RGS Version 36 Release Notes

Release Summary

In this release, we added support for the Smart Card class, added an option to import PKCS #12 files, added GTK cabinet commands for G2S 2.1, corrected issues and made several improvements to the Tester Toolkit.

New Features

RGS now supports the smart card class.

Since implementation of the smart card class requires the use of a proprietary Smart Card Application Protocol (SCAP), RadBlue has implemented the smart card class with its own version of a SCAP. This implementation demonstrates the smart card class between RST and RGS. Also, depending on your host or EGM implementation, there are three ways that you can use the RadBlue smart card:

1. If the EGM does not support SCAP, RGS can be the smart card device owner.
2. If EGM supports a proprietary SCAP implementation, RGS can be a guest of the smart card device.*
3. If the host supports no SCAP, RST can be used as the EGM.

See the RGS User Guide or **RGS Help** for information on using the RadBlue smart card implementation.

Your proprietary SCAP can be implemented in the tools as a custom project. Contact RadBlue for more information.

If you do not have RGS or RST, you can request a free student license. These two products communicate with each other by default, so set up and configuration is minimal.

Configuration Modifications

- A new **Import PKCS #12 File** option (**Tools** > **Configure** > **Security Options** > **Key Store**) lets you quickly import the certificates stored in a P12 or PFX file into the tool's **client.jks** and **trusted.jks**. All certificates in the PKCS #12 file are imported to client.jks. Only non-key-entry certificates are imported to trusted.jks. Once the certificates are successfully imported, they can be viewed from the Key Store tab.
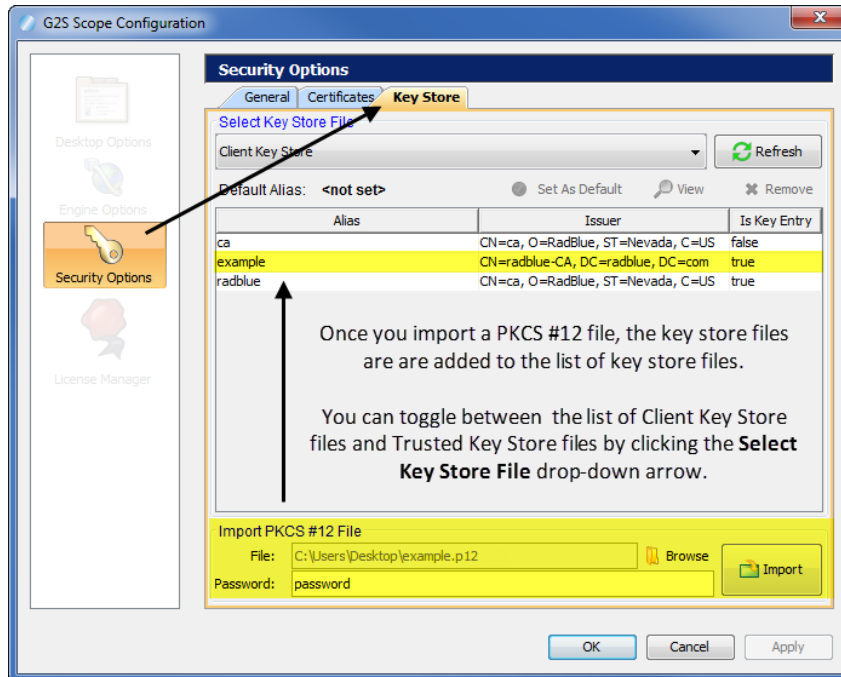

Engine Modifications

- In RGS 34, the `bonus.setBonusAward` command was sent as a G2S notification. This issue has been corrected.
- In previous versions, the Snapshot option ran slower than expected. Snapshot has been modified to improve performance.

User Interface Modifications

- Error messages that display in the **GSA Message Validator**, accessed through the Tools option on the menu bar, have been improved with clearer information when XML is invalid or the G2S message does not conform to the protocol.

- The performance of the Debug Console has been improved to display messages more efficiently, especially during long runs of the tool.

- When sending a command using the Custom Send option in the Send Command layout, the *timeToLive* attribute is now set to zero (0) by default. This change ensures that the command does not time-out in the time it takes you to make changes to it. You can change this value, as needed, when you are modifying the command.

- Attributes that have been deprecated in G2S 2.1 now display in a strike-through font in the Data Model Viewer (DMV).
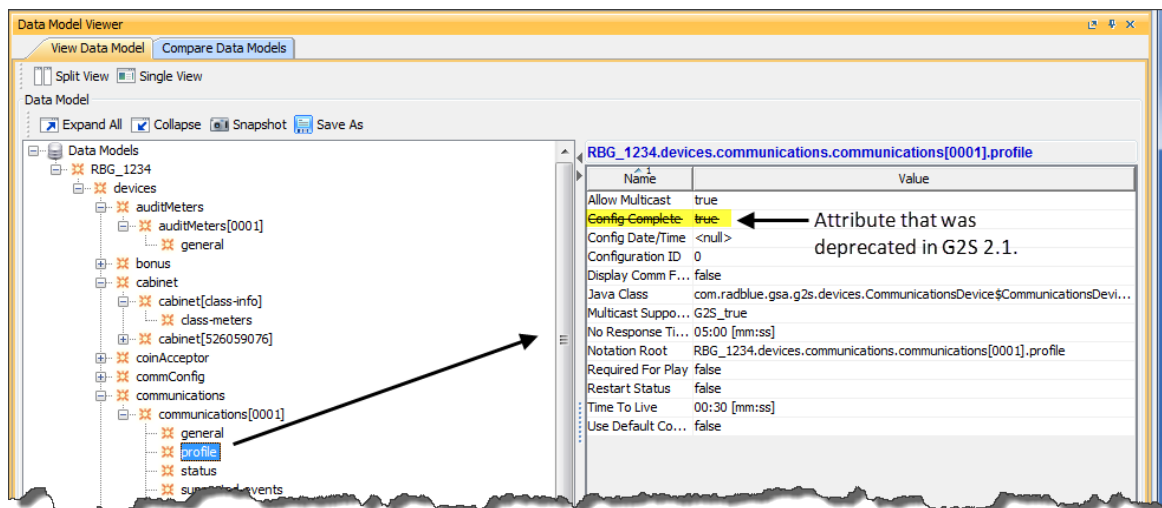

- The following cabinet class master reset commands have been added to the Send Command object for G2S 2.1:
  - `masterReset`
  - `authorizeMasterReset`
  - `cancelMasterReset`

- You can now quickly view the URLs for the optional RGS REST interface (Remote Control) and the Scratch Pad example interface by going to the **Engine** layout and clicking **RGS URLs**.
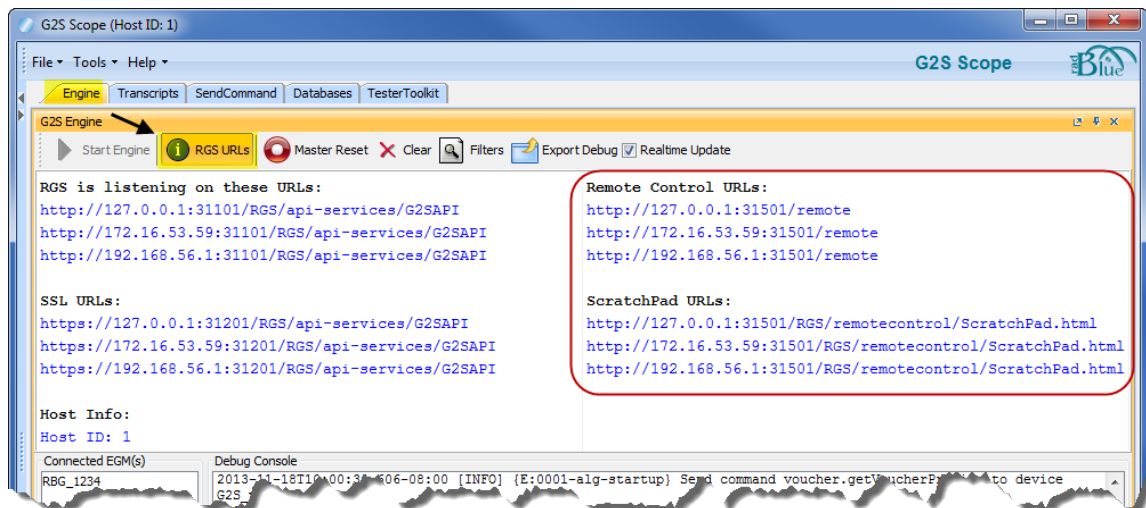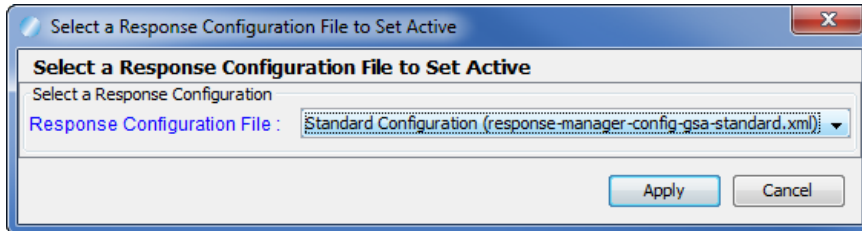

- When you send the `bonus.cancelBonusAward` from the Send Command layout, if the bonus state is pending, the bonus ID and transaction ID are now automatically retrieved from the latest log entry. Previously, this information was not available in the interface.
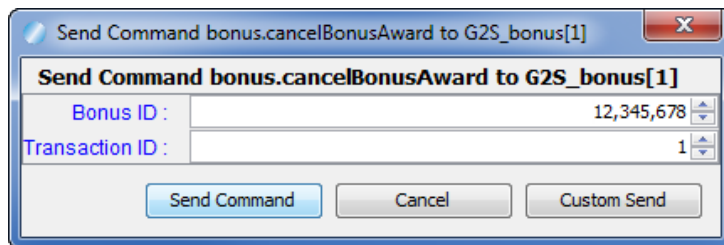
- The IP Ref ID value is now assigned automatically for the `informedPlayer.setIpMessage` command in the Send Command layout.

- The Configuration ID and Transaction ID values for the `commConfig.cancelCommConfig` command in the Send Command layout are now retrieved automatically just before the execution of the command from the latest log entry, if the status is Pending or Authorized.

Tester Toolkit Modifications

- A new **Set Active Response Configuration** script verb has been added to Custom Scripting. This verb lets you set a custom response from a custom script by selecting the response configuration you want to add.

  Once you add a response configuration file through the Response Manager, it becomes available in the Set Active Response Configuration drop-down.

- The `bonus.cancelBonusAward` command has been added to Custom Scripting. Note that, if the bonus state is pending, the bonus ID and transaction ID are automatically retrieved, just before the execution of the command, from the latest log entry.

- The Configuration ID and Transaction ID values in the following Custom Scripting commands are now retrieved automatically just before the execution of the command from the latest log entry, if the status is Pending or Authorized:
  - `commConfig.authorizeCommChange`
  - `commConfig.cancelCommChange`
  - `optionConfig.authorizeOptionChange`
  - `optionConfig.cancelOptionChange`
  - `optionConfig.getOptionChangeStatus`

- The Configuration ID value for the `commConfig.setCommChange` command in Custom Scripting is now assigned automatically when the script in which it has been inserted is run.

- A unique Script ID value for the `download.setScript` command in Custom Scripting is now assigned automatically by RGS.

- A unique Verification ID value for the `gat.runSpecialFunction` command in Custom Scripting is now assigned automatically by RGS.

- A unique IP Ref ID value for the `informedPlayer.setIpMessage` command in Custom Scripting is now assigned automatically by RGS.

- You can now add and edit components from the Custom Scripting interface for the following commands:
  - `eventHandler.setEventSub`
  - `meters.setMeterSub`

- `commConfig.setCommChange` (edit only)
- `optionConfig.setOptionChange` (edit only)
- `mediaDisplay.loadContent`

- You can now add and edit the **Run Macro** script verb from the Custom Scripting interface.

- A unique Override ID value is now automatically assigned for the following Custom Scripting commands:

  - `playerMulticast.setCountdownOverride`
  - `player.setCountdownOverride`

- RGS now uses the current override ID from the latest log entry for the following commands:

  - `player.setCarryOver`
  - `player.setHostPoints`
  - `player.setPlayerOverride`
  - `player.setPointBalance`

Transcript Modifications

- An issue in which the **Clear Display** option in the Message Transcript cleared the transcript display as well as the EGM selector list has been corrected. The **Clear Display** option now clears the transcript messages from the table only. The **Clear Database** option removes all transcript records from the transcript database, clears transcript messages from the table and clears the EGM selector list.

- Attributes that have been deprecated in G2S 2.1 now display in a strike-through font in the Message Transcript detail screen's Command view. The command details screen can be accessed by double-clicking any message in the Message Transcript.

- Comments associated with a message are now displayed next to the Command and View XML tabs when you view the message details as well as in the Message Transcript object.

radblue

Release Summary

In this release, we added a Use Existing License feature to the RGS installer, added and modified several commands in Custom Scripting and made general improvements to the tool.

Did you know....

You can see how the EGM's data model changes during testing through the Data Model Viewer. The Data Model Viewer uses commands sent from the EGM to construct a data model of the EGM that is modified as additional commands are received. You can snapshot the data model at different points in the testing process to see how certain activity changes the data model. For more information, see "About the Data Model Viewer" in RGS Help or the RGS User Guide.

Installation Modifications

- For version 34 and higher, if you install a version of RGS over an existing version, you now have the option of using the existing license. If you do not want to use the existing license, you can browse to a new license. Note that this option is only available when you install RGS over a previous installation. All components of the previous installation are removed by the installer except the license file and any backup files.
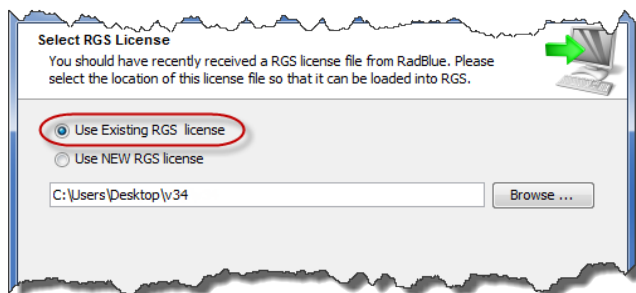
Configuration Modifications

- A new security option has been added that lets you fine-tune certificate requirements for the tool. Select **Require Client Certificate** if the other endpoint *must* have a certificate or it fails authentication. If this option is cleared, the other endpoint is not asked to send its client certificate. By default, this option is selected.

User Interface Modifications

- An error on the Send Command `cabinet.setOperatingHours` user interface has been corrected. The value entered in the Time Stamp field now matches the time value sent in the command.

- The `player.setPlayerOverride` user interface in the Send Command panel has been modified so that the default Transaction ID value is now taken from the last log entry. If there is no log entry, the value is set to 1.

- The `player.setPointBalance` user interface in the Send Command panel has been modified so that the default Transaction ID value is now taken from the last log entry. If there is no log entry, the value is set to 1.

Tester Toolkit Modifications

- The `bonus.setBonusAward` command in Custom Scripting has been enhanced to include all of the fields and options available through the Send Command layout.

- In Custom Scripting, you can now set the Device ID and Configuration ID values for the following commands:
  - `commConfig.authorizeCommChange`
  - `optionConfig.authorizeOptionChange`
- Custom Scripting now supports the following commands:
  - `bonus.setBonusAward`

- `bonusMulticast.setBonusAward`

- `bonusMulticast.setBonusMessage`

- `cabinet.getMasterResetStatus`

- `commConfig.cancelCommChange`

- `eventHandler.getEventSub` with wildcards

- `eventHandler.clearEventSub` with wildcards

- `informedPlayer.setIpGameDelay`

- `optionConfig.cancelOptionChange`

- `player.setCarryOver`

- `player.setHostPoints`

- `player.setPlayerOverride`

- `player.setPointBalance`

- `playerMulticast.setCountdownOverride`

- `playerMulticast.setPlayerMessage`

- The **Request ID** field in the `wat.initiateRequest` command in Custom Scripting has been removed. This value is now automatically generated by RGS and is always unique.

- The **Expire Date/Time** field in the `wat.initiateRequest` command and `bonus.setBonusAward` command in Custom Scripting is now configured as a relative value.
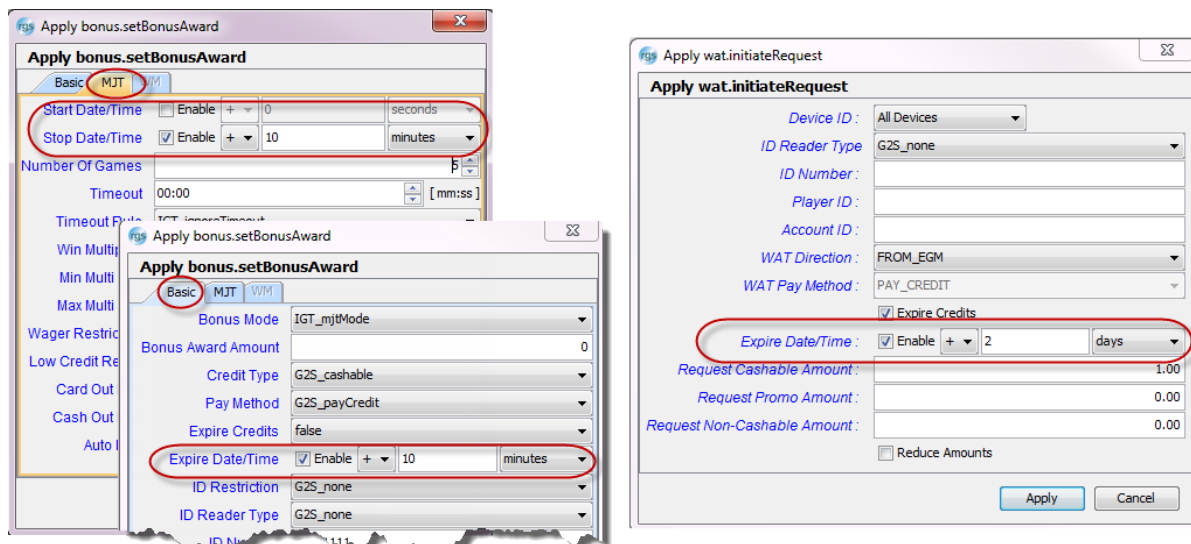
Advanced Transcript Analyzer Modifications

- The Summary version of the Advanced Transcript Analysis Report now displays one incident of each error type for each communications session.

Transcript Modifications

- On the Command tab for the `cabinet.setOperatingHours` command and the `cabinet.operatingHoursList` command, accessed by double-clicking that command in the Message Transcript, a standard Time value is now displayed instead of a Time value in milliseconds.

Release Summary

In this release, we updated the required Java version, added support for the Hardware class, added support for several commands in the Tester Toolkit, added the ability to configure a guest host through the `commConfig.setCommChange` command user interface and corrected minor issues.


Did you know....

Using the Tester Toolkit module's Custom Scripting feature, you can automate the testing of the Media Display class. When you use Media Display with Custom Scripting, remember that the Content ID must be the same in the Media Component file used in the custom script, in the `mediaDisplay.setActive` command and the `mediaDisplay.releaseActiveContent` command.

New Features

- *New for 2.1!* RGS now supports the **Hardware** class.
    - The `hardware.getHardwareDevices` and `hardware.getHardwareStatus` commands have been added to the Send Command object.
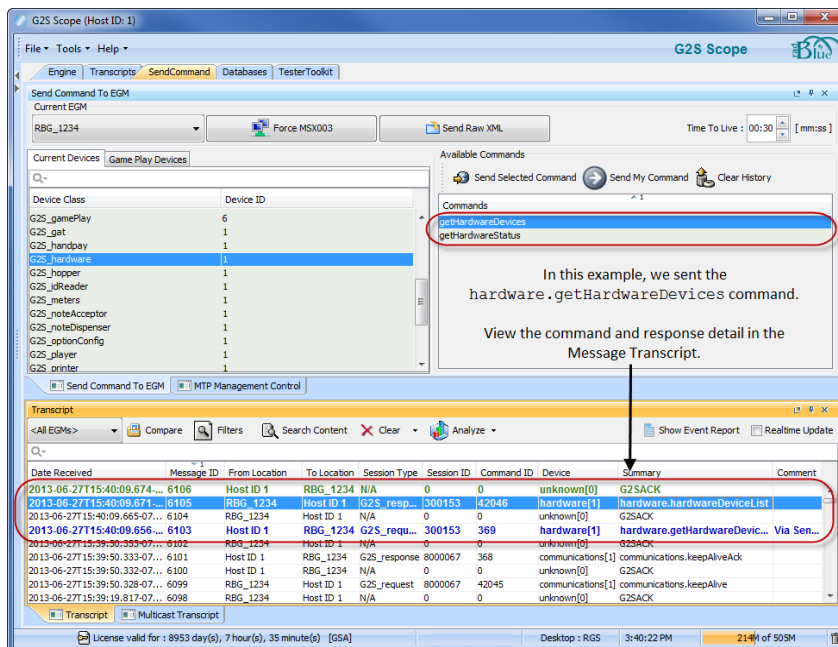
- You can view the hardware command details in the Message Transcript.
- The Hardware class can be used in the following Tester Toolkit features:
    - Custom Scripting
    - Startup Algorithm
    - Components (Macros and Event Subscriptions)
- The Hardware class has been added to the Data Model Viewer (DMV).
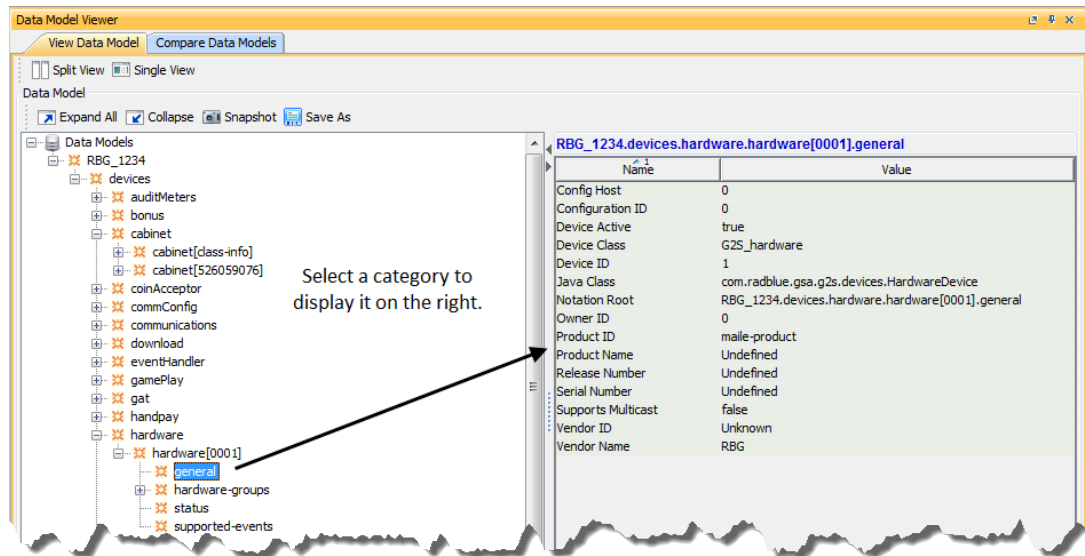

Installation Modifications

- All RadBlue products now require **Java SE 7u25** or higher. If the installer cannot find this version on your computer, you are prompted to download and run the Java installer, *or* navigate to the correct Java version. This update should have minimal effect on your environment, but keeps our products on modern versions of the underlying technology.

User Interface Modifications

- The **Data Model Viewer (DMV)** now displays times in the format defined in the **Default Format for Milliseconds Value** configuration option (**Tools** > **Configure** > **Engine** > **General**). The default is **(mm:ss)**.

- You can now type the Time Stamp when editing the `cabinet.setOperatingHours` command (through either the Send Command or Custom Scripting) rather than using the up/down arrows.
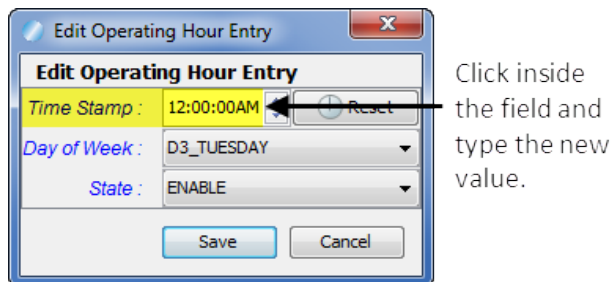
- An issue with adding a guest host through the user interface when sending the `commConfig.setCommChange` command has been corrected.

  To add one or more guest hosts to a device:

  1. Go to: **Send Command** > **G2S_commConfig** > **setCommChange**

  2. Click **Devices**.

  3. Select the device you want to edit, and click **Edit**.

  4. Click the drop-down arrow. The available guest hosts displays.

  5. Select the guest host(s) you want to add.

  6. Click **Save**.
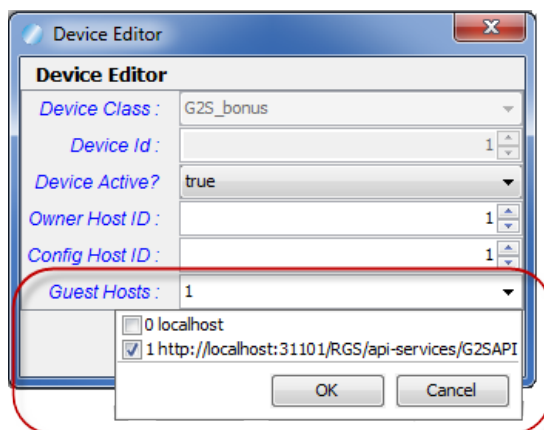
- An **Access Token** field has been added to the `mediaDisplay.loadContent` command user interface (**Send Command > IGT_mediaDisplay > loadContent**). This field lets you add the *mdAccessToken* attribute through the user interface. Note that this field automatically increments each time you send the `mediaDisplay.loadContent` command, so the Token Access value is always unique. A value of zero (0) disables the local connection.
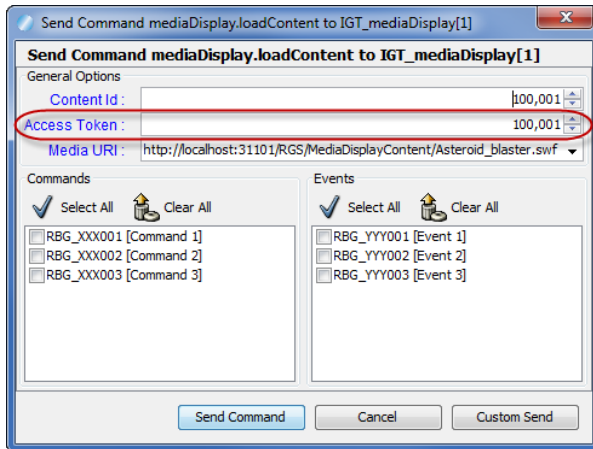

Tester Toolkit Modifications

Startup Algorithm

- The `cabinet.setOperatingHours` command has been added to the Startup Algorithm.

Custom Scripting

- The following commands have been added to Custom Scripting:
  - `cabinet.setOperatingHours`
  - `communications.closeComms`
  - `commConfig.setCommChange`
  - `download.authorizeScript`
  - `download.cancelScript`
  - `gamePlay.setActiveDenom`
  - `gat.getVerificationSupport`
  - `informedPlayer.setIpMessage`
 - `optionConfig.getOptionSeries`
  - `wat.initiateRequest`

Components

- A new **Set Comm Change** component has been added that lets you use pre-configured commConfig.setCommChange commands in Custom Scripts. To create a file for the Set Comm Change component, configure the `commConfig.setCommChange` command through the Send Command object, and then click **Save to Components**.
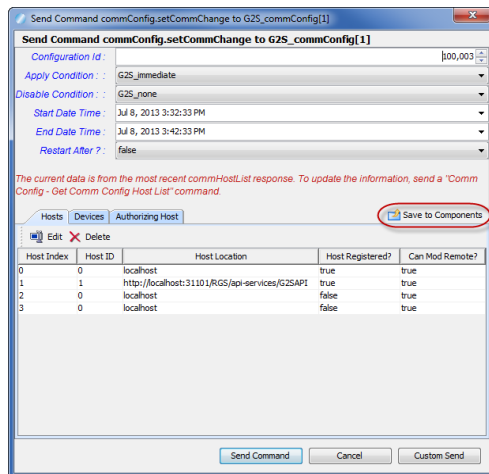

Once you have saved a `commConfig.setCommChange` command, it displays in the Set Comm Change screen (**Tester Toolkit > Components**), and can be edited or deleted through that screen. If you don't see your new component in the list, press the **Refresh** button on the Set Comm Change screen.

- An **Access Token** field has been added to the **Edit Media Display Load Content File** user interface (**Tester Toolkit > Components > Media Load Content > Edit** or **Add**). This field lets you edit the *mdAccessToken* attribute through the user interface. Note that this field automatically increments each time you send the `mediaDisplay.loadContent` command, so the Token Access value is always unique. A value of zero (0) disables the local connection.
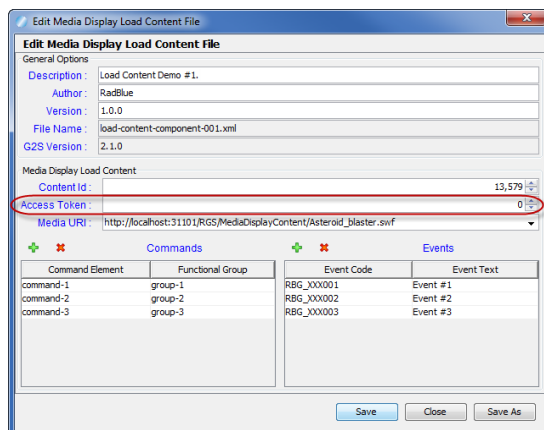
Transcript Modifications

- An issue in which no information displayed in the Command tab on the `handpay.handPayProfile` command detail screen has been corrected. Command tab information now displays correctly for this command.

- In the Message Transcript, the Time column value for the detail view of the `cabinet.operatingHoursList` command now displays in military time.
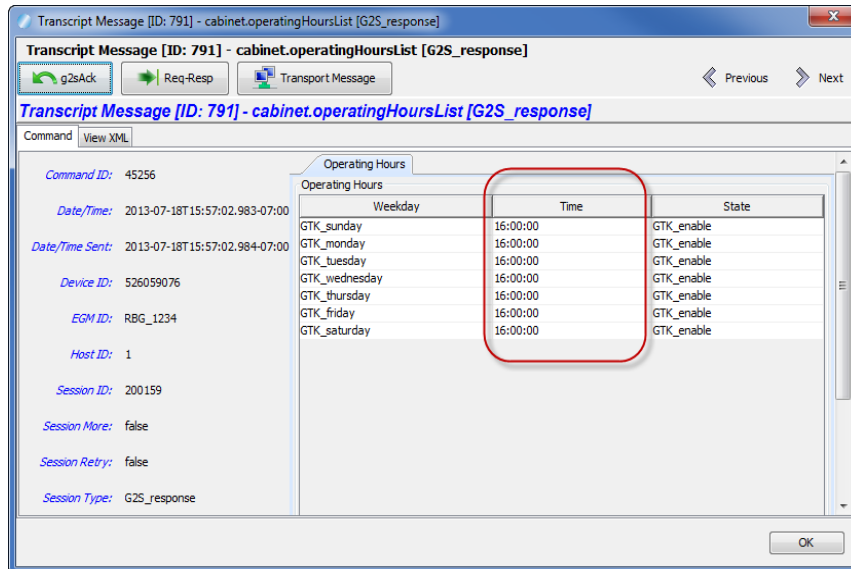
- You can now view multiple EGMS at once in the Message Transcript (rather than one or all) by holding down the CTRL button and selecting the EGMs you want to view.
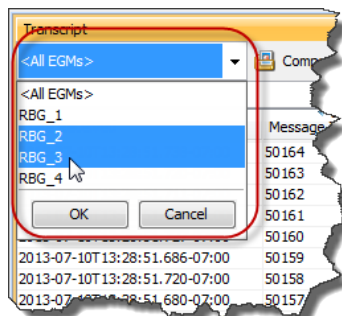
Release Summary

In this release, we updated the required Java version, added a dateTime offset option, updated the `gameplay.setActiveDenoms` command for 2.1, made changes to the Data Model Viewer for 2.1, added a Pause Script verb to the Tester Toolkit Startup Algorithm, updated the RGS REST Programming Interface Guide and corrected some minor issues.

Did you know....

4 GB is now the **minimum** Memory requirement for computers running RGS.

Installation Modifications

- All RadBlue products now require **Java SE 7u17** or higher. If the installer cannot find this version on your computer, you are prompted to download and run the Java installer, *or* navigate to the correct Java version. This update should have minimal effect on your environment, but keeps our products on modern versions of the underlying technology.

- The RGS installer now displays a warning if the license you load during installation is not valid.

## Configuration Modifications

- **Current RGS Offset [HH:MM:SS.MMM]** - Set the RGS time offset from the system clock, which is part of the *dateTime* calculation applied to all messages sent by RGS. The calculation for the *dateTime* stamp is:

  PC time + **offset** [Current RGS Offset value] + **setDateTime offset**

  Use this option if you want to test an incorrect *dateTime* value.

Engine Modifications

- Previously, RGS sent a `communications.setCommsState` command for every `communications.commsDisabled` command it received during start-up, which caused communications issues when RGS received *multiple* `communications.commsDisabled` commands. RGS no longer responds to every `communications.commsDisabled` command.

User Interface Modifications

- *New for 2.1!* **Apply Condition** and **Disable Condition** fields have been added to the **gamePlay.setActiveDenoms** screen (**Send Command** > **G2S_gameplay** > **setActiveDenoms**).
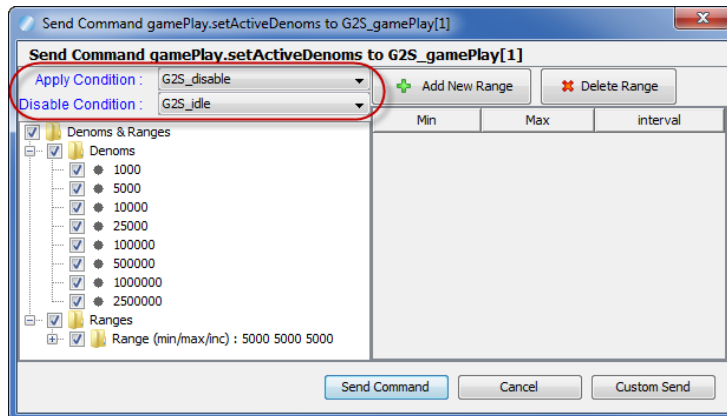
- *New for 2.1!* The **Data Model Viewer** now displays all version 2.1 status and profile attributes for the cabinet, gameplay and handpay classes.

- A display issue that occurred when a component was added to the Send Command **gat.doVerification** screen has been corrected.

- An issue in which a device guest host could not be modified through the Send Command **commConfig.setCommChange** screen has been corrected.

- Two new options, accessed through the File option on the menu bar, let you export all RGS configuration options and import those options into another instance of RGS. With these options, you can now quickly configure new instances of RGS.

  - **Export Configuration...** - Select to export all configuration settings for the tool, including security certificates. The resulting ZIP file can then be imported into another version of the same tool.

  - **Import Configuration...** - Select to import all configuration settings for the tool, exported from another version of the same tool, including security certificates. This option is used when you want to quickly set up a specific configuration for the tool that is already set up in another version of the tool.

Tester Toolkit Modifications

- A **Pause Script** verb has been added to the Tester Toolkit Startup Algorithm.
  The Pause Script verb pauses the script for the time, or within the time period, that you specify. Enter a Minimum Delay time and a Maximum Delay time in seconds. If the values are different, RGS pauses the script a random number of seconds between the minimum and maximum values. If both values are the same, RGS pauses the script for the number of seconds specified.

Remote Control Modifications

- A Scratch Pad demo program has been added to RGS Remote Control that assists you in developing an RGS REST interface by letting you see what an RGS REST interface might look like.
- The following information has been added to the [RGS REST Interface Programming Guide](#):

  - Sample use cases.
  - A `fetchStatus` method and an `insertTranscriptMarker` method, both for use with transcript records.
  - A "Getting Started" chapter to get you up and running with RGS Remote Control.
  - Information on how to use Scratch Pad.

Transcript Modifications

- If the tool receives a date/time with the seconds parsed greater than milliseconds, the time/date is truncated to milliseconds in the transcripts. For example:

  The date/time **2013-04-30T08:03:46.1234567890-07:00**

  displays as **2013-04-30T08:03:46.123-07:00** in the transcript.

  You can view the longer date/time format on the **XML** tab of the command object. To access the command object, double-click any message in the transcript.

- An issue in which the Message Transcript was reporting application errors incorrectly in the Summary column (for example, when sending an invalid attribute value in a message) has been corrected. The error is now noted correctly, for example:
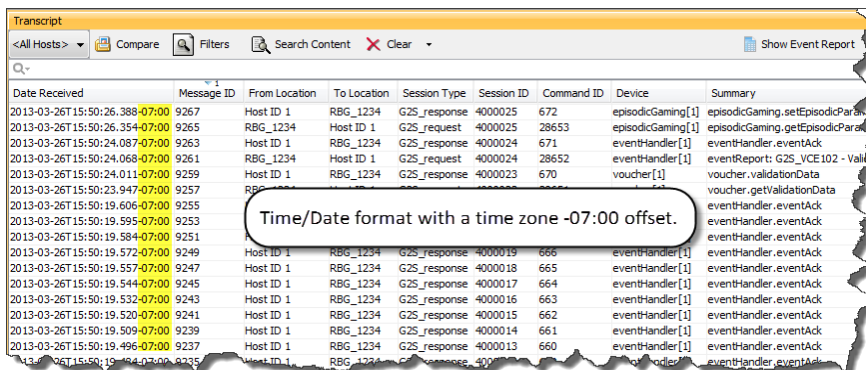 **optionConfig.error [G2S_OCX013, Invalid Value Selected For Option - G2S_none is invalid for eventSubscriptions]**

Release Summary

In this release, we added the ability to define the time zone and offset format for all messages sent from RGS, updated the required Java version, added Send My Command to the Tester Toolkit Startup Algorithm feature, and made several usability improvements and corrections.

New Features

- RST now supports time zone configuration for testing with custom time zones. Once configured, RST sends all messages using the defined time zone and offset format. When the selected time zone is Coordinated Universal Time (UTC), Zulu or Greenwich Mean Time (GMT), a "Z" appears at the end of the date/time stamp in the Message Transcript. You can change the "Z" to an offset notation (+00:00) through the Configuration screen.

Note: Date/Time fields in the user interface are in local time. G2S messages use the configured time zone and offset format for all date/time information.

A new configuration option, **Default TimeZone** (**Tools > Configure > Engine Options > General**), allows you to change the time zone used by the tool.
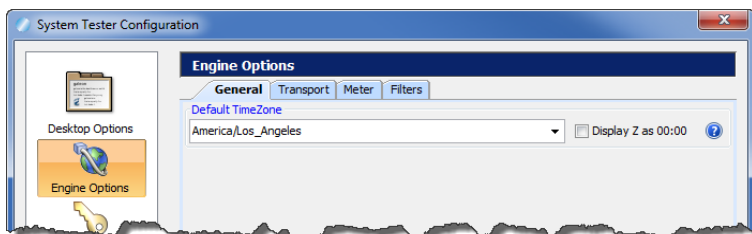
You can select the time zones in the following ways:

- **Scroll** through the list and select your country/city time zone *or* time preference (UTC / Zulu / GMT).

- **Type** the country/city time zone *or* time preference by clicking inside the **Default TimeZone** field and typing the first character(s) of the time zone or time preference you want. The field will auto-complete the entry. For example, if you want the time zone for Los Angeles, California, you can type: **America/L** and "Los Angeles" will be completed for you.

- **SelectDisplay Z as 00:00** if you **do not** want to see the default "Z" at the end of the message date/time stamp in the Message Transcript, which indicates that the time uses a zero offset (UTC, Zulu or GMT).

Note: You must restart the tool for changes to the **Default TimeZone** option to take effect.

- If you are using the *Tester Toolkit module*, Startup Algorithms can now include the **Send My Command** script verb. Select **Send My Command**, and then select the class, deviceID and command to be sent. The command is sent if the device is in the EGM descriptor list, and the deviceID is valid.
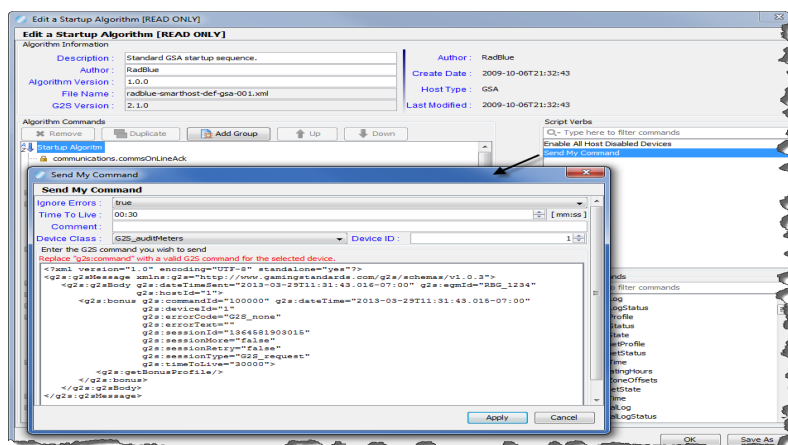
Note: All known namespaces are supplied by RGS.

Installation Modifications

- All RadBlue products now require **Java SE 7u11** or higher. If the installer cannot find this version on your computer, you are prompted to download and run the Java installer, *or* navigate to the correct Java version. This update should have minimal effect on your environment, but keeps our products on modern versions of the underlying technology.
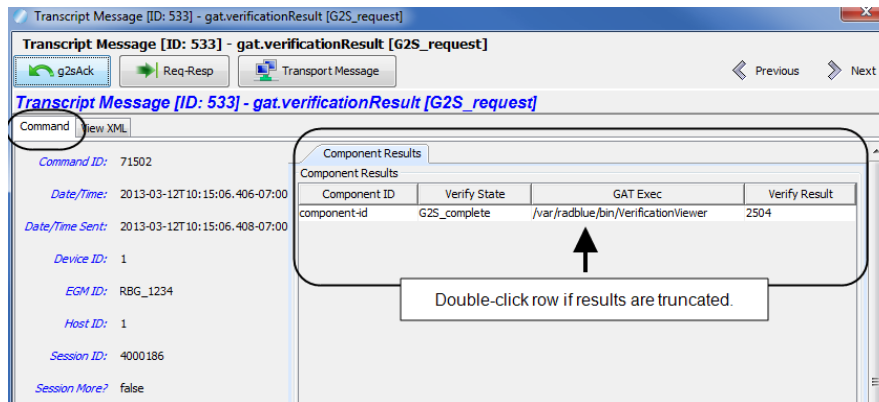
Engine Modifications

- In an ongoing effort to ensure that RadBlue tools use the latest technologies, RGS has been updated to Apache CXF version 2.7.2. This change should not affect tool performance or operation.

- When RGS receives a semantically incorrect `communications.descriptorList` response command, it now logs the issue and ignores the command.

- RGS now properly handles a `progressiveProfile` command with zero (0) `gameLevelConfig` elements.

- The process for cleaning up the **logs** directory when the tool is started has been improved.

User Interface Modifications

- The `download.deletePackage` option now shows you a list of all active packages for an EGM instead of just the latest package that was added. This option is found under **Send Command** > **G2S-download** > **deletePackage**.
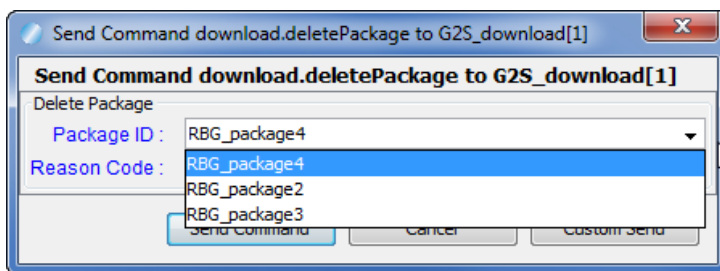
- A new table has been added to the Command tab on the `gat.verificationResult` message detail screen showing **Component Results**. Previously, this information was only viewable through the **View XML** tab.

Transcript Modifications

- The detail screen in the Message Transcript has been streamlined from two **View XML** options to one. The View XML button on the Command tab has been removed. To see messages in XML, click the **View XML** tab.

Release Date: 05 FEB 2013
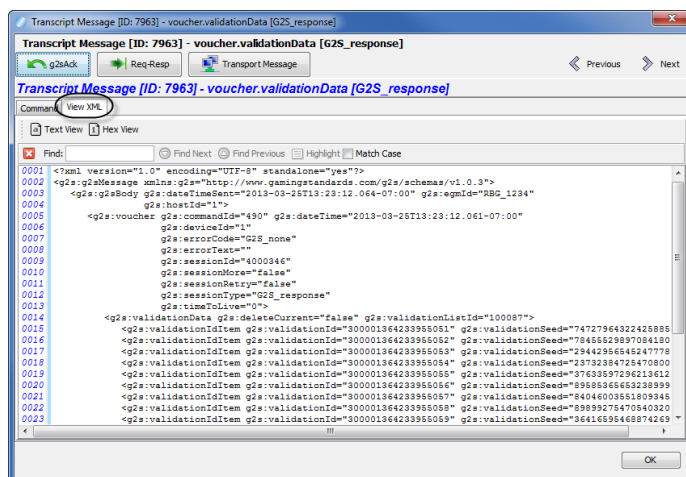
Release Summary

In this release, we updated several classes for G2S 2.1, updated RGS to 2013 licenses, made improvements to security options, made enhancements and corrected minor issues.

New Features

- *New for 2.1!* The following classes have been updated in accordance with the 2.1 protocol:
  - `gat`
  - `optionConfig`
  - `eventHandler`
  - `meters`

Installation Modifications

- RGS now requires a 2013 license. If you have not received a 2013 license, please contact RadBlue Support.

Security and Transport Modifications

- An issue in which the default keystore alias was not being saved has been corrected. You can set the default keystore alias by going to **Tools** > **Configure** > **Security Options** > **Keystore**, and clicking **Set As Default**.
- In a testing environment, you now have the option to enable minimum security. This option is found at **Tools > Configure > Security Options**.

When you enable this option:

- The **Transport Layer Security (TLS) 1.0** is the *only* supported protocol for client-side TLS sessions. Note that host-side sessions are not restricted.

- The only supported cipher suite is SSL_RSA_WITH_3DES_EDE_CBC_SHA for both client- and host-side TLS sessions.

Engine Modifications

- An issue in which the tool referred to the wrong namespaces for Audit Meters (http://www.gamingstandards.com/g2s/schemas/v1.0.3/g2sAM instead of http://www.gamingstandards.com/g2s/schemas/v1.0.3/ExtAM) has been corrected. RST now refers to the correct Audit Meters namespaces.

- An issue in which an error occurred when executing the `cabinet.resetProcessor` has been corrected.

- A new *second-pass-validate* attribute in the **g2s-validation-manager-sample.xml** file ([**installation directory**] > **schemas** > **g2s** > [**schema directory**]) that is designed to give you a second validation pass to detect syntax errors that are not found in the first validation pass attribute. Set this attribute to **false** if the second validation pass takes too much CPU resources, or the EGM/host is in violation of the G2S schema and you are unable to fix your code. The default is **true**.

```
<?xml version="1.0" encoding="UTF-8"?>
- <schema second-pass-validation="true" validation="true" nice="true" version="2.1.0">
    <loader classname="com.radblue.gsa.g2s.core.types.SchemaLoader"/>
    <source root-package="com.radblue.gsa.g2s"/>
  - <command-classpath>
      <classpath value="com.radblue.cvt.commands"/>
  </command-classpath>
  - <namespaces>
```

- RGS now detects when communications have been enabled on the EGM. If RGS has enabled communications, and the EGM sends a `commsDisabled` command, it automatically tries to enable communications again.

User Interface Modifications

- G2S 2.1 `optionConfig` modifications.
  The following `optionConfig` commands have been added to the Send Command layout:
  - `optionConfig.getOptionSeries`
  - `optionConfig.getOptionChangeStatus`
- The user interface for the **commConfig.setCommChange** command on the Send Command layout has been modified.
  - A new **Delete** button on the Devices tab lets you remove any selected device.
  - An **Automated Host Devices** field has been added to Host Editor (accessed by clicking **Edit** on the Host tab). When **Automate Host Devices** is selected, changes to the host ID are automatically applied to the host devices. If this field is not selected, the user must manually modify the host devices using the Add, Edit and Modify options on the Devices tab. Note that the **Host Registered** field must be set to **true** to use this option.
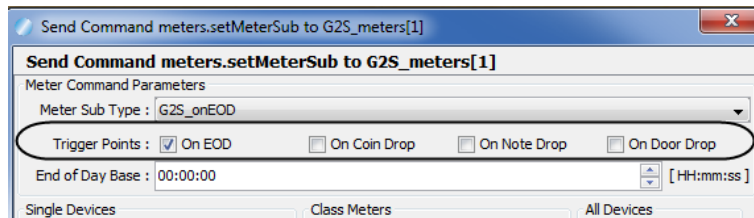
- G2S 2.1 `meter` modifications.

  You can now send four types of EOD subscriptions through the `meters.setMeterSub` command: `onEOD`, `onCoinDrop`, `onNoteDrop`, and `onDoorOpen`. The `meters.setMetersSub` command can be sent through the Send Command layout as well as added to the RGS meter subscription (**Custom Configuration** > **Components** > **Meter Subscription** *or* **Tester Toolki**t > **Components** > **Meter Subscription** > **Standard EOD and Periodic Subscriptions**).

- The **meters.setMeterSub** user interface has been reworked to make it more usable.

Tester Toolkit Modifications

Custom Scripting

- The following commands have been added to the **Available Commands** list (**Tester Toolkit** > **Custom Scripting**) :
  - `download.setScript`
  - `gameplay.setActiveDenoms`
- An issue in which Custom Scripting only worked if RGS owned the affected device(s) has been corrected. Custom Scripting now works whether RGS is the device owner, guest or configurator.
- The `gat.getSpecialFunctions` command will now only be sent if the EGM supports `gat` special functions.
- An issue in which the Wait for Events verb failed when the EGM restarted has been corrected.
- RGS will now continue to process a script, even though the EGM restarts communications (as is the case when the processor is reset or new devices are added to the EGM).

Transcript Modifications

- RGS now supports a Public Facing Transcript (PFT). The PFT is a program designed to generate a file of Message Transcript data in the `transcripts` directory (`public-facing-transcript.PFT`) that can be imported into a program of your choice for further analysis. Each record in the PFT follows a standard record format and is either a G2S message or a Transcript marker. Each Transcript record contains:

  - the date and time the record was posted to the transcript file.

  - a unique message ID to serialize all messages processed.

  - a sub-element containing a G2S message record or a marker record that was created to record a milestone in the test run.

  To enable the PFT, go to **Tools** > **Configure** > **Engine Options** > **General**, and select **Enable Public Facing Transcript** (restart required).

- The database for the Transcript Analyzer is now configurable through **Tools** > **Configure** > **Engine Options** > **Databases**.

  - **Enable General Transcript Analyzer** - Select to enable the EGM Transcript Analysis Report (**Transcripts** > **Analyze** > **General**). Once selected, you can configure the number of records to save for use with the EGM Transcript Analysis Report. Increasing the number of records, increases disk use. To improve RGS performance, clear this option.

  - **Save Transcript Messages to Database** - Select this option to save all transcript messages to the transcript database. Transcript records only need to be saved in the database if you want them to be persisted between runs of the tool or after you clear the transcript display. Saving a large number of transcript messages will impact performance. By default, this option is disabled.