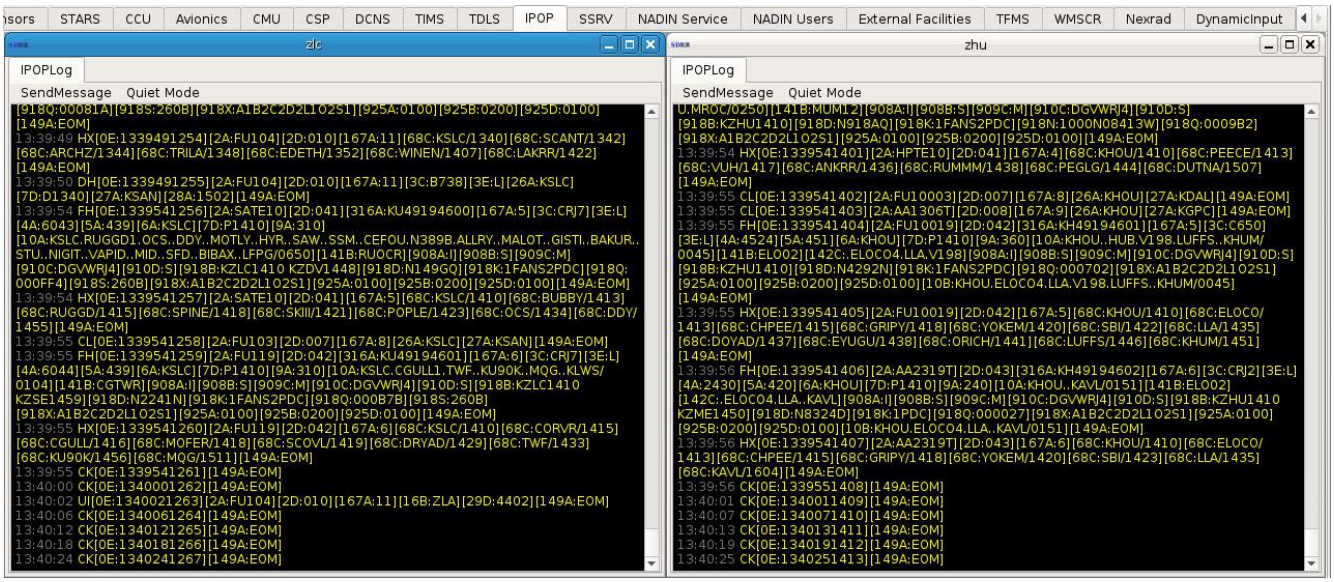

Figure 65. IPOP Tab

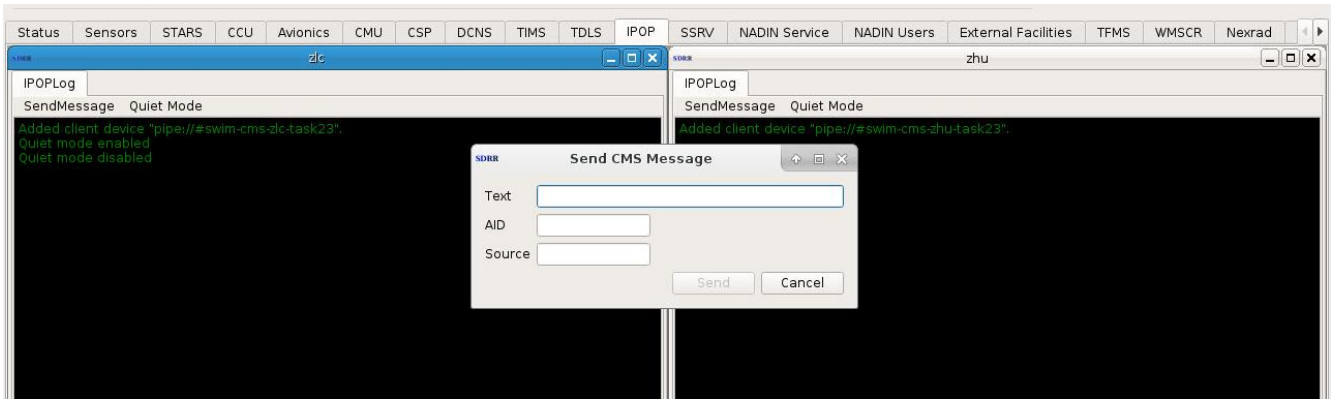

Figure 66. Send CMS Message

5.2.17. SSRV Tab

The SSRV tab displays the PSIM and SSIM status messages and ERAM console messages. On the lower right-hand corner of each En Route facility channel, there is an indicator to show the PSIM/SSIM status. Once the PSIM/SSIM status is green, the scenario can be started.

- Red indicates that the scenario needs a PSIM
- Yellow indicates that the PSIM was successful
- Green indicates the channel is ready for scenario to be started

Channel A/B displays can be checked to be the active channel and the flight information is sent through the active channel(s). Cyan messages are messages injected via the scenario, whereas yellow is the response SDRR receives from ERAM in regards to those messages.

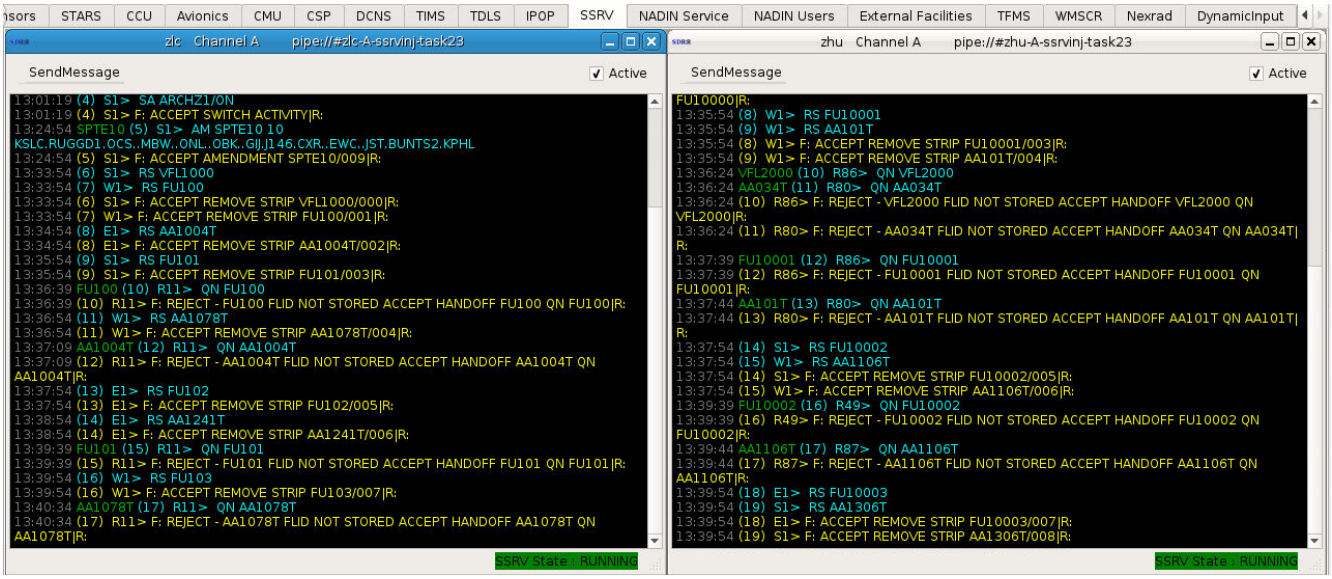

Figure 67. SSRV Tab

If a message needs to be manually sent, it can be sent through the **SendMessage** button on the SSRV tab. The channel display will indicate if the message was accepted or rejected.

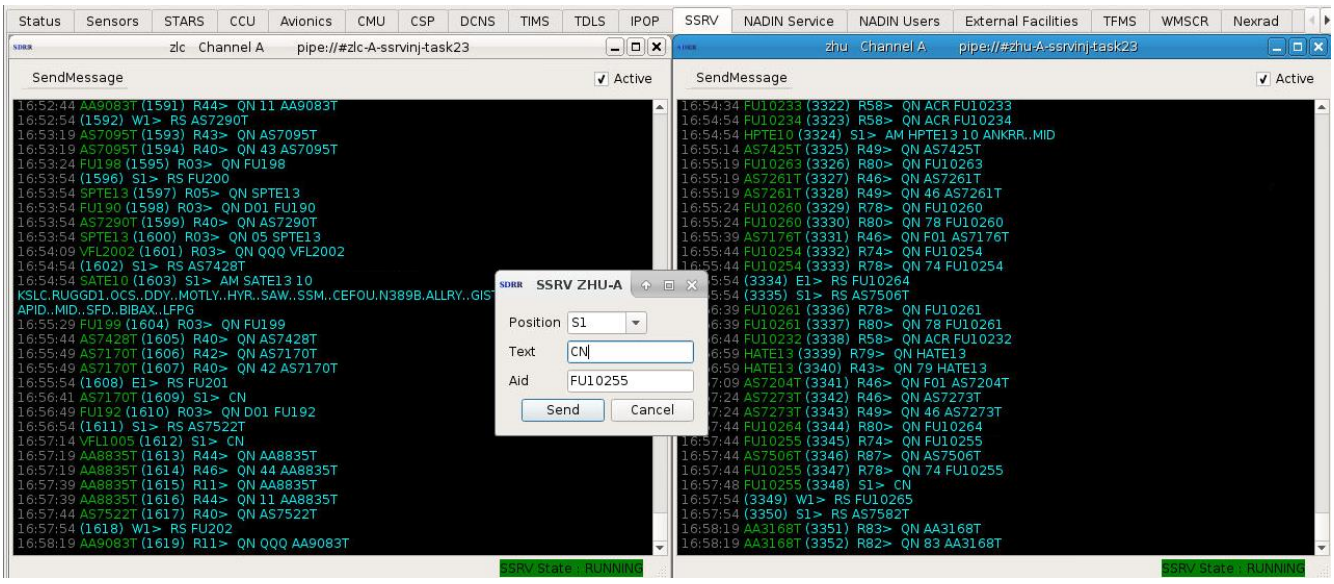

Figure 68. Send ERAM Message

5.2.18. TFDM Tab – TDLS Connection

The TFDM tab shows all the messages TFDM receives from TDLS.

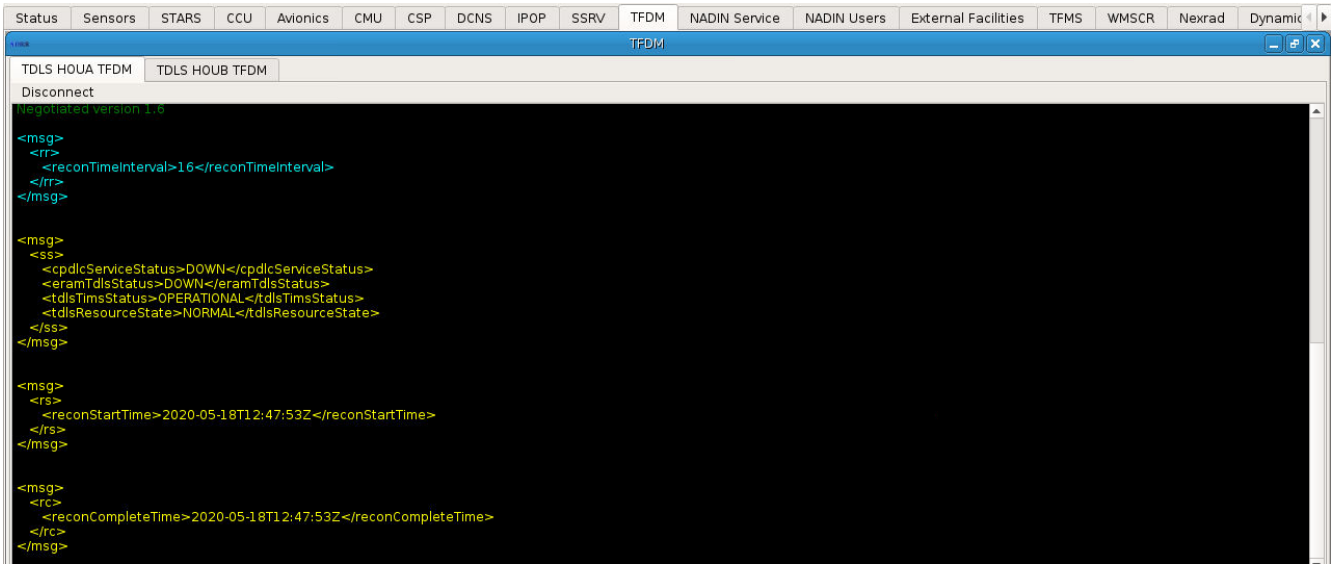

Figure 69. TFDM Tab - TDLS

5.2.19. TFDM Tab – TBFM Connection

TFDM can also be configured with TBFM connections for MIS, RTCS, and TTP. If configured, one MIS tab is displayed per TBFM/ARTCC. If configured, an RTCS tab is displayed for each RTCS airport included in the SDRR configuration. If configured, a TTP tab is displayed for each TTP airport included in the SDRR configuration.

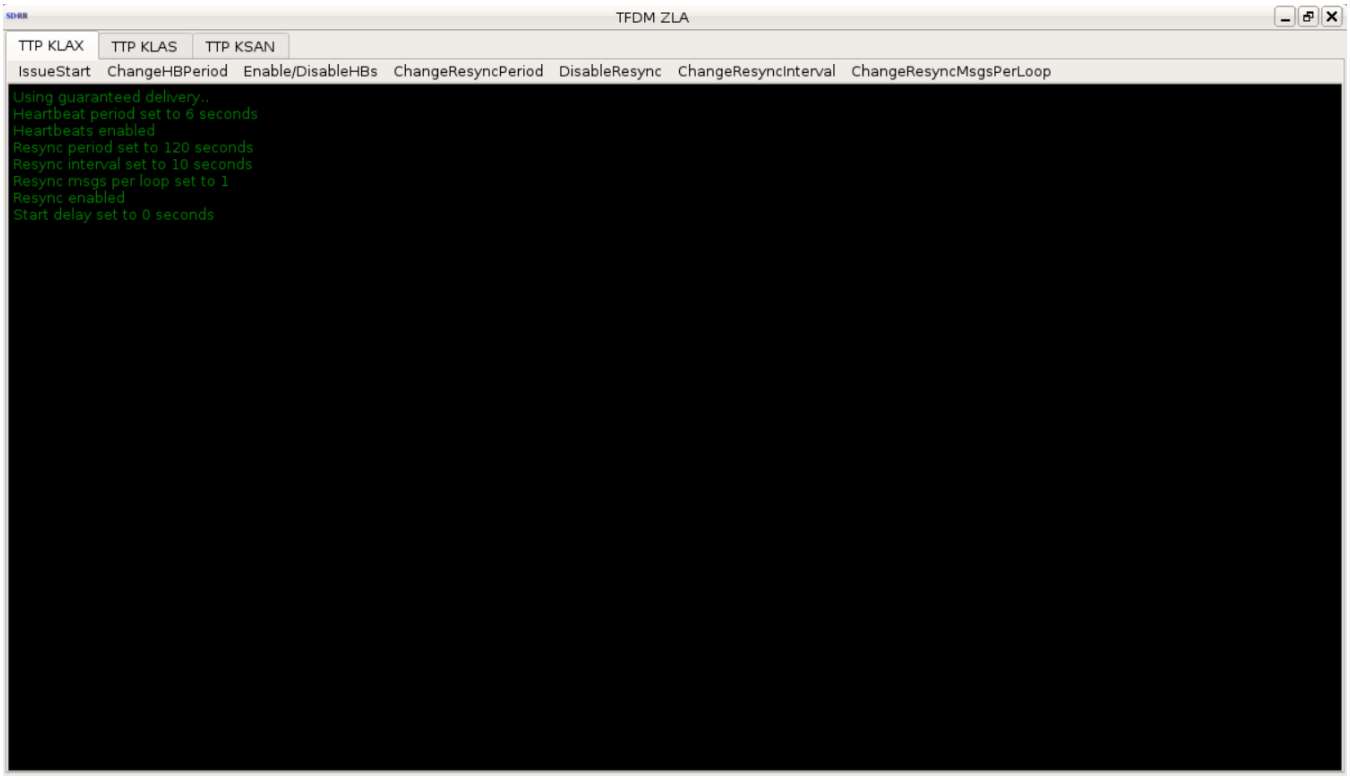

Figure 70. TFDM Tab - TBFM

Each RTCS tab has the following buttons:

IssueReconRequest

Clicking on this button causes a solicited reconstitution request message to be sent.

ChangeHBPeriod

This button launches a dialog box where the heartbeat period can be changed. The heartbeat period is specified in seconds and controls the amount of time between heartbeat messages.

DisableHBs

This button launches a dialog box where the heartbeat can suspended for the specified number of counts or permanently disabled.

DisableReceiptAcks

This button launches a dialog box where the ReceiptAcks value can be enabled or disabled. When disabled, the RTCS Airport will not reply with a receipt acknowledgement message.

DelayReceiptAcks

This button launches a dialog box where the AckDelay value can be changed. The AckDelay value sets the number of seconds the RTCS Airport will wait before replying with a receipt acknowledgement message.

MaxRetries

This button launches a dialog box where the Maximum Retries value can be changed. The Maximum Retries value sets the number of times the RTCS Airport will resend a message for which an acknowledgement message was not received.

RetransmitTimeout

This button launches a dialog box where the Retransmit Timeout value can be changed. The Retransmit Timeout value sets the number of seconds the RTCS Airport will wait before resending a message for which an acknowledgement message was not received.

UnsolicitedReconWaitTime

This button launches a dialog box where the unsolicited recon wait time can be changed. The recon wait time is specified in seconds. After receiving a heartbeat message with a new or changed service start time, the RTCS Airport will wait the indicated amount of time for an unsolicited reconstitution message. If a reconstitution message is not received in the specified time, the RTCS Airport will send a reconstitution request.

Show Flights

This button displays the RTCS Flight Viewer table with all of the flights received in rtcsFlt messages that are applicable to the RTCS Airport. Right clicking on an aircraft ID in the table displays options to schedule, cancel, and acknowledge a release time request. Selecting the schedule option opens a dialog where a runway and an external release time can be entered. Selecting the cancel option causes a release request message with a schedule activity of CANCEL to be sent. Selecting the acknowledge option causes a release request message with a schedule activity of ACK to be sent. Note that this Viewer is not updated dynamically; it must be closed and re-opened to view the most current entries.

Each TTP window has the following buttons:

IssueStart

Clicking on this button causes the startup sequence of messages (System Start, Periodic Start, and Periodic End) to be sent.

ChangeHBPeriod

This button launches a dialog box where the heartbeat period can be changed. The heartbeat period is specified in seconds and controls the amount of time between heartbeat messages.

Enable/DisableHBs

This button launches a dialog box where the heartbeat can suspended for the specified number of counts or permanently disabled.

ChangeResyncPeriod

This button launches a dialog box where the resync period can be changed. The resync period is specified in minutes and controls the amount of time between publications of the startup sequence of messages (System Start, Periodic Start, and Periodic End).

DisableResync

This button launches a dialog box where the resync publications of the startup sequence of messages (System Start, Periodic Start, and Periodic End) can suspended for the specified number of counts or permanently disabled.

ChangeResyncInterval

This button launches a dialog box where the resync batch interval wait time can be changed. The resync batch interval wait time is specified in milliseconds and controls the amount of time between the end on one batch and the start of the next batch. The default value is 1000ms.

ChangeResyncMsgsPerLoop

This button launches a dialog box where the resync messages per loop can be changed. The resync messages per loop is controls the number of TTP messages that are sent per batch. The default batch size is 10 messages.

5.2.20. NADIN Service Tab

The NADIN Service tab sends the FPLs to target destinations. These destinations can be added through the Create User button. The Toggle Quiet Mode stops the messages from appearing.

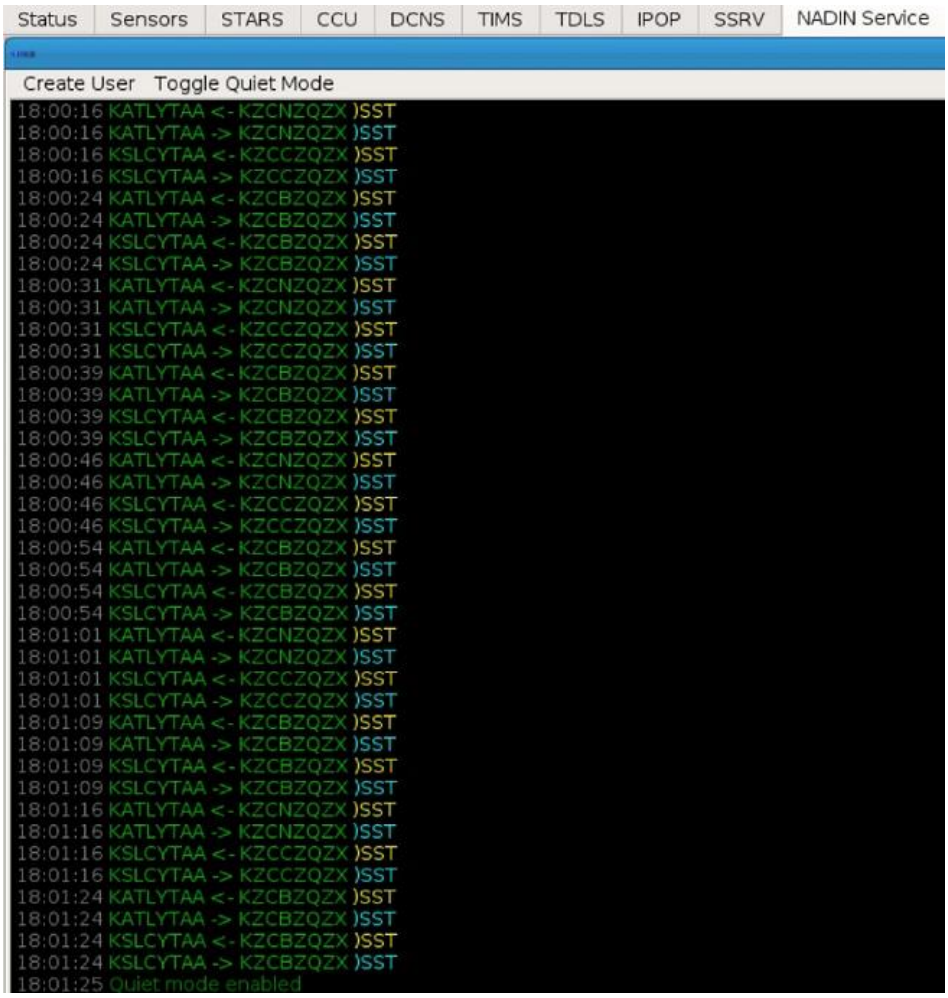

Figure 71. NADIN Service Tab

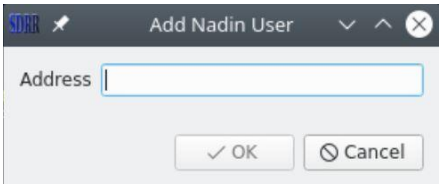

Figure 72**. Add NADIN User Dialog**

5.2.21. NADIN Users Tab

NADIN Users tab shows the FPLs receiving an ACK. If the FPL does not receive an ACK, the user can try to send a message to manually attempt to ACK the FPL. The send NADIN message allows the user to input a destination address of the NADIN client the user is sending the message to. The text is the actual message being sent. Send Cancel sends a NADIN CNL message to cancel a flight.

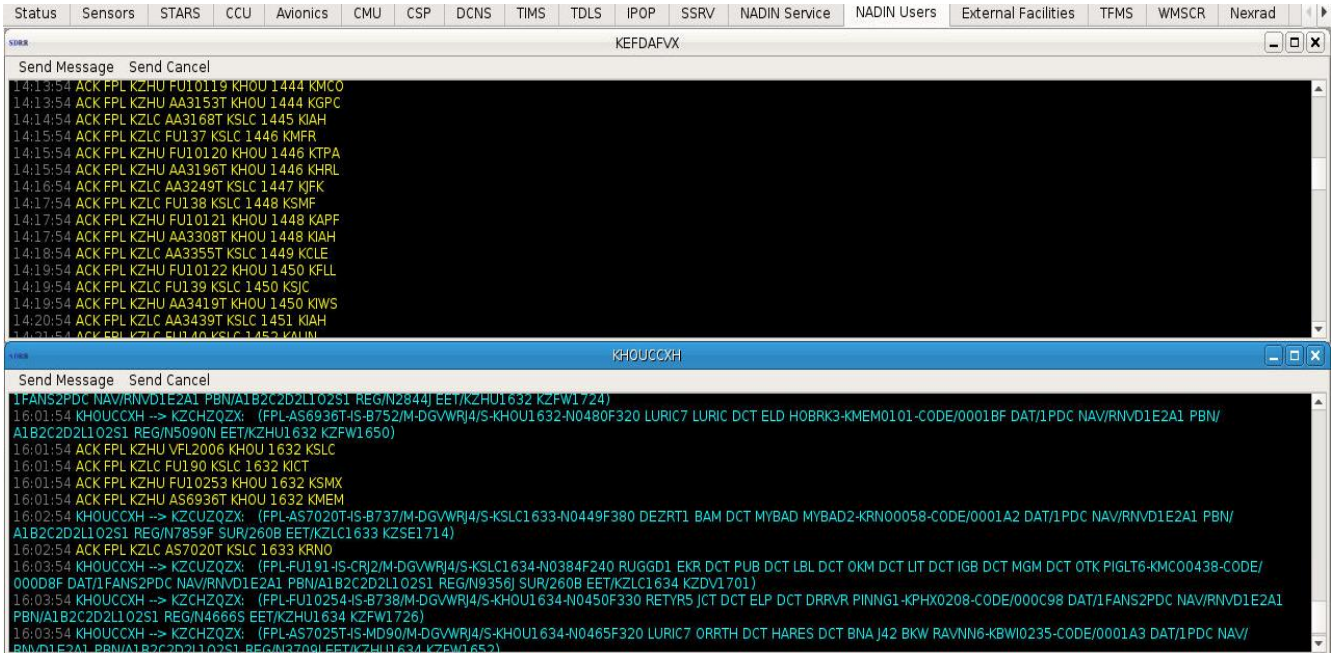

Figure 73. NADIN Users Tab

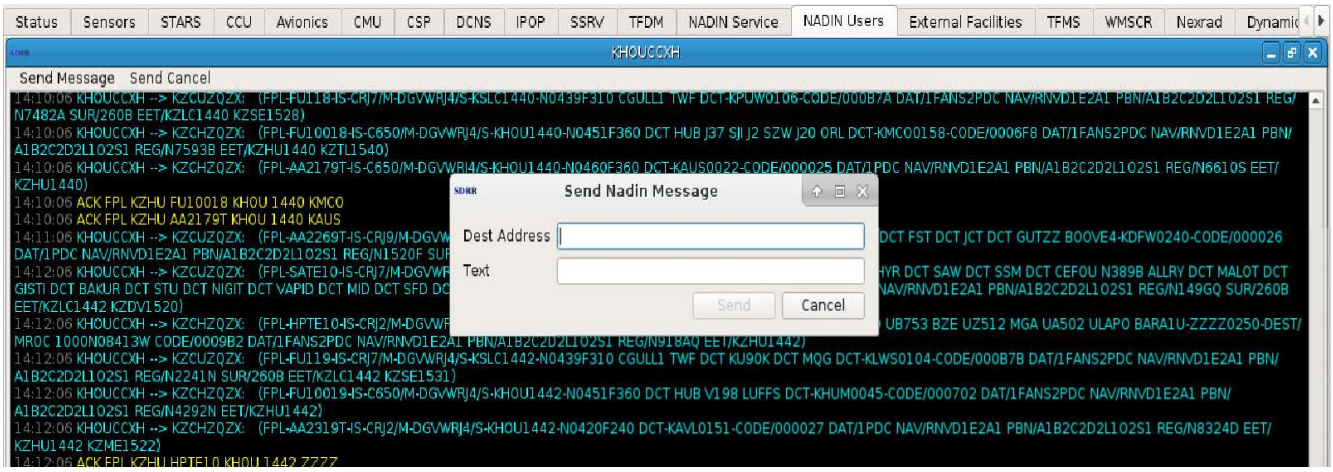

Figure 74. Send NADIN Message

5.2.22. External Facilities Tab

The External Facilities tab displays all the facilities involved with the users chosen NAP/Local Site(s). For Host, the user will be able to send IFMessages from the external site to the NAP/local sites. For NonUS, the user will be able to send ICAO messages and MOD/CHG from the external site to the NAP/local sites.

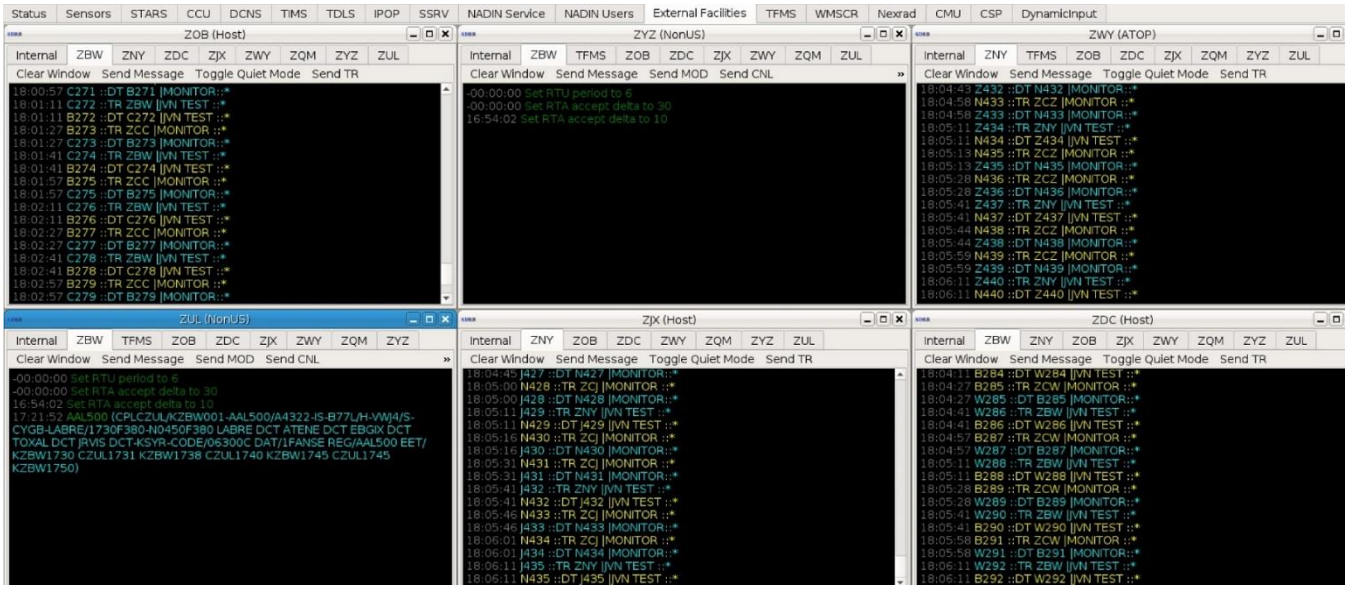

Figure 75. External Facilities

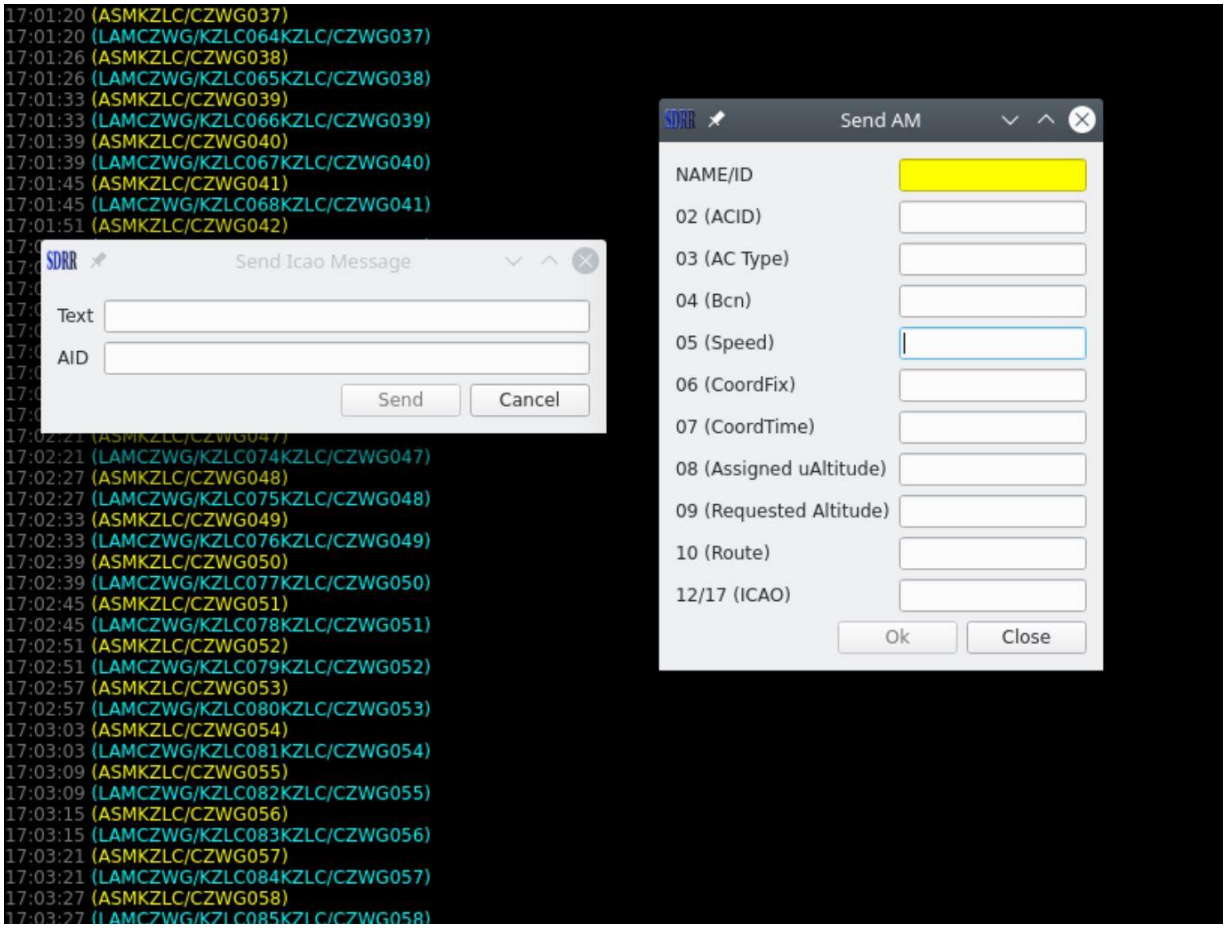

Figure 76. Send Message Dialog – Non US

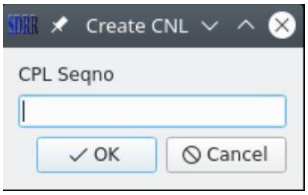

Figure 77. Send CNL Dialog

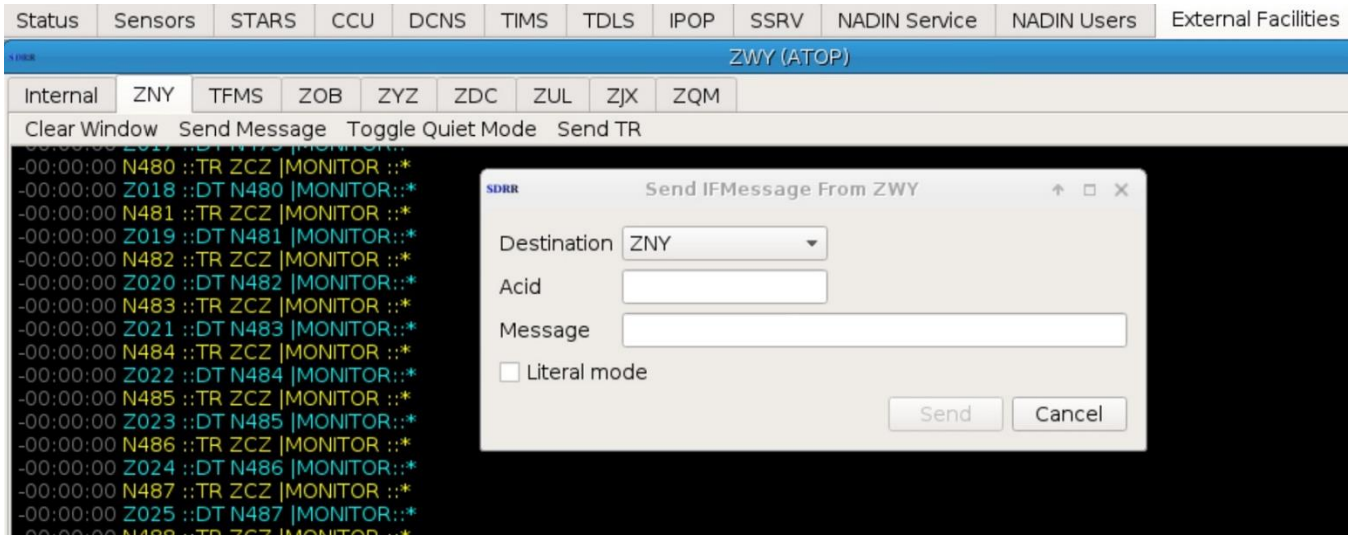

Figure 78**. Send Message Dialog – ATOP**

5.2.23. TFMS Tab

The TFMS tab displays ASDI, SWIM, and CMS messages.

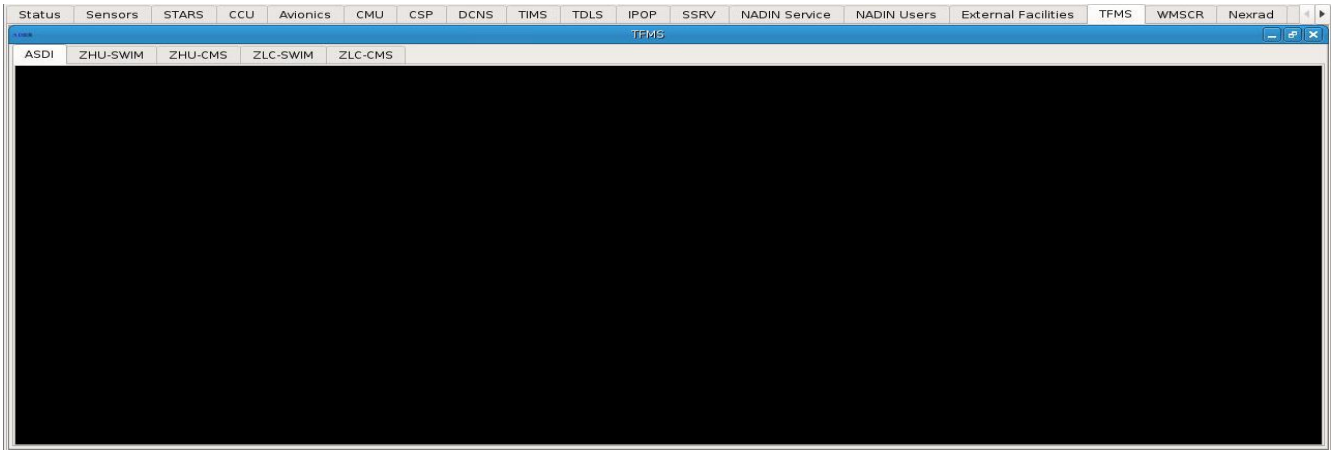

Figure 79. TFMS Tab - ASDI

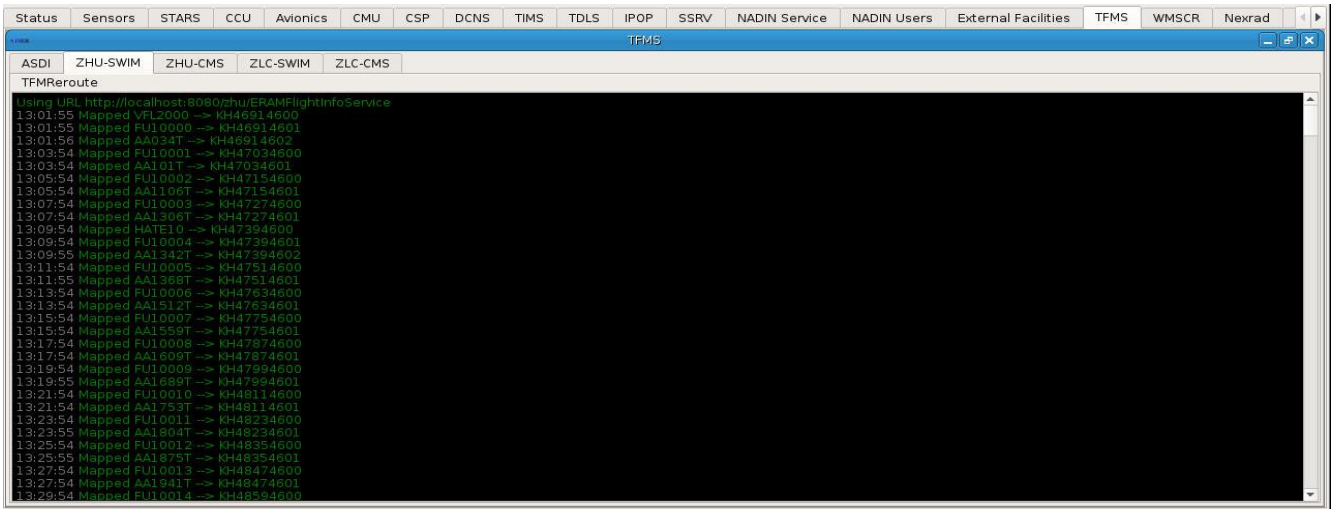

Figure 80. TFMS Tab - SWIM

Figure 81. TFMS Tab – CMS

5.2.24. WMSCR Tab

The WMSCR tab allows viewing and sending WMSCR messages in the SDRR interface.

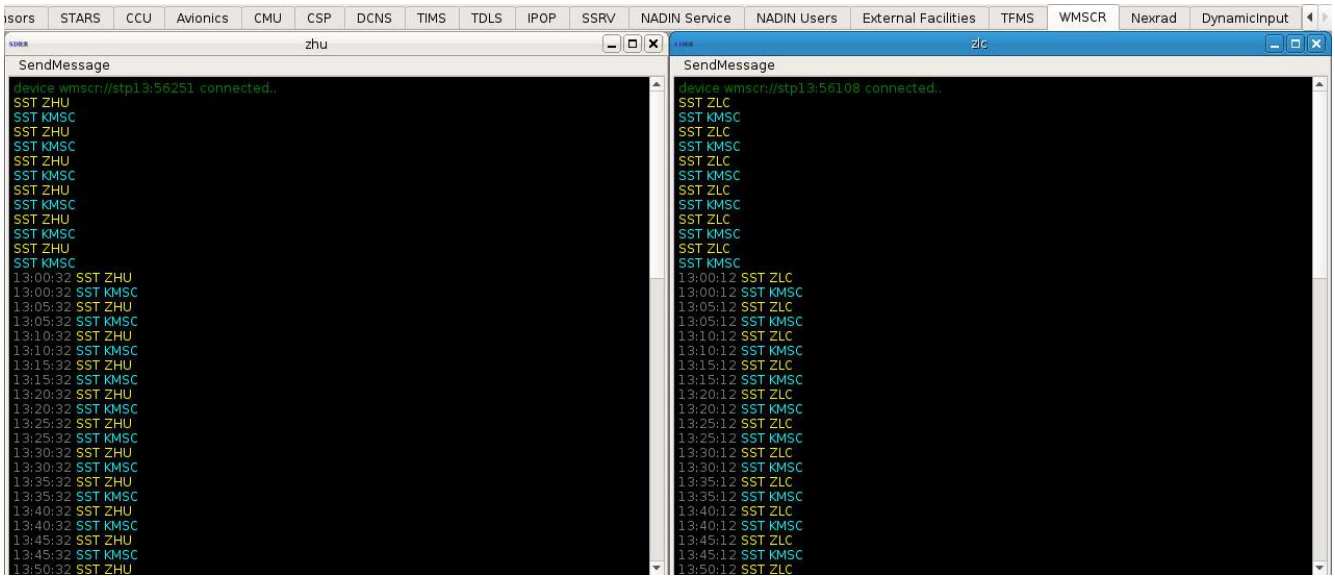

Figure 82. WMSCR Tab

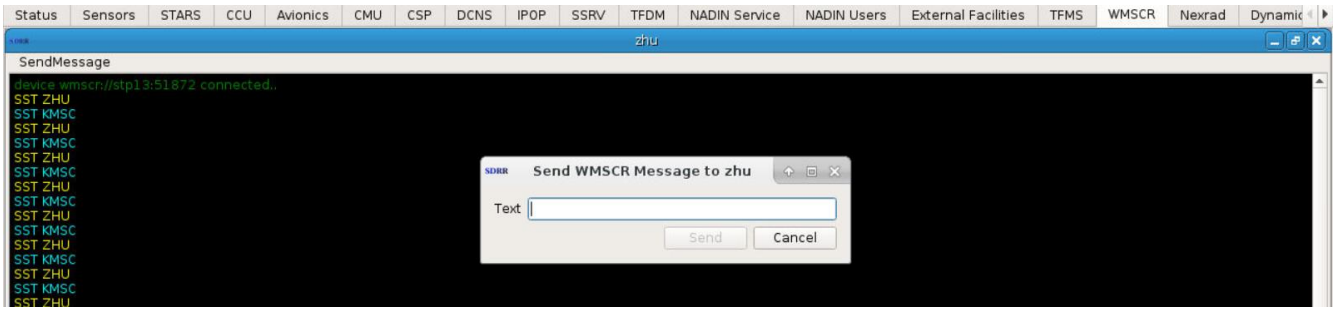

Figure 83. Send WMSCR Message

5.2.25. Nexrad Tab

The Nexrad tab allows the ability see precipitation data on the SDRR interface.

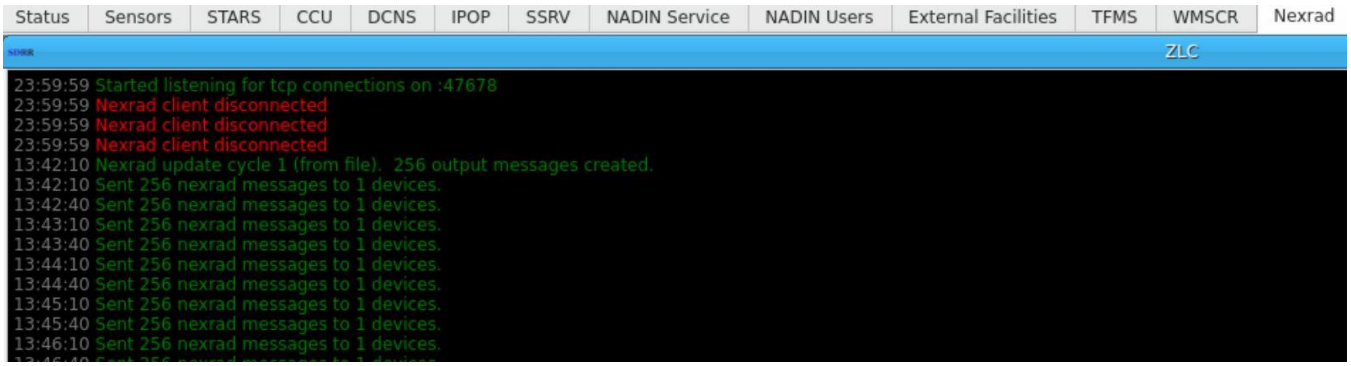

Figure 84. NEXRAD Tab

5.2.26. Dynamic Input Tab

The Dynamic Input tab displays Dynamic Precip Status, Dynamic Target Status, and Dynamic Message Status. These windows all display port statistics: whether the connection is active, down and the packet count

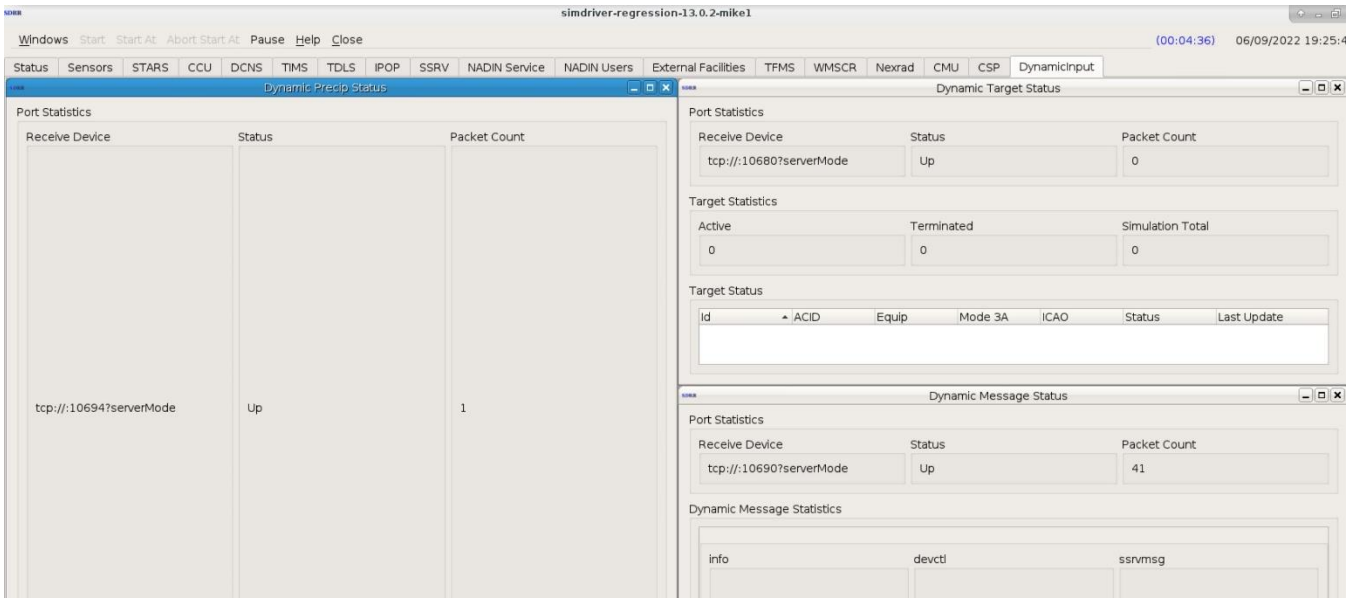

Figure 85. Dynamic Input Tab

6. Log Files

SDRR log files provide a record of the status of each run and the messages exchanged with the live and simulated systems. This file contains details from the run including the configuration file(s), scenario, start time, any special feature that was enabled, and all the message transactions that took place with timestamps. The figure below is a screenshot of an SDRR log file. By default, the log file is located in the directory specified by the environment variable ${SDRR_LOG_PATH}. The log files include a timestamp in the file name so they are not overwritten.

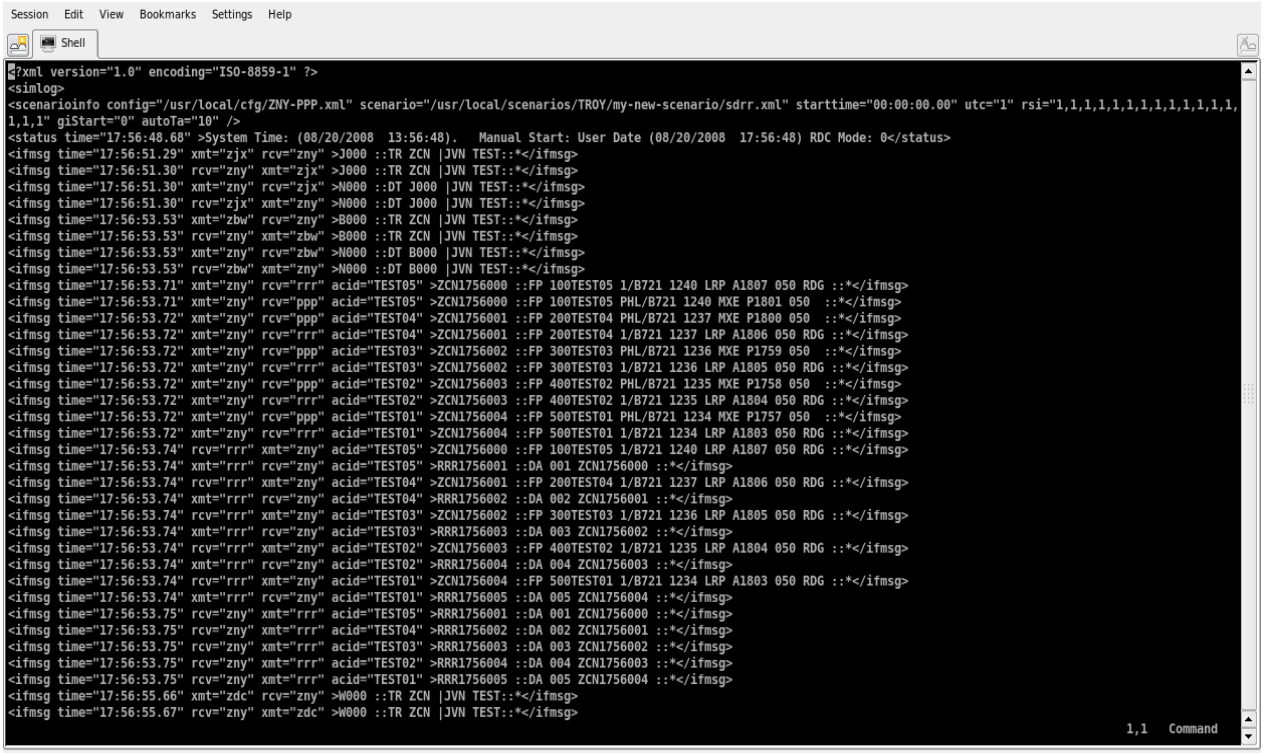

Figure 86. SDRR Log File

7. Surveillance Simulation, Recording, and Playback

SDRR has the capability to simulate surveillance data from several types of sources.

7.1. Automatic Dependent Surveillance – Broadcast (ADS-B)

SDRR has the capability to simulate ADS-B data when adapted in the configuration file. The 'svol' XML tag supports the generation of multiple streams of ADS-B data. Each stream represents the data on a different UDP port. The example SDRR configuration file (cfg.xml) below assumes a route has been configured on the processor. If a route has not been defined, an ethX device is added to the multicast address; e.g., "multi://224.1.1.100:59950?interface=eth3".

Example cfg.xml file:

```
<simconfig>
   <svol name="ppp-ads" sac=”0xab” sic=”0x0d” svType=”1”>
       <stream name="uat" device="multi://224.1.1.100:59950?interface=eth3" />
       <stream name="1090" device=" multi://224.1.1.100:59951?interface=eth3" />
       <stream name="equip" device=" multi://224.1.1.100:59952?interface=eth3" />
       <stream name="svol" device=" multi://224.1.1.100:59953?interface=eth3" />
       <stream name="sdp" device=" multi://224.1.1.100:59954?interface=eth3" />
   </svol>
</simconfig>
```

This configuration can be used for both playback and recording, allowing ADS-B data to be recorded simultaneously with radar data without using separate programs such as wireshark, ethereal, or tcpdump. This recording can also be played back on the same individual streams. Wireshark recordings can be played back on individual streams also by running the pcap2jvn utility once for each stream specifying the UDP port and a unique file name. A scenario file can then be created with the converted stream files. The scenario file (sdrr.xml) below shows an example of ADS-B data exported from a scenario or created from recordings with each stream in a separate .ast file.

Example sdrr.xml file:

```
<sim>
   <svol name="ppp-ads">
      <stream file="ppp-ads-uat.ast" name="uat"/>
      <stream file="ppp-ads-1090.ast" name="1090"/>
      <stream file="ppp-ads-equip.ast" name="equip"/>
      <stream file="ppp-ads-svol.ast" name="svol"/>
      <stream file="ppp-ads-sdp.ast" name="sdp"/>
   </svol>
</sim>
```

7.2. Wide Area Multilateration (WAM)

SDRR has the capability to simulate and record WAM data when adapted in the config.xml file. The 'wam' XML tag supports the playback of multiple streams and each stream represents the data on a different UDP port. See below for cfg.xml and sdrr.xml examples: The example below assumes a route has been configured on the processor, if not defined, an ethX device is added to the multicast address; e.g., "multi://224.1.1.100:59970?interface=eth3".

Example cfg.xml file:

```
<simconfig>
   <wam name="wamsvg" period="3" radius="60" sac="0xbb" sic="0x02" spos="+39:51:33.00,-
075:16:00.30">
         <stream name="modeS" device=" multi://224.1.1.100:59970?interface=eth3"/>
         <stream name="1090" device=" multi://224.1.1.100:59971?interface=eth3"/>
         <stream name="uat" device=" multi://224.1.1.100:59972?interface=eth3"/>
         <stream name="atcrbs" device=" multi://224.1.1.100:59973?interface=eth3"/>
         <stream name="svol" device=" multi://224.1.1.100:59974?interface=eth3"/>
         <stream name="sdp" device=" multi://224.1.1.100:59975?interface=eth3"/>
         <radio_station name="WM1" lid="7"/>
   </wam>
</simconfig>
```

Example sdrr.xml file:

```
<sim>
   <wam name="wamsvg">
      <stream name="modeS" file="srv/wm1-modeS.ast"/>
      <stream name="1090" file="srv/wm1-1090.ast"/>
      <stream name="uat" file="srv/wm1-uat.ast"/>
      <stream name="atcrbs" file="srv/wm1-atcrbs.ast"/>
      <stream name="wam" file="srv/wm1-wam.ast"/>
      <stream name="sdp" file="srv/wm1-sdp.ast"/>
   </wam>
</sim>
```

8. SDRR Configuration Files

The SDRR configuration file is an XML file which defines the facilities and sensors to be simulated or physically connected.

8.1. Configuration File Format

localhost

Defines a NAS En Route facility. This facility has one interfacility device per I/O thread.

Attributes:

name

Used as the title of the display window for this facility, and is matched with the "src" field in the scenario file format for messages (see Scenario File Format).

facName

Defines the 3-letter identifier to be used on the wire to identify this site. If not defined, it defaults to the first 3 characters of "name".

facID

Defines the 1-letter identifier to be used on the wire to identify this site. If not defined, defaults to the 3rd character of "facName".

autoTA

Enables the automatic TA response for this facility. Defines the number of seconds to wait before sending an automatic TA response to TI message (default is 0).

autoTR

Enables periodic transmission of TR message, defines the interval in seconds (default is 0).

Threads:

hostio

Defines an interface to a NAS host.

Attributes:

name

Used as the title of the display window for this facility, and is matched with the "dest" field in the scenario file format for messages (see Scenario File Format).

facName

Defines the 3-letter identifier to be used on the wire to identify this site. If not defined, it defaults to the first 3 characters of “name”.

facID

Defines the 1-letter identifier to be used on the wire to identify this site. If not defined, defaults to the 3rd character of “facName”.

device

Device or file name.

txclock

The baud rate of the transmit side of the assigned device. 0 means accept the transmit clock from the DCE (default is 2400).

rxclock

The baud rate of the receive side of the assigned device. 0 means accept the receive clock from the DCE (default is 0).

tangent

Lat/Long Point Of Tangency.

org

X/Y value used for dynamic interfacility messaging.

magdev

Magnetic deviation, in degrees.

starsio

Defines an interface to a Terminal facility.

Attributes:

name

Used as the title of the display window for this facility, and is matched with the “dest” field in the scenario file format for messages (see Scenario File Format).

facName

Defines the 3-letter identifier to be used on the wire to identify this site. If not defined, it defaults to the first 3 characters of “name”.

facID

Defines the 1-letter identifier to be used on the wire to identify this site. If not defined, defaults to the 3rd character of "facName".

device

Device or file name.

txclock

The baud rate of the transmit side of the assigned device. 0 means accept the transmit clock from the DCE (default is 2400).

rxclock

The baud rate of the receive side of the assigned device. 0 means accept the receive clock from the DCE (default is 0).

tangent

Lat/Long Point Of Tangency.

org

X/Y value used for dynamic interfacility messaging.

magdev

Magnetic deviation, in degrees.

eramsim

Defines a simulated ERAM facility.

Attributes:

name

Used as the title of the display window for this facility.

Threads:

eddsServer

Defines the connection between the simulated ERAM and an external EDDS.

Attributes:

listenAddress

Defines the local address on which SDRR will listen for the EDDS connects. The EDDS must be configured with the same address.

port

Defines the port number for the EDDS connection. The EDDS must be configured with the same port number.

clientDevice

Defines the address and port which SDRR will use for CMS data feedback.

stars

Defines a terminal STARS facility within the boundaries of the simulated ERAM center.

Attributes:

name

Used as the title of the display window for this facility.

device

Physical device or file name.

txclock

The baud rate of the transmit side of the assigned device. 0 means accept the transmit clock from the DCE (default is 2400).

rxclock

The baud rate of the receive side of the assigned device. 0 means accept the receive clock from the DCE (default is 0).

tangent

Lat/Long Point Of Tangency.

magdev

Magnetic deviation, in degrees.

stars

Defines a terminal STARS facility. This facility has one interfacility device associated with it, with one or more I/O threads to handle communications to other facilities which are all routed through the single interfacility device.

Attributes:

device

Physical device or file name.

name

Used as the title of the display window for this facility, and is matched with the “src” field in the scenario file format for messages (see Scenario File Format).

facName

Defines the 3-letter identifier to be used on the wire to identify this site. If not defined, it defaults to the first 3 characters of “name”.

facID

Defines the 1-letter identifier to be used on the wire to identify this site. If not defined, defaults to the 3rd character of “facName”.

autoTA

Enables the automatic TA response for this facility. Defines the number of seconds to wait before sending an automatic TA response to TI message (default is 0).

autoTR

Enables periodic transmission of TR message, defines the interval in seconds (default is 0).

Threads:

hostio

Defines an interface to a NAS host.

Attributes:

name

Used as the title of the display window for this facility, and is matched with the “dest” field in the scenario file format for messages (see Scenario File Format).

facName

Defines the 3-letter identifier to be used on the wire to identify this site. If not defined, it defaults to the first 3 characters of “name”.

facID

Defines the 1-letter identifier to be used on the wire to identify this site. If not defined, defaults to the 3rd character of “facName”.

txclock

The baud rate of the transmit side of the assigned device. 0 means accept the transmit clock from the DCE (default is 2400).

rxclock

The baud rate of the receive side of the assigned device. 0 means accept the receive clock from the DCE (default is 0).

tangent

Lat/Long Point Of Tangency.

org

X/Y value used for dynamic interfacility messaging.

magdev

Magnetic deviation, in degrees.

artsio

Defines an interface to a Terminal facility.

Attributes:

name

Used as the title of the display window for this facility, and is matched with the "dest" field in the scenario file format for messages (see Scenario File Format).

facName

Defines the 3-letter identifier to be used on the wire to identify this site. If not defined, it defaults to the first 3 characters of "name".

facID

Defines the 1-letter identifier to be used on the wire to identify this site. If not defined, defaults to the 3rd character of "facName".

txclock

The baud rate of the transmit side of the assigned device. 0 means accept the transmit clock from the DCE (default is 2400).

rxclock

The baud rate of the receive side of the assigned device. 0 means accept the receive clock from the DCE (default is 0).

tangent

Lat/Long Point Of Tangency.

org

X/Y value used for dynamic interfacility messaging.

magdev

Magnetic deviation, in degrees.

radar

Defines a radar sensor.

Attributes:

name

Used as the title of the display window for this facility, and is matched with the “src” field in the scenario file format for messages (see Scenario File Format).

type

Radar type (defaults to LRR).

device

Device or file name.

chans

Number of channels (default is 0, which means auto-determine based on type).

magdev

Magnetic deviation of the radar, in degrees.

parrots

Defines the beacon code(s) squawked by parrot targets.

scantime

Amount of time the radar takes to complete one sweep, in seconds.

elev

Elevation of the radar, in feet above MSL.

pos

Lat/long position of the radar.

svol

Defines a service volume for ADS-B.

Attributes:

name

Used as the title of the display window for this facility, and is matched with the “src” field in the scenario file format for messages (see Scenario File Format).

pos

Lat/long position.

Threads:

stream

Defines the type(s) of message supported.

Attributes:

name

Used as the title of the display window, and is matched with the “src” field in the scenario file format for messages (see Scenario File Format). It must be one of: UAT, 1090, Equip, SVol, or SDP.

device

Device or file name.

wam

Defines a service volume for WAM.

Attributes:

name

Used as the title of the display window for this facility, and is matched with the “src” field in the scenario file format for messages (see Scenario File Format).

pos

Lat/long position.

Threads:

stream

Defines the type(s) of message supported.

Attributes:

name

Used as the title of the display window, and is matched with the “src” field in the scenario file format for messages (see Scenario File Format). It must be one of: UAT, 1090, Equip, SVol, or SDP.

device

Device or file name.

asdex

Defines an ASDEX Stream.

Attributes:

name

Used as the title of the display window for this facility, and is matched with the “src” field in the scenario file format for messages (see Scenario File Format).

ttl

Defines the time to live.

etms

Defines an ETMS facility.

Attributes:

starsid

Defines a three-letter ID for use by a STARS facility.

name

Used as the title of the display window for this facility, and is matched with the “src” field in the scenario file format for messages (see Scenario File Format).

device

Device or file name.

rate

Baud rate.

8.2. Example Configuration Files

8.2.1. ERAM Simulation in Direct Mode

To simulate ERAM, SDRR must be configured with an eramsim source and an eddserver definition with connection information for an EDDS server. To connect to an EDDS server, the SDRR configuration file needs to have the "listenAddress" set to the network interface address of the processor running SDRR. On the EDDS server, the configuration file ${HDDS_SSP}/hid_address.adp should have the same host/port pair configured.

```
<root>
  <sources localhost="zla"
    <eramsim name="zla">
      <eddserver port="%{ZLA_EDDS_PORT}" listenAddress="${SDRR_HOSTNAME}"/>
      <clientDevice>pipe:zla-cms-%{USER}</clientDevice>
      <stars name="ttt" device="tcps:${SDRR_HOSTNAME}/%{AIG1_SCT_PORT}"
       tangent="+33:47:30.41,-118:00:08.06" magdev="14.0"/>
    </eramsim>
  </sources>
</root>
```

8.2.2. ERAM Simulation in Mixed Mode

Except for the stars definition, the eramsim stanza should be configured the same way as for the Direct Mode simulation. To drive a live STARS system, the site should be added to the SDRR configuration file inside the eramsim stanza. The site should have the facility name as it is adapted in ERAM adaptation and the device should be configured for a physical IFDT card connected to the STARS system. For example, the physical card that connects to the STARS system is installed in an SDRR SIRS processor, such as sirs16@/dev/if0. Note that this interface may be configured differently for each STARS system.

```
<root>
  <sources localhost="zdc"
    <eramsim name="zdc">
      <eddserver port="%{ZDC_EDDS_PORT}" listenAddress="${SDRR_HOSTNAME}"/>
      <clientDevice>pipe:zdc-cms-%{USER}</clientDevice>
      <stars name="acy" device="sirs16@/dev/if0" tangent="+39:27:10.00,-074:35:31.00"
       magdev="-12.0" rxclock="2400" txclock="2400"/>
    </eramsim>
  </sources>
</root>
```

8.2.3. ERAM in a Box Interface

The configuration format is slightly different for ERAM in a Box (EIB), such as those running in the Virtual Test Lab (VTL), than it is for an ERAM Test Bed connected to an En Route Communications Gateway (ECG). Note that the examples below are only a sample of the most commonly used types of interfaces and devices; there are many more types and optional attributes that could be configured depending on the specific test need.

The local terminal facilities, CCUs, SSRV injection positions, and hgi interfaces (ECG emulation devices) are defined in the SDRR non-surveillance configuration file for each en route facility:

```
<root>
  <sources localhost="zdv">
    <stars name="ras" device="hgi://%{SDRR_ZDV}?eram=ZDV&amp;device=RAS" facName="ras"
autoTR="0">
     <hostio name="zdv" facName="zcd" magdev="9.00" tangent="+39:13:54.00,-106:52:59.00"/>
     <aig>
       <clientDevice>xmlstream://%{SIMDRIVER_IP}:%{SIMDRIVER_AIG_PORT}</clientDevice>
     </aig>
    </stars>
...
    <stars name="rdc" device="hgi://%{SDRR_ZDV}?eram=ZDV&amp;device=RDC" facName="rdc"
autoTR="0">
     <hostio name="zdv" facName="zcd" magdev="8.00" tangent="+39:51:17.00,-104:43:06.00"/>
     <starsio name="rco" magdev="9.00" tangent="+38:48:02.00,-104:40:42.00"/>
     <starsio name="rcy" magdev="9.00" tangent="+41:07:59.00,-104:52:01.00"/>
     <aig>
       <clientDevice>xmlstream://%{SIMDRIVER_IP}:%{SIMDRIVER_AIG_PORT}</clientDevice>
     </aig>
    </stars>
...
    <ccu facility="zdv" id="1" indevice="hgi://%{SDRR_ZDV}?eram=ZDV&amp;device=CCU1O"
outdevice="hgi://%{SDRR_ZDV}?eram=ZDV&amp;device=CCU1I"/>
    <!--ssrvManager note: ZDV ssrv (k3.eab on mmp) is listening on port 48023.-->
    <ssrvinj facility="zdv" device="pipe:zdv-A-ssrvinj-user" active="1" channel="A"
exercise="15" maxMsgsPerSec="-1">
     <positions>
       <position>D03</position>
...
       <position>E9</position>
     </positions>
```

```
    </ssrvinj>
  </sources>
  <hgi name="ZDV" clientInterface="%{SDRR_ZDV}" hgiInterface="%{ZDV_ERAM_INTERFACE}">
    <ccu name="CCU1I" id="1" lda="0x100" writeOnly="1"/>
    <ccu name="CCU1O" id="1" lda="0x101" readOnly="1"/>
    <interfacility name="RAS" lda="0x102"/>
    <interfacility name="RCO" lda="0x103"/>
    <interfacility name="RCY" lda="0x104"/>
    <interfacility name="RDC" lda="0x105"/>
    <interfacility name="RKP" lda="0x106"/>
    <interfacility name="RRC" lda="0x107"/>
    <interfacility name="ZAB" lda="0x108"/>
    <interfacility name="ZKC" lda="0x109"/>
    <interfacility name="ZLA" lda="0x10a"/>
    <interfacility name="ZLC" lda="0x10b"/>
    <interfacility name="ZMP" lda="0x10c"/>
  </hgi>
</root>
```

8.2.4. ERAM Test Bed Interface

For the ERAM Test Bed, the local terminal facilities, CCUs, and SSRV injection positions are defined in the SDRR non-surveillance configuration file for each En Route facility:

```
<root>
  <sources  localhost="zdv">
    <stars  name="ras"  device="(ecgif://pipa?device=RAS+
ecgif://pipb?device=RAS)"  facName="ras"  autoTR="0">
      <hostio  name="zdv"  facName="zcd"  magdev="9.00"  tangent="+39:13:54.00,
-106:52:59.00"/>
    </stars>
...
    <stars  name="rdc"  device="(ecgif://pipa?device=RDC+
ecgif://pipb?device=RDC)"  facName="rdc"  autoTR="0">
      <hostio  name="zdv"  facName="zcd"  magdev="8.00"  tangent="+39:51:17.00,
-104:43:06.00"/>
      <starsio  name="rcy"  magdev="9.00"  tangent="+41:07:59.00,
-104:52:01.00"/>
```

```
     <starsio  name="rco"  magdev="9.00"  tangent="+38:48:02.00,
-104:40:42.00"/>
   </stars>
...
   <ccu  facility="zdv"  id="1"
indevice="(ecggpo://pipa?device=CCU1O+ecggpo://pipb?device=CCU1O)"
outdevice="(ecggpi://pipa?device=CCU1I+ecggpi://pipb?device=CCU1I)"/>
   <ssrvinj  facility="zdv"  device="mhp:24000/localhost"  active="1"  channel="A"
exercise="13"  maxMsgsPerSec="-1">
     <positions>
       <position>D03</position>
       <position>R03</position>
...
       <position>E9</position>
     </positions>
   </ssrvinj>
  </sources>
</root>
```

8.2.5. National Configuration for EIB

For the EIB, all of the neighboring EnRroute facilities, ATOP, non-US, NADIN, TFMS, and WMSCR interfaces are defined in one SDRR national file for the entire configuration:

```
<root>
  <sources>
   <externalFacility  name="zoa"  isHost="1"  srcATS="kzoa">
     <eramInterface  name="zla"  destATS="kzla">
       <interfacility  device="hgi://%{SDRR_ZLA}?eram=ZLA&amp;device=ZOA"  org="-468.56,-
339.00"  tangent="+34:58:41.00,-116:07:07.00"/>
     </eramInterface>
     <eramInterface  name="zlc"  destATS="kzlc">
       <interfacility  device="hgi://%{SDRR_ZLC}?eram=ZLC&amp;device=ZOA"  org="-491.42,-
478.81"  tangent="+42:25:31.00,-110:41:21.00"/>
     </eramInterface>
   </externalFacility>
   <externalFacility  name="zyz"  srcATS="czyz">
     <eramInterface  name="zmp"  destATS="kzmp">
       <!--<direct  device="nam://%{NMR_IP}:12016"/>-->
```

```
      <nadin  device="nadin://%{NMR_IP}:12017"  destAddress="KZCPZQZX"
srcAddress="CZYZZTON"/>
     </eramInterface>
   </externalFacility>
...
   <nadinService>
     <eramServer  listenAddress="%{NATIONAL_INTERFACE}"  port="20047"/>
     <externalServer  listenAddress="%{SDRR_NATIONAL}">
       <interface  caatsAddress="CZYZZTON"  eramAddress="KZCPZQZX">12017</interface>
       <interface  caatsAddress="MMTYZRZX"
eramAddress="KZCAZQZX">%{MTY_TO_ZAB_NADIN_PORT}</interface>
       <interface  caatsAddress="CZEGZGGG"  eramAddress="KZCUZQZX">12024</interface>
       <interface  caatsAddress="CZWGZPEG"  eramAddress="KZCUZQZX">12022</interface>
       <interface  caatsAddress="CZWGZPPP"  eramAddress="KZCPZQZX">12014</interface>
       <interface  caatsAddress="MMZTZRZX"
eramAddress="KZCLZQZX">%{MZT_TO_ZLA_NADIN_PORT}</interface>
       <interface  caatsAddress="MMZTZRZX"
eramAddress="KZCAZQZX">%{MZT_TO_ZAB_NADIN_PORT}</interface>
     </externalServer>
   </nadinService>
   <tfms>
     <asdiServer  listenAddress="%{NATIONAL_INTERFACE}"  port="9092"/>
     <eramInterface  name="zdv">
       <esas  url="http://localhost:8080/zdv/ERAMFlightInfoService"/>
       <cmsInput  device="pipe:swim-cms-zdv-user"/>
     </eramInterface>
...
   </tfms>
   <wmscr  listenAddress="%{NATIONAL_INTERFACE}">
     <eramServer  name="zdv"  port="50055"/>
     <eramServer  name="zla"  port="50061"/>
     <eramServer  name="zkc"  port="50060"/>
     <eramServer  name="zab"  port="50050"/>
     <eramServer  name="zlc"  port="50062"/>
     <eramServer  name="zmp"  port="50065"/>
   </wmscr>
  </sources>
</root>
```

8.2.6. National Configuration for ERAM Test Bed

For the ERAM Test Bed, all of the neighboring En Route facilities, ATOP, non-US, NADIN, TFMS, and WMSCR interfaces are defined in the SDRR national file for each en route facility:

```
<root>
  <sources>
    <externalFacility  name="zmp"  isHost="1"  srcATS="kzmp">
      <eramInterface  name="zdv"  destATS="kzdv">
        <interfacility  device="(ecgif://pipa?device=ZMP+
ecgif://pipb?device=ZMP)"  org="-381.94,-476.82"  tangent="+41:11:51.00,-106:27:55.00"/>
      </eramInterface>
    </externalFacility>
    <externalFacility  name="zab"  isHost="1"  srcATS="kzab">
      <eramInterface  name="zdv"  destATS="kzdv">
        <interfacility  device="(ecgif://pipa?device=ZAB+
ecgif://pipb?device=ZAB)"  org="-381.94,-476.82"  tangent="+41:11:51.00,-106:27:55.00"/>
      </eramInterface>
    </externalFacility>
...
    <nadinService>
      <eramServer  listenAddress="%{NATIONAL_INTERFACE}"  port="20047"/>
      <externalServer  listenAddress="%{SDRR_NATIONAL}"/>
    </nadinService>
    <tfms>
      <asdiServer  listenAddress="%{NATIONAL_INTERFACE}"  port="9092"/>
      <eramInterface  name="zdv">
        <esas  url="http://localhost:8080/zdv/ERAMFlightInfoService"/>
        <cmsInput  device="pipe:swim-cms-zdv-sdrr"/>
      </eramInterface>
    </tfms>
    <wmscr  listenAddress="%{NATIONAL_INTERFACE}">
      <eramServer  name="zdv"  port="50055"/>
    </wmscr>
  </sources>
</root>
```

8.2.7. STARS Simulation in Direct Mode

To simulate STARS and inject AIG messages directly into TBFM, each STARS site should be added to the SDRR configuration file inside the eramsim stanza and also as a stars stanza. The sites should have the

facility name as it is adapted in ERAM, and the TBFM name. The multicast addresses and ports defined in the TBFM customization should be added as tsas datasets inside the stars stanzas. These data sets define the devices that will be the interfaces for the various categories of AIG message.

```
<stars name="ttt" device="tcp:${SDRR_HOSTNAME}/%{AIG1_SCT_PORT}" facName="ttt"
rxclock="0" txclock="0" autoTR="0">
  <hostio name="zla" facName="zcl" tangent="+33:47:30.41,-118:00:08.06" magdev="14.0"/>
  <aig pot="+33:58:50.00,-116:59:27.00" potOffset="1154.65,1706.24">
    <dataset id="7" outDev="(multi://%{AIG_MULTICAST_ADDRESS}:%{AIG1_SCT_PORT}"/>
    <dataset id="8" outDev="(multi://%{AIG_MULTICAST_ADDRESS}:%{AIG1_SCT_PORT}"/>
    <dataset id="9" outDev="(multi://%{AIG_MULTICAST_ADDRESS}:%{AIG1_SCT_PORT}"
 inDev="(multi://%{AIG_MULTICAST_ADDRESS}:%{AIG2_SCT_PORT}"/>
    <dataSet id="10" inDev="(multi://%{AIG_MULTICAST_ADDRESS}:%{AIG2_SCT_PORT}"/>
  </aig>
</stars>
```

8.2.8. TFMS Emulation

SDRR can emulate the Traffic Flow Management System (TFMS) and generate Aircraft Situation Display to Industry (ASDI) data.

```
<externalFacility name="mzt" bzDelta="10" srcATS="mmzt">
  <tfmsInterface device="pipe:mzt-tfms-%{USER}"/>
</externalFacility>
<externalFacility name="mty" bzDelta="10" srcATS="mmty">
  <tfmsInterface device="pipe:mty-tfms-%{USER}"/>
</externalFacility>
<externalFacility name="zak" isATOP="1" srcATS="kzak">
  <tfmsInterface device="pipe:zak-tfms-%{USER}"/>
</externalFacility>
<tfms>
  <asdiServer listenAddress="tbfmsdrr01" port="41507"/>
  <externalInterface name="mzt" device="pipe:mzt-tfms-%{USER}" asdiName="mmex"/>
  <externalInterface name="mty" device="pipe:mty-tfms-%{USER}" asdiName="mmex"/>
  <externalInterface name="zak" device="pipe:zak-tfms-%{USER}" asdiName="kooa"/>
</tfms>
```

8.2.9. Terminal Radar

SDRR will need configuration files for the Terminal radar sites and static messages to generate radar data. The radar sites should be configured with physical radar interface cards connected to the STARS system. For example, the physical cards that connect to the STARS system are often installed in an SDRR SIRS slave processor, such as sirs16s1@/dev/srr0. Note that this could be configured differently for each STARS system. The radar configuration files should be specified on the SDRR command line.

Terminal sensors file including status message definitions:

```
<radar name="acy" device="sirs16s1@/dev/srr0" type="asr9-modes" elev="165.00"
psrMaxRange="60" scantime="4.69" spos="+39:27:09.80,
-074:35:31.10" ssrMaxRange="60">
  <brtqc acps="2102" alt="-1000" bcn="7770" range="59.1"/>
  <srtqc acps="10" range="55.1" runlength="24"/>
  <parrot acps="977" mode3a="1274" modec="730" range="50"/>
  <permanentEcho acps="879" modec="0" range="1.6" runlength="24"/>
</radar>
...
<svol name="acy-ads" sac="0xac" sic="0x1e" svType="1">
  <stream name="uat" device="(sirs16@multi://239.161.7.30/59950?interface=2,3)"/>
  <stream name="1090" device="(sirs16@multi://239.161.7.30/59951?interface=2,3)"/>
  <stream name="equip" device="(sirs16@multi://239.161.7.30/59952?interface=2,3)"/>
  <stream name="svol" device="(sirs16@multi://239.161.7.30/59953?interface=2,3)"/>
  <stream name="sdp" device="(sirs16@multi://239.161.7.30/59954?interface=2,3)"/>
  <radio_station name="TTNGS" lid="3000" maxRange="60.00" spos="+40:16:40.11,
-074:49:10.16">
    <receiver id="0xd0260" icao="0xfaafaa" period="10.0" spos="+40:16:40.11,
-074:49:10.16" uat="0"/>
    <receiver id="0xd0261" icao="0xfaafaa" period="10.0" spos="+40:16:40.11,
-074:49:10.16" uat="0"/>
    <receiver id="0xd0262" icao="0xfaafaa" period="10.0" spos="+40:16:40.11,
-074:49:10.16" uat="0"/>
    <receiver id="0xd0263" icao="0xfaafaa" period="10.0" spos="+40:16:40.11,
-074:49:10.16" uat="0"/>
    <receiver id="0x90260" icao="0xfaafaa" period="5.0" spos="+40:16:40.11,
-074:49:10.16" uat="1"/>
  </radio_station>
...
</svol>
```

8.2.10. En Route Radar for EIB

For the EIB, En Route and Terminal radar sites and static messages are defined in the SDRR surveillance configuration file for each En Route facility:

```
  <sources>
    <radar  name="cdc-eram"  device="ecgp://zdvserver?artcc=ZDV&amp;radar=CDC"  type="arsr2"
elev="10786.21"  genStaticMsgs="0"  psrMaxRange="225"  scantime="12.00"
spos="+37:35:35.48,-112:51:49.20"  ssrMaxRange="225">
      <brtqc  acps="2080"  alt="-12"  bcn="0000"  range="1"/>
      <srtqc  acps="32"  range="1"  runlength="64"/>
      <parrot  acps="887"  mode3a="1274"  modec="10"  range="111.5"/>
      <parrot  acps="3149"  mode3a="1275"  modec="4087"  range="97.125"/>
    </radar>
    <radar  name="dbl-eram"  device="ecgp://zdvserver?artcc=ZDV&amp;radar=DBL"
type="atcbi6"  elev="11786.68"  genStaticMsgs="0"  psrMaxRange="225"  scantime="12.12"
spos="+39:26:39.41,-106:54:10.21"  ssrMaxRange="225">
      <brtqc  acps="2080"  alt="839"  bcn="7777"  range="1"/>
      <srtqc  acps="32"  range="1"  runlength="64"/>
      <parrot  acps="1442"  mode3a="1274"  modec="990"  range="37"/>
      <parrot  acps="1907"  mode3a="1275"  modec="800"  range="17.875"/>
    </radar>
...
    <radar  name="cos-term"  device="sirs16s1:/dev/asr11-5"  type="asr11"  elev="6280.00"
genStaticMsgs="0"  magdev="9.00"  psrMaxRange="60"  psrRangeUnits="64.00"  scantime="4.84"
spos="+38:48:02.10,-104:40:42.50"  ssrMaxRange="60"  ssrRangeUnits="64.00">
      <brtqc  acps="2104"  alt="0"  bcn="7777"  range="59.1"/>
      <srtqc  acps="175"  range="56"  runlength="24"/>
      <parrot  acps="3473"  mode3a="1274"  modec="600"  range="46.2"/>
    </radar>
    <radar  name="dbl-term"  device="sirs16s1:/dev/lrr8"  type="lrr"  elev="11779.00"
psrMaxRange="1"  scantime="12.00"  spos="+39:26:39.40,
-106:54:10.20"  ssrMaxRange="250">
      <brtqc  acps="2080"  alt="839"  bcn="7777"  range="1"/>
      <srtqc  acps="32"  range="1"  runlength="24"/>
      <parrot  acps="1442"  mode3a="1274"  modec="990"  range="36.9"/>
      <parrot  acps="1907"  mode3a="1275"  modec="800"  range="17.9"/>
    </radar>
    <radar  name="den-term"  device="sirs16s1:/dev/srr0"  type="asr9-modes"  elev="5441.00"
magdev="8.00"  psrMaxRange="60"  scantime="4.62"  spos="+39:51:16.80,-104:43:05.90"
ssrMaxRange="60">
      <brtqc  acps="2102"  alt="-10"  bcn="7770"  range="59.1"/>
```

```
    <srtqc  acps="1036"  range="45"  runlength="24"/>
    <parrot  acps="1679"  mode3a="0305"  modec="-2"  range="27.7"/>
    <parrot  acps="3496"  mode3a="0306"  modec="-2"  range="11.6"/>
   </radar>
...
   <svol  name="zdvasv"  genStaticMsgs="0"  sac="0xc1"  sic="0x11"  svType="0">
    <stream  name="uat"  device="multi://239.161.17.32:48040"/>
    <stream  name="1090"  device="multi://239.161.17.32:48041"/>
    <stream  name="equip"  device="multi://239.161.17.32:48042"/>
    <stream  name="svol"  device="multi://239.161.17.32:48043"/>
    <stream  name="sdp"  device="multi://239.161.17.32:48044"/>
    <radio_station  name="RSXXZDV"  lid="4170"  maxRange="150.00"  spos="+44:49:08.71,-
110:33:28.45">
      <receiver  id="0xdf7e0"  icao="0xfaafaa"  period="10.0"  spos="+44:49:08.71,-
110:33:28.45"  uat="0"/>
      <receiver  id="0xdf7e1"  icao="0xfaafaa"  period="10.0"  spos="+44:49:08.71,-
110:33:28.45"  uat="0"/>
      <receiver  id="0xdf7e2"  icao="0xfaafaa"  period="10.0"  spos="+44:49:08.71,-
110:33:28.45"  uat="0"/>
      <receiver  id="0xdf7e3"  icao="0xfaafaa"  period="10.0"  spos="+44:49:08.71,-
110:33:28.45"  uat="0"/>
      <receiver  id="0x9f7e0"  icao="0xfaafaa"  period="5.0"  spos="+44:49:08.71,-
110:33:28.45"  uat="1"/>
    </radio_station>
...
   </svol>
   <svol  name="rdc-ads"  genStaticMsgs="0"  sac="0xc2"  sic="0x19"  svType="1">
    <stream  name="uat"  device="(sirs16@multi:eth2:239.162.25.32/59950+
sirs16@multi:eth3:239.162.25.32/59950)"/>
    <stream  name="1090"  device="(sirs16@multi:eth2:239.162.25.32/59951+
sirs16@multi:eth3:239.162.25.32/59951)"/>
    <stream  name="equip"  device="(sirs16@multi:eth2:239.162.25.32/59952+
sirs16@multi:eth3:239.162.25.32/59952)"/>
    <stream  name="svol"  device="(sirs16@multi:eth2:239.162.25.32/59953+
sirs16@multi:eth3:239.162.25.32/59953)"/>
    <stream  name="sdp"  device="(sirs16@multi:eth2:239.162.25.32/59954+
sirs16@multi:eth3:239.162.25.32/59954)"/>
    <radio_station  name="38A"  disabled="0"  lid="2500"  maxRange="60.00"
spos="+37:54:42.87,-103:59:04.28">
      <receiver  id="0xd0650"  icao="0xfaafaa"  period="10.0"  spos="+37:54:42.87,-
103:59:04.28"  uat="0"/>
```

```
      <receiver  id="0xd0651"  icao="0xfaafaa"  period="10.0"  spos="+37:54:42.87,-
103:59:04.28"  uat="0"/>
      <receiver  id="0xd0652"  icao="0xfaafaa"  period="10.0"  spos="+37:54:42.87,-
103:59:04.28"  uat="0"/>
      <receiver  id="0xd0653"  icao="0xfaafaa"  period="10.0"  spos="+37:54:42.87,-
103:59:04.28"  uat="0"/>
      <receiver  id="0x90650"  icao="0xfaafaa"  period="5.0"  spos="+37:54:42.87,-
103:59:04.28"  uat="1"/>
    </radio_station>
...
   </svol>
   <nexradServer  name="zdv"  nexradOrigin="+33:00:00.00,-114:00:00.00"
port="%{ZDV_NEXRAD_SERVER_PORT}"  tangent="+41:11:51.00,-106:27:55.00"/>
  </sources>
  <ecgp  name="zdvserver"  device="multi://239.255.1.50:48020">
   <artcc  name="ZDV">
    <radar  name="CDC"  id="2"  ecgpType="1"/>
    <radar  name="DBL"  id="12"  ecgpType="1"/>
...
   </artcc>
  </ecgp>
```

8.2.11. En Route Radar for ERAM Test Bed

For the ERAM Test Bed, En Route and terminal radar sites and static messages are defined in the SDRR surveillance configuration file for each En Rroute facility:

```
<root>
  <sources>
   <radar  name="cdc-eram"  device="(ecgrdr://pipa?device=CDC-1&amp;device=CDC-2
&amp;device=CDC-3+ecgrdr://pipb?device=CDC-1&amp;device=CDC-2&amp;
device=CDC-3)"  type="arsr2"  elev="10786.21"  psrMaxRange="225"  scantime="12.00"
spos="+37:35:35.48,-112:51:49.20"  ssrMaxRange="225">
    <brtqc  acps="2080"  alt="-12"  bcn="0000"  range="1"/>
    <srtqc  acps="32"  range="1"  runlength="64"/>
    <parrot  acps="887"  mode3a="1274"  modec="10"  range="111.5"/>
    <parrot  acps="3149"  mode3a="1275"  modec="4087"  range="97.125"/>
   </radar>
```

```
   <radar  name="dbl-eram"  device="(ecgrdr://pipa?device=DBL-1&amp;device=DBL-2
&amp;device=DBL-3+ecgrdr://pipb?device=DBL-1&amp;device=DBL-2&amp;
device=DBL-3)"  type="atcbi6"  elev="11786.68"  psrMaxRange="225"  scantime="12.12"
spos="+39:26:39.41,-106:54:10.21"  ssrMaxRange="225">
     <brtqc  acps="2080"  alt="839"  bcn="7777"  range="1"/>
     <srtqc  acps="32"  range="1"  runlength="64"/>
     <parrot  acps="1442"  mode3a="1274"  modec="990"  range="37"/>
     <parrot  acps="1907"  mode3a="1275"  modec="800"  range="17.875"/>
   </radar>
...
   <radar  name="cos-term"  device="sirs16s1:/dev/asr11-5"  type="asr11"  elev="6280.00"
magdev="9.00"  psrMaxRange="60"  psrRangeUnits="64.00"  scantime="4.84"
spos="+38:48:02.10,-104:40:42.50"  ssrMaxRange="60"  ssrRangeUnits="64.00">
     <brtqc  acps="2104"  alt="0"  bcn="7777"  range="59.1"/>
     <srtqc  acps="175"  range="56"  runlength="24"/>
     <parrot  acps="3473"  mode3a="1274"  modec="600"  range="46.2"/>
   </radar>
   <radar  name="dbl-term"  device="sirs16s1@/dev/lrr8"  type="lrr"  elev="11779.00"
psrMaxRange="1"  scantime="12.00"  spos="+39:26:39.40,
-106:54:10.20"  ssrMaxRange="250">
     <brtqc  acps="2080"  alt="839"  bcn="7777"  range="1"/>
     <srtqc  acps="32"  range="1"  runlength="24"/>
     <parrot  acps="1442"  mode3a="1274"  modec="990"  range="36.9"/>
     <parrot  acps="1907"  mode3a="1275"  modec="800"  range="17.9"/>
   </radar>
   <radar  name="den-term"  device="sirs16s1@/dev/srr0"  type="asr9-modes"  elev="5441.00"
magdev="8.00"  psrMaxRange="60"  scantime="4.62"  spos="+39:51:16.80,-104:43:05.90"
ssrMaxRange="60">
     <brtqc  acps="2102"  alt="-10"  bcn="7770"  range="59.1"/>
     <srtqc  acps="1036"  range="45"  runlength="24"/>
     <parrot  acps="1679"  mode3a="0305"  modec="-2"  range="27.7"/>
     <parrot  acps="3496"  mode3a="0306"  modec="-2"  range="11.6"/>
   </radar>
...
   <svol  name="zdvasv"  pos="+44:49:08.71,-110:33:28.45"  sac="0xc1"  sic="0x11"
svType="0">
     <stream  name="uat"  device="multi://239.161.17.32:59950"/>
     <stream  name="1090"  device="multi://239.161.17.32:59951"/>
     <stream  name="equip"  device="multi://239.161.17.32:59952"/>
     <stream  name="svol"  device="multi://239.161.17.32:59953"/>
     <stream  name="sdp"  device="multi://239.161.17.32:59954"/>
```

```
     <radio_station  name="RSXXZDV"  lid="4170"  maxRange="150.00"  spos="+44:49:08.71,-
110:33:28.45">
       <receiver  id="0xdf7e0"  icao="0xfaafaa"  period="10.0"  spos="+44:49:08.71,-
110:33:28.45"  uat="0"/>
       <receiver  id="0xdf7e1"  icao="0xfaafaa"  period="10.0"  spos="+44:49:08.71,-
110:33:28.45"  uat="0"/>
       <receiver  id="0xdf7e2"  icao="0xfaafaa"  period="10.0"  spos="+44:49:08.71,-
110:33:28.45"  uat="0"/>
       <receiver  id="0xdf7e3"  icao="0xfaafaa"  period="10.0"  spos="+44:49:08.71,-
110:33:28.45"  uat="0"/>
       <receiver  id="0x9f7e0"  icao="0xfaafaa"  period="5.0"  spos="+44:49:08.71,-
110:33:28.45"  uat="1"/>
     </radio_station>
...
   </svol>
   <svol  name="rdc-ads"  sac="0xc2"  sic="0x19"  svType="1">
     <stream  name="uat"  device="(sirs16@multi:eth2:239.162.25.32/59950+
sirs16@multi:eth3:239.162.25.32/59950)"/>
     <stream  name="1090"  device="(sirs16@multi:eth2:239.162.25.32/59951+
sirs16@multi:eth3:239.162.25.32/59951)"/>
     <stream  name="equip"  device="(sirs16@multi:eth2:239.162.25.32/59952+
sirs16@multi:eth3:239.162.25.32/59952)"/>
     <stream  name="svol"  device="(sirs16@multi:eth2:239.162.25.32/59953+
sirs16@multi:eth3:239.162.25.32/59953)"/>
     <stream  name="sdp"  device="(sirs16@multi:eth2:239.162.25.32/59954+
sirs16@multi:eth3:239.162.25.32/59954)"/>
     <radio_station  name="38A"  lid="2500"  maxRange="60.00"  spos="+37:54:42.87,-
103:59:04.28">
       <receiver  id="0xd0650"  icao="0xfaafaa"  period="10.0"  spos="+37:54:42.87,-
103:59:04.28"  uat="0"/>
       <receiver  id="0xd0651"  icao="0xfaafaa"  period="10.0"  spos="+37:54:42.87,-
103:59:04.28"  uat="0"/>
       <receiver  id="0xd0652"  icao="0xfaafaa"  period="10.0"  spos="+37:54:42.87,-
103:59:04.28"  uat="0"/>
       <receiver  id="0xd0653"  icao="0xfaafaa"  period="10.0"  spos="+37:54:42.87,-
103:59:04.28"  uat="0"/>
       <receiver  id="0x90650"  icao="0xfaafaa"  period="5.0"  spos="+37:54:42.87,-
103:59:04.28"  uat="1"/>
     </radio_station>
...
   </svol>
```

```
   <nexradServer  name="zdv"  nexradOrigin="+33:00:00.00,-114:00:00.00"
port="%{ZDV_NEXRAD_SERVER_PORT}"  tangent="+41:11:51.00,-106:27:55.00"/>
  </sources>
</root>
```

8.2.12. Connections File for EIB

For the EIB, a connections file is used to allow SDRR to relay the IFDT messages from ERAM to the STARS system:

```
<connections>
  <connection>
    <interfacility  device="hgi://%{SDRR_ZDV}?eram=ZDV&amp;device=RDC"  txclock="2400"
rxclock="2400"/>
    <interfacility  device="sirs16@/dev/if0"  txclock="2400"  rxclock="2400"/>
  </connection>
</connections>
```

8.2.13. Connections File for ERAM Test Bed

For the ERAM Test Bed, a connections file is used to allow SDRR to relay the IFDT messages from ERAM to the STARS system:

```
<connections>
  <connection>
    <interfacility  device="(ecgif:pipa/RDC+ecgif:pipb/RDC)"  txclock="0"  rxclock="0" />
    <interfacility  device="sirs16@/dev/if0"  txclock="2400"  rxclock="2400" />
    </connection>
</connections>
```

8.2.14. TFDM Emulation

SDRR can simulate other data sent to TBFM in all simulation modes. This includes Terminal Flight Data Manager (TFDM) Release Time Coordination Service (RTCS) and Terminal Publication (TTP) and Metering Information Service (MIS) via System-Wide Information Management (SWIM) NAS Enterprise Messaging System (NEMS). Note that the “tfdm” configuration attribute “cmsInput” must match the “clientDevice” value in the “eramsim” stanza.

8.2.14.1. RTCS

```
<tfdm artcc="ZLA" cmsInput="pipe:zla-cms-%{USER}">
  <rtcs name="klax"
    publishDevice="solace://tbfmsolacedev01:55003?queueName=
RTCSPublish_08&amp;compressed&amp;userName=solace&amp;passwd=solace1&amp;
vpn=TBFM_SW_IS_DEPLOYMENT"
    requestDevice="solace://tbfmsolacedev01:55003?queueName=
RTCSRequest_08&amp;compressed&amp;userName=solace&amp;passwd=solace1&amp;
vpn=TBFM_SW_IS_DEPLOYMENT"
    retransmitTimeout="5"               <!-- specified in seconds -->
    maxRetries="5"
    delayReceiptAck="1"                 <!-- specified in seconds -->
    unsolicitedReconWaitTime="10"       <!-- specified in seconds -->
    disableReceiptAck="1"
    hbPeriod="6"                        <!-- specified in seconds -->
    disableHBs="0"
    userName="TFDM"
    version="2.0.0"
    dpt="KLAX"/>
</tfdm>
```

8.2.14.2. TTP

```
<tfdm artcc="ZLA" cmsInput="pipe:zla-cms-%{USER}">
  <ttp name="klax"
  device="solace://tbfmsolacedev01:55003?compressed&amp;username=solace&amp;passwd=solace1
  &amp;vpn=TBFM_SW_IS_DEPLOYMENT_DMR_2"
  cmsInput="pipe:klax-cms-%{USER}"
  hbPeriod="6"                          <!-- specified in seconds -->
  disableHBs="0"
  resyncPeriod="00:05:00"
  disableResync="0"
  version="2.0.1"
  resyncInterval="10.0"                 <!-- specified in seconds -->
  resyncMsgsPerLoop="100"
  startDelay="00:00:00"
   departureRunway="RW24L"              <!-- Only if defined in activeRunways.xml -->
   fcdtOffset="00:00:00"
   flightAddDelay="6"                   <!-- specified in seconds -->
   initialGateTodOffset="00:12:00"/>
</tfdm>
```

8.2.14.3. MIS

```
<tfdm artcc="ZLA" cmsInput="pipe:zla-cms-%{USER}">
  <mis device="solace://tbfmsolacedev01:55003?queueName=MIS_08
&amp;compressed&amp;userName=solace&amp;passwd=solace1&amp;
vpn=TBFM_SW_IS_DEPLOYMENT"/>
</tfdm>
```

9. Exported SDRR Scenario

Exported scenarios are custom user created scenario that have been scripted in the Graphic Simulation Generation Tool (GSGT) and exported for SDRR injection.

9.1. Scenario File

The primary component of an exported scenario is called the scenario file and is typically named sdrr.xml. This file defines parameters for the scenario and all of the other components of the scenario to be read by SDRR.

9.1.1. Scenario File Format

sim

Defines options for running SDRR.

Required Attributes:

name

Name of the scenario.

Optional Attributes:

starttime

Specifies scenario start time.

earlyFPMargin

Specifies the time flight plans are going to be injected.

sysSetupDelay

Specifies the time when system commands will be injected. Specifies the time when system commands will be injected.

tgtIntraMsgDelay

Specifies the time when messages between HOST to HOST will be injected.

rsiList

Specifies a list of RSI-tagged messages to be injected.

logfile

Specifies the location of the SDRR log file.

comments

Freeform scenario comments.

radar

Defines radar file inputs.

Required Attributes:

name

Name of the radar.

srv

Specifies the location of the binary radar file.

svol

Defines ADSB file inputs.

Required Attributes:

name

Name of the radio station.

ast

Specifies the location of the binary ads file.

wam

Defines radar file inputs.

Required Attributes:

name

Name of the wam radio station.

ast

Specifies the location of the binary wam file.

msgs

Defines message file inputs.

Required Attributes:

file

Specifies the location of the msgs file.

Optional Attributes:

facility

Specifies the facility used in the msgs file.

tracks

Defines an SDRR track file. The track file is normally generated by GSGT and is used by SDRR in generating TU messages. If no file is specified, SDRR will extrapolate the position information from the TI message.

Required Attributes:

file

Specifies the location of the tracks file.

tgtctl

Defines customized responses to specified messages.

Required Attributes:

file

Specifies the location of the tgtctl file.

9.1.2. Example Scenario File

```
<sim earlyFPMargin="00:05:00" name="example" starttime="00:00:00" sysSetupDelay="00:00:05"
tgtIntraMsgDelay="00:00:00">
   <radar file="terminalSrv/dov.srv" name="dov"/>
   <radar file="terminalSrv/dox.ast" name="dox"/>
   <radar file="terminalSrv/nxy.srv" name="nxy"/>
   <radar file="terminalSrv/phl.srv" name="phl"/>
   <radar file="terminalSrv/qie.srv" name="qie"/>
   <radar file="terminalSrv/wri.srv" name="wri"/>
   <mlat file="terminalSrv/mlt.ast" name="mlt"/>
   <svol name="ppp-ads">
      <stream file="terminalADSB/ppp-ads-uat.ast" name="uat"/>
      <stream file="terminalADSB/ppp-ads-1090.ast" name="1090"/>
      <stream file="terminalADSB/ppp-ads-equip.ast" name="equip"/>
      <stream file="terminalADSB/ppp-ads-svol.ast" name="svol"/>
      <stream file="terminalADSB/ppp-ads-sdp.ast" name="sdp"/>
   </svol>
   <wam name="wamsvg">
      <stream file="wam/wamsvg-modeS.ast" name="modeS"/>
      <stream file="wam/wamsvg-1090.ast" name="1090"/>
      <stream file="wam/wamsvg-uat.ast" name="uat"/>
      <stream file="wam/wamsvg-atcrbs.ast" name="atcrbs"/>
      <stream file="wam/wamsvg-svol.ast" name="svol"/>
      <stream file="wam/wamsvg-sdp.ast" name="sdp"/>
   </wam>
   <tracks file="tracks.xml"/>
   <msgs file="nonRadar/msgs.xml"/>
   <scriptDefinitions file="nonRadar/scriptDefinitions.xml"/>
</sim>
```

9.2. Messages File

The messages file is an XML file which defines the messages to be injected during the scenario playback.

9.2.1. Messages File Format

ifmsg

Defines an interfacility message.

Required Attributes:

src

Specifies the source facility of the message.

dest

Specifies the destination facility of the message.

time

Specifies the injection time of the message.

Optional Attributes:

acid

Specifies the target aircraft ID for the message. Any target-specific substitutions will be based on the acid. Messages with an acid will be retried up to 5 times.

literalMode

When enabled, message text will be injected exactly as it appears (no token substitution).

doField23

When set on a TI message, field23 will be auto-generated based on data in the tracks file.

Value:

text

Specifies the text of the message (varies according to message type).

fdiomsg

Defines a FDIO message.

Required Attributes:

src

Specifies the source facility of the message.

dest

Specifies the destination facility of the message.

time

Specifies the injection time of the message.

Optional Attributes:

acid

Specifies the target acid for the message. Any target-specific substitutions will be based on the acid. Messages with an acid will be retried up to 5 times.

literalMode

When enabled, message text will be injected exactly as it appears (no token substitution).

Value:

text

Specifies the text of the message (varies according to message type).

tcwMsg

Defines a TCW injector message.

Required Attributes:

fac

Specifies the facility of the message.

time

Specifies the injection time of the message.

position

Specifies the TCW position the message will be sent to.

Optional Attributes:

id

Specifies a target ACID on the TCW.

In all flight data message types, SDRR performs substitution of special tokens.

Special token substitutions in message text processed by SDRR:

@@@

Remote CID.

###

Local CID.

$hhmm$

Time substitution relative to scenario start time.

%hhmm%

Time substitution relative to current simulation time.

{TOD+hh:mm:ss}

Time substitution relative to flight plan time of departure.

9.2.2. Example Messages File

```
<cmds>
   <tod id="DEP01" sysSetup="1" time="-10:00:00">00:00:00</tod>
   <fdiomsg ccu="1" device="PHLD" host="zny" id="DEP01" kbnum="2" rcu="23"
time="00:00:00">FP DEP01 B737 1234 100 KPHL P%{TOD+00:00:00} 100 KPHL..KACY/0023 EQ
P R SRV EB1 WAK M FLT S FLR I OTH ++SUR/260B CODE/100001++</fdiomsg>
   <ifmsg dest="ppp" id="DEP01" injectorName="zny" time="00:00:03">FP %LCID%DEP01 PHL/B737
1234 PD4 P%{TOD+00:00:00} 100 </ifmsg>
   <ifmsg dest="zny" id="DEP01" injectorName="zny:ppp" time="00:00:06">DM %RCID%
%0000%</ifmsg>
   <ifmsg dest="zdc" id="DEP01" injectorName="zny" time="00:00:09">FP DEP01 B737 1234 100
3940N/07454W E%{TOD+00:11:20} 100 KPHL./.KACY/0023 EQP R SRV EB1 WAK
 M FLT S FLR I OTH ++SUR/260B CODE/100001++</ifmsg>
   <tcwMsg fac="ppp" id="DEP01" position="*"
time="00:09:20">[CLEAR][HND_OFF]OK[SPACE]C[SPACE]DEP01[ENTER]</tcwMsg>
</cmds>
```

9.3. Response Control File

The response control file is an xml file which customizes SDRR behavior in response to real-time simulated events.

9.3.1. Response Control File Format

response

Defines a custom message response.

Required Attributes:

facility

Specifies the facility that will be affected by the response control statement.

acid

Specifies the target acid for the statement.

mode

Specifies the message response for SDRR to take. One of the following must be specified: “NO_REPLY”, “DX”, “DR”, “LRM”, or “LAM”.

Optional Attributes:

time

Specifies the start time of the statement. Default is 0.

msgtype

Specifies the message type for which this response control is to be used. Examples are “FP”, “TI”, etc. Default is all message types.

cnt

Specifies the number of times this response control will be active. Specify -1 for “forever”. Zero is invalid. Default is 1.

rejcode

Specifies an optional code to be included in the DR message generated by SDRR. (Only used when **mode**=“DR”).

taCtl

Defines a customized TA response time.

Required Attributes:

facility

Specifies the facility that will be affected by the response control statement.

acid

Specifies the target acid for the statement.

Optional Attributes:

delta

Specifies the time in seconds for an auto-TA response. Zero means immediate. -1 means never. Default is 0 (immediate).

relayCtl

Defines custom relay (HNH) behavior when a TI is received with an “00” in field 16.

Required Attributes:

facility

Specifies the facility that will be affected by the response control statement.

acid

Specifies the target acid for the statement.

*Optional Attributes (*At least 1 of the following must be specified*):*

destFac

Specifies the facility to relay this message to.

initialController

The controller to be specified in field 71 of the DA to the TI (SDRR defaults to “22”).

finalController

Used for field 48 of the TA (SDRR defaults to “22”).

vfrCtl

Customizes the FP sent in response to a received VFR FP.

Required Attributes:

facility

Specifies the facility that will be affected by the response control statement.

acid

Specifies the target acid for the statement.

*Optional Attributes (*At least 1 of the following must be specified*):*

coordFix

Specifies field 6 of the return FP. (SDRR defaults to using the last fix of field 10 in the VFR FP).

bcn

Specifies field 4 of the return FP. (SDRR will auto-assign by default).

9.3.2. Example Response Control File

```
<tcrs>
   <!-- one DX to FP -->
   <response facility="zdc" acid="DPT01" msgType=”FP” mode="DX"  />

  <! – NO response to all msgs forever -->
   <response facility="zdc" time="00:02:19.00" acid="DPT01" mode=”NO_REPLY” cnt="-1" />

   <!-- auto-TA after 5 seconds -->
   <taCtl facility="zdc" time="00:10:00.00" acid="TEST01" delta="5” />

    <!-- do an immediate TA -->
   <taCtl facility="zdc" time="00:10:00.00" acid="TEST02" delta="0” />

   <! – relay a flight to ZOB -->
   <relayCtl facility="zdc" acid="HNH01" destFac=”zob” />

</tcrs>
```

Appendix A. Acronyms

ACID	Aircraft Identification
ADS-B	Automatic Dependent Surveillance – Broadcast
ARSR	Air Route Surveillance Radar
ARTCC	Air Route Traffic Control Center
ARTS	Automated Radar Terminal System
ASR	Airport Surveillance Radar
ASR-9	Airport Surveillance Radar Model-9
ASTERIX	All Purpose Structured Eurocontrol Radar Information Exchange
ATC	Air Traffic Control
AViD	Airspace Visualization Display
BRTQC	Beacon Real Time Quality Control
CAS	Commercially Available Software
CD	Common Digitizer
CDR	Continuous Data Recording
CMS	Common Message Set
DASI	Digital Altimeter Setting Indication System
DASR	Digital Airport Surveillance Radar
DYSIM	Dynamic Simulation
ECG	External Communications Gateway
ECGP	External Communications Gateway Protocol
EDDS	En Route Data Distribution System
ERAM	En Route Automation Modernization
ETMS	Enhanced Traffic Management System
FAA	Federal Aviation Administration
FDIO	Flight Data Input/Output
GSGT	Graphic Simulation Generation Tool

GUI	Graphical User Interface
IFDT	Interfacility Flight Data Transfer
Mode 3/A	Identification Reporting Mode of Secondary Radar
Mode C	Altitude Reporting Mode of Secondary Radar
Mode S	Mode Select Beacon System
MLAT	Multilateration
NAS	National Air Space
RAPPI	Random Access Plan Position Indicator
RSI	Record Select Indicator
RTQC	Real Time Quality Control
SAC	System Area Code
SDRR	Simulation Driver Radar Recorder
SIC	System Identity Code
SIRS	STARS Interfacility and Radar Simulator
SSRV	Simulation Services
STARS	Standard Terminal Automation Replacement System
SWIM	System-Wide Information Management
TARP	Time-based Archive Recording Player
TBFM	Time Based Flow Management
TSIM	TBFM Simulation
TRACON	Terminal Radar Approach Control
WAM	Wide Area Multilateration
WJHTC	William J. Hughes Technical Center