System Tester, G2S Scope and Protocol Analyzer Quick Start

Looking for more information?

Radical Blue Gaming, Inc.

At the RadBlue forum you can find the latest release information, report issues, get your questions answered, and submit suggestions for improving our products. Simply log on to: http://radblue.mywowbb.com

phone: +1.775.329.0990
website:http://www.radblue.com
e-mail: sales@radblue.com

Gaming Standards Association

If you need a copy of the Gaming Standards Association's latest protocol documents, or you want to find out more about the GSA and the work being done by over 70 companies in the areas of protocol standardization for the gaming industry, we encourage you to discover more about the organization.

website: http://www.gamingstandards.com
e-mail: sec@gamingstandards.com

Contents

Introduction to RadBlue Tools 1
Before You Begin 2
Additional Resources 2

Module 1: Install the RadBlue Tools 3
Installation Overview 3
Installing the Tools 4
Install RGS 4
Install RPA 6
Install RST 7
Special Configuration Instructions for Windows 7 Operating Systems. 9

Module 2: Get Going! 11
Activity 1: Launch the RST SmartEGM 12
For Windows Systems 12
For Linux Systems 12
Activity 2: Launch RGS 13
Activity 3: Getting Around 14
Review the RST Interface 14
Review the RGS Interface 19
Activity 4: Configure the Transcript for Real-Time Updates (RGS and RST) 21
Activity 5: Start the SmartEGM (RST) 22
Activity 6: View Messages in the Transcript (RST and RGS) 23
What Are You Looking for in the Transcript? 23
View Message XML in the RST Transcript 24
Flying Solo Activity 1 . . . . . *26*
Activity 7: Look at the Debug Log (RST and RGS). 27
Activity 8: Look at the Descriptor List (RGS). 28

Module 3: Creating Activity 29
Flying Solo Activity 1 30
Activity 1: Subscribe to Events and Meters (RGS) 31
Flying Solo Activity 2 33
Activity 2: Send G2S Commands to the EGM (RGS) 34
Activity 3: Create EGM Activity (RST) 35
Activity 4: View Events as Command Objects 38
Flying Solo Activity 3 40

Module 4: Configure the Start-Up Algorithm 41
Flying Solo Activity 1 42
Activity 1: Start EGM with Disabled Communications (RGS) 43
Activity 2: View Messages with Communications Disabled (RST) 44
Activity 3: Enable EGM Communications (RGS) 45
Flying Solo Activity 2 46

Module 5: Insert RPA Between RST and RGS 47
Flying Solo Activity 1 47
Activity 1: Getting Around RPA 48
Activity 2: Create the RPA "From URL" (RPA) 49
Activity 3: Configure Client-Side (RST) Information (RPA) 50
Activity 4: Configure RGS Information (RPA) 51

Module 6: View Message Data in RPA. 53
Flying Solo Activity 1 53
Activity 1: Change the SmartEGM Configuration File (RST) 54
Activity 2: View Data (RPA) 55
Activity 3: Create a Message Error (RST) 56
Activity 4: View Message Error (RPA) 57

Module 7: Advanced Skills 59
Activity 1: Run RST and RGS on Two Different Computers 60
Activity 2: Add RPA Between RGS and RST 62
Activity 3: Generate the Transcript Analysis Report (RGS, RST or RPA) 64
Navigating the Report File 64
Generate the Report 65
Sample Report 66
Activity 4: Review Changes in the EGM's Data Model (RST) 70
Flying Solo Activity 1 73

Introduction to RadBlue Tools

The RadBlue G2S Scope (RGS) and RadBlue System Tester (RST) exercise G2S implementations of either an EGM or host system. However, you can easily configure RST and RGS to communicate with one another (on either the same computer or two different computers) to better understand the Gaming Standards Association's G2S message protocol. Once RGS and RST are communicating, you can add the RadBlue Protocol Analyzer (RPA) in the middle of the two applications to give you a window into what's going on in between.

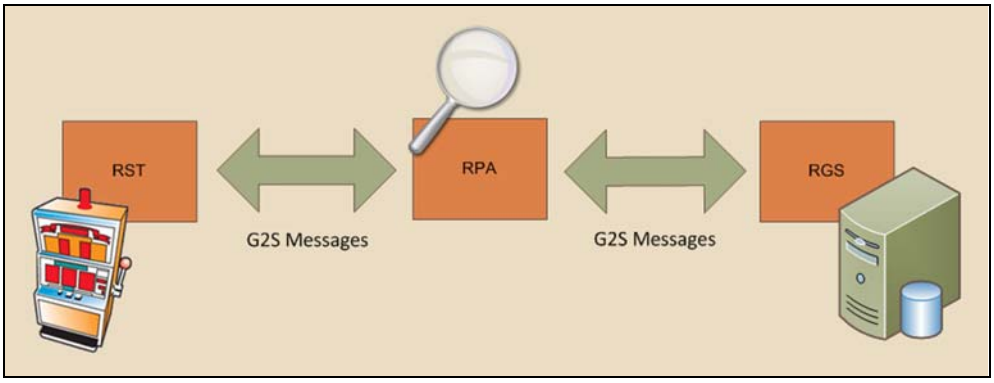


Figure 1: G2S message flow between RST, RPA and RGS.

The purpose of this guide is to get you up and running on RST SmartEGM, RGS and RPA, giving you the basics as well as some more advanced skills to help you move forward with your G2S implementation. Whether you are developing a new G2S application, testing an existing application or just learning about G2S, this guide will give you the information you need to get started quickly.

First, we'll walk you through RST and RGS. Once you're up and running, we'll add RPA between the two applications and show you how you can use it to troubleshoot messaging issues.

Before You Begin

- *At a minimum*, your computer must meet the following requirements to run all three applications simultaneously:
  - Operating System: Windows XP, Windows Vista, or Linux
  - Memory: 2 GB
  - Disk Space: 750 MB
  - Processor: Intel 2.8 GHz or comparable
- You should have received a license for each tool from RadBlue. If you haven't already done so, save the attachments to your desktop or a convenient folder where they can be easily located while installing the tools.

  No license files? Contact Russ Ristine at 775.329.0990 or Russ@RadBlue.com.
- Download all product installers from the following RadBlue web site: www.radblue.com/downloads/downloads.htm.
- Make sure you are familiar with the Gaming Standards Association (GSA) *G2S Message Protocol*. At a minimum, you should read chapters 1 and 2. The *G2S Message Protocol* can be downloaded from the GSA web site.

Additional Resources

The RadBlue web site contains information on all RadBlue products, including overviews, release notes, bulletins and user guides. Product user guides contain detailed information, reference material and step-by-step procedures.

- RGS User Guide
- RPA User Guide
- RST User Guide

Module 1

Install the RadBlue Tools

Installation Overview

1. Verify the location on your computer of the licenses for RGS, RST and RPA.
2. If you haven't done so already, download the RST, RGS and RPA installers from the RadBlue web site: www.radblue.com/downloads/downloads.htm
3. Install RGS.
4. Install RPA.
5. Install RST.

Installing the Tools

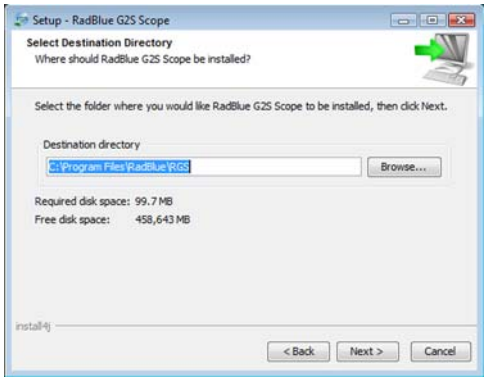
Install RGS

Perform the following steps to install RGS on a Windows operating system. Note that you must have the RGS license file on your computer prior to installation. If you have not received an RPA license file, contact Radical Blue Gaming.

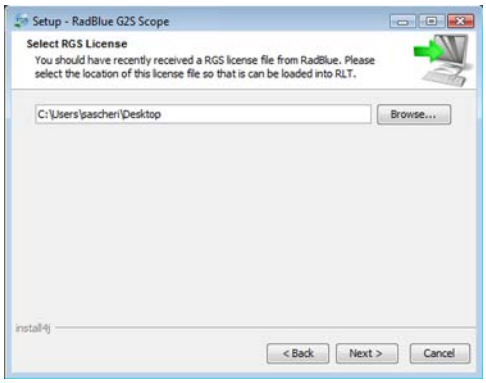
1. Double-click **RGS_*x*_*x*_*x*.exe**.
2. Click **Next**.

3. Review the RadBlue click-through agreement, and select **I accept the agreement** to accept the agreement.
4. Click **Next**.

5. Type the location where you want the RGS application installed, or click **Browse** to navigate to the location.
6. Click **Next**.



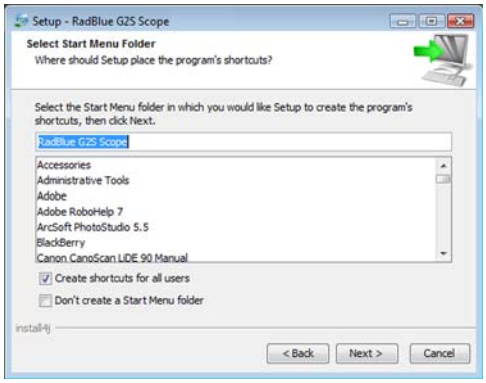
7. Navigate to the location of the RGS license file, and click **Next**.



8. Select the **Start Menu folder** for RGS.

   If you only want to create a shortcut for the current user, clear the **Create shortcuts for all users** checkbox.

   If you do not require a Start Menu folder for RGS, select **Don't create a Start Menu** folder.



9. Click **Finish**.

10. Double-click the RGS desktop icon to launch the application.

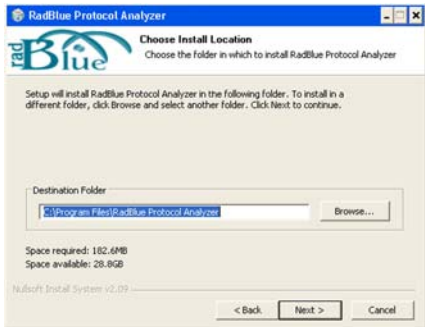
Install RPA

Perform the following steps to install Protocol Analyzer. Note that you must have the Protocol Analyzer license file on your computer prior to installing Protocol Analyzer. If you have not received an Protocol Analyzer license file, contact Radical Blue Gaming.

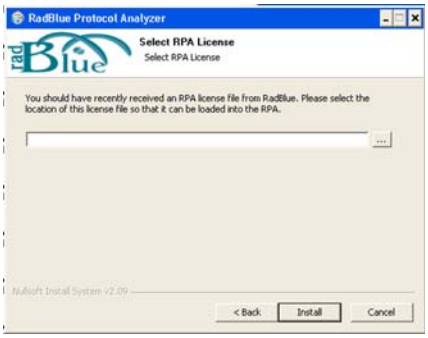
1. Double-click **Setup_RPA-*x.x*.exe**.
2. Click **Next**.

3. Review the RadBlue click-through agreement, and click **I Agree** to accept.

4. Select the location of the Protocol Analyzer application, and click **Next**.



5. Navigate to the location of the Protocol Analyzer license file, and click **Install**.



6. Click **Finish**.
7. Double-click the RPA desktop icon to launch the application.

Install RST

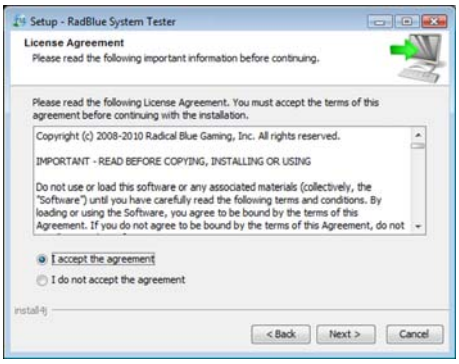
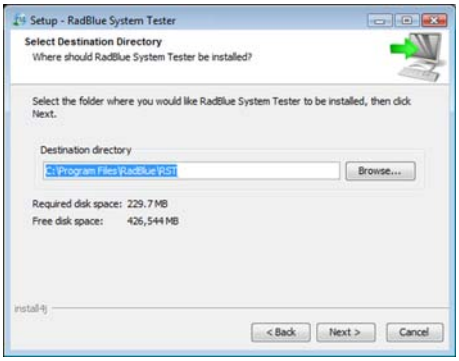
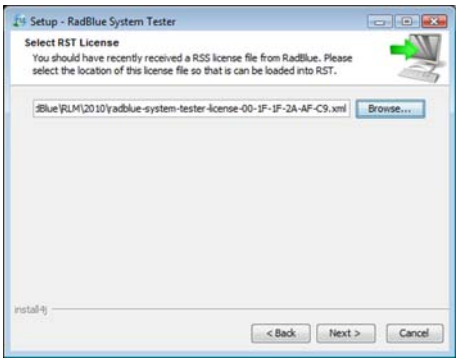
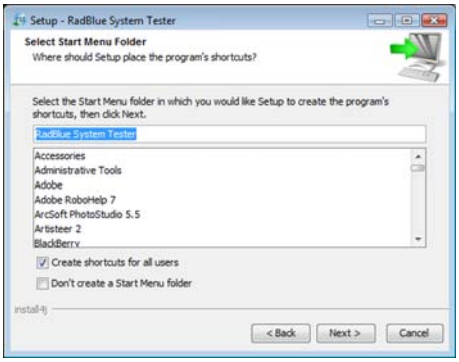
Perform the following steps to install RST on a Windows operating system. Note that you must have the RST license file on your computer prior to installing RST. If you are using a special version of RST, you must have a license for that version.

If you have not received an RST license file, contact Radical Blue Gaming.

Note If you are installing a version of the RadBlue System Tester (RST) released *prior* to version 9 (April 7, 2010) on a Windows 2007 operating system, you may need to perform additional configuration after completing the standard installation.

See Bulletin 02: Configuring RST for Windows 2007.

1. Double-click **RST_*x*_*x*_*x*.exe**.
2. Click **Next**.

3. Review the RadBlue click-through agreement, and select **I accept the agreement** to accept the agreement. **4.** Click **Next**.	
5. Type the location where you want the RST application installed, or click **Browse** to navigate to the location. **6.** Click **Next**.	
7. Navigate to the location of the RST license file, and click **Next**.	
8. Select the **Start Menu folder** for RST. If you only want to create a shortcut for the current user, clear the **Create shortcuts for all users** checkbox. If you do not require a Start Menu folder for RGS, select **Don't create a Start Menu** folder.	

9. Click **Finish**.	

Module 2

Get Going!

In this module, you will learn to:

- launch RST.
- launch RGS.
- navigate the RST and RGS user interfaces.
- configure the Transcript for real-time updates.
- load an RST SmartEGM file.
- start an RST SmartEGM file.
- work with the Transcript.
- work with the Debug Log.
- view the `descriptorList` command.

Activity 1: Launch the RST SmartEGM

For Windows Systems

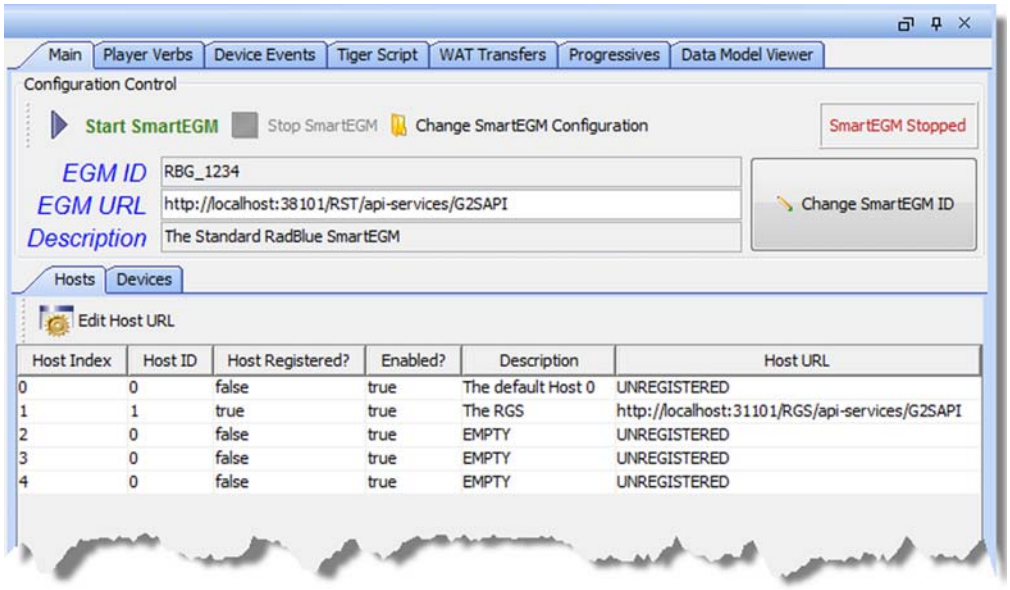
When RST is installed, a shortcut is created for each default desktop (SmartEGM, Edge and Central) under the Start menu. For this exercise, you will use the **RadBlue System Tester - SmartEGM** shortcut.

1. Navigate to: **Start** > **All Programs** > **RadBlue System Tester**
2. Drag and drop the **RadBlue System Tester - SmartEGM** shortcut onto your desktop to create a desktop icon.
3. Double-click the SmartEGM desktop icon to launch the RST-SmartEGM user interface.

For Linux Systems

1. Execute the **rst-smart-egm.sh** from the installation directory.

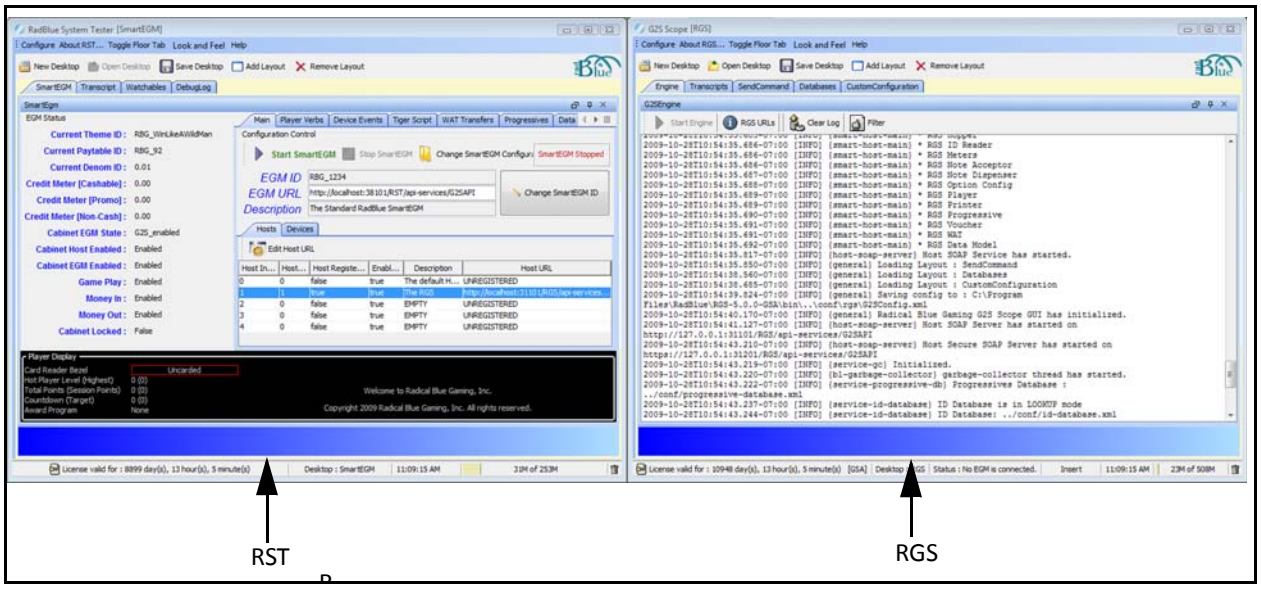
   Once you've launched the RST SmartEGM, you see the **Configuration Control** screen, which you'll use to start communications between RST and RGS.



Activity 2: Launch RGS

1. Double-click the **RGS** icon on your desktop to launch the RGS user interface.

   Your desktop should now look something like this:



If you don't have the desktop real estate, you can overlap the applications and toggle back and forth to complete the activities and tasks.

Activity 3: Getting Around

Before we go further, let's take a few minutes to review the RST and RGS interfaces.

Review the RST Interface

The SmartEGM layout lets you perform EGM activity and view the result of that activity on the EGM. The SmartEGM layout contains several object tabs, but let's look at the entire layout first.

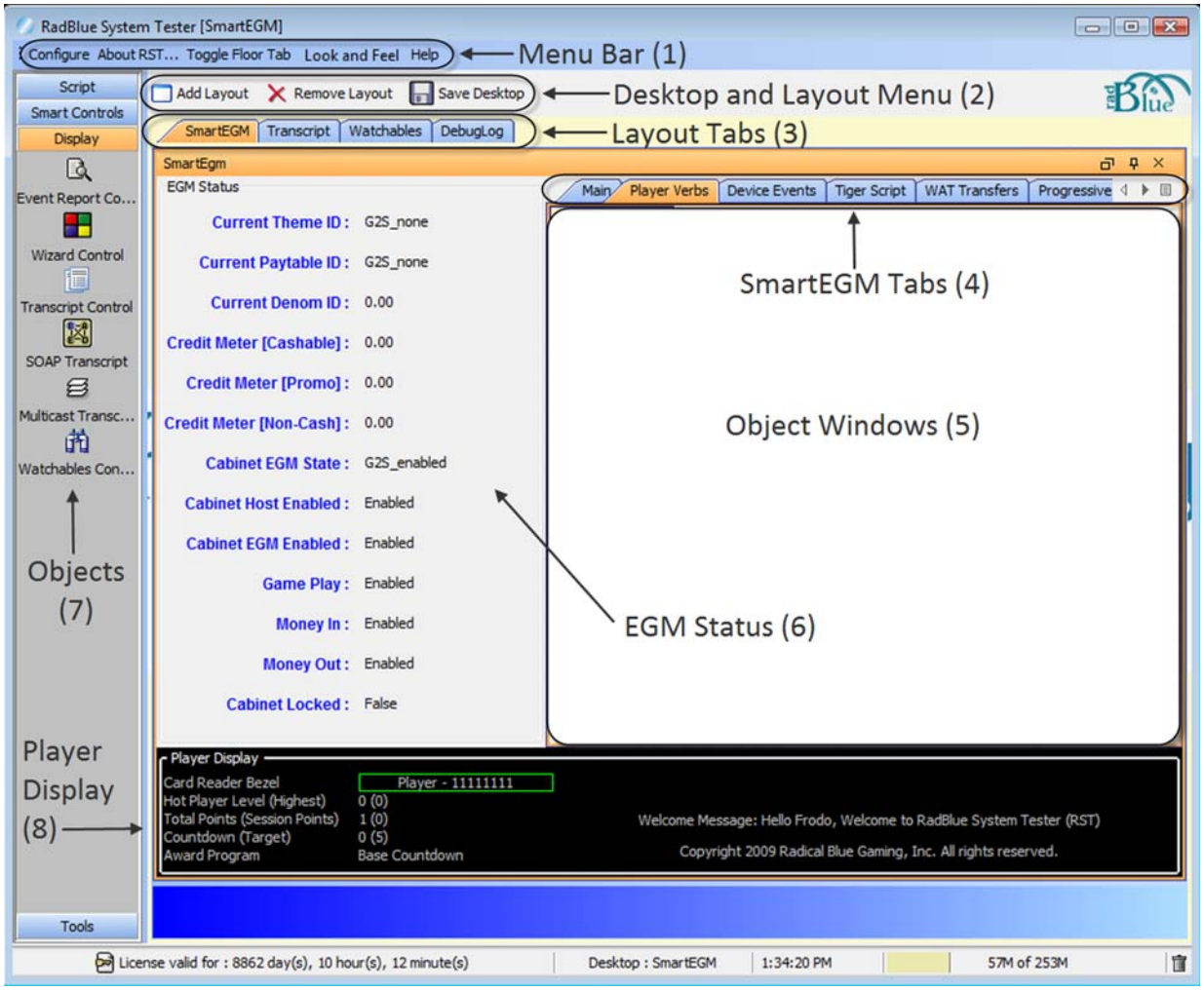
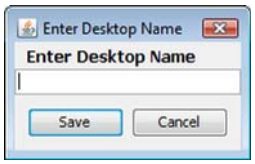


Figure 1 The SmartEGM layout is comprised of three static areas: EGM Status, Player Display, and the Object Window.

1. **Menu Bar** - From the menu bar you can configure various tool options (**Configure**), view tool version information (**About RST**), show/hide the floor tab (**Toggle Floor Tab**), configure the interface's appearance (**Look and Feel**), and access the RST **Help** system.

2. **Desktop and Layout Menu** - A Desktop is a collection of objects and tabs that constitutes the work area of the tool. Use this menu to open, add and save custom application desktops, and add and remove layouts.

   To add a new desktop, click **New Desktop**.



   Type the name of the new desktop, and click **Save**.

3. **Layout Tabs** - A series of tabs on the desktop, used to organize objects by function. Within each layout, objects can be placed next to each other, or on top of each other (in which case they are accessed by object tabs).

4. **SmartEGM Tabs** - The SmartEGM tabs allow you to perform various EGM functions, run scripts to automate EGM activity, and view EGM data. Below is a description of each SmartEGM tab.

Layout Tab	Description
Main	From the Main tab, you can load a SmartEGM configuration file and stop/start the SmartEGM engine.
Player Verbs	Player Verbs allow you to simulate EGM actions (player and employee) with the push of a button.
Device Events	The Device Events tab lets you create the events associated with physical device tilts. Whether a device tilt also tilts the SmartEGM depends on whether the device is required for play (in the SmartEGM configuration file). The individual device event tabs all look similar to that of the noteAcceptor. At the top you can select which *deviceId* to affect (from the currently available noteAcceptor devices). On the left side are the tilt events, some of which cause the device to be automatically disabled by the SmartEGM. Selecting the “Clear All Faults” button clears all selected faults, re-enables the device, and if the device was required for play, will also then cause the SmartEGM to be enabled.
Tiger Script	Tiger Script allows you more testing flexibility through the use of Tiger scripts, which let you automate the test environment, define what you test and how much testing of a given script you do. You can use the sample Tiger scripts provided or create your own, and automatically receive e-mail notification of the outcome of scripted tests. Note that this optional module is only active if your company has purchased the SmartEGM Tiger scripting module for the RST.

Layout Tab	Description
WAT Transfers	WAT Device 1 represents a host-controlled WAT device (where the host initiates the WAT transfer), and Device 2 is an EGM controlled device (where the player initiates the WAT transfer using the EGM's GUI). You are led through the steps of doing a WAT transfer with either type of device by the icons that are available to you. The Transfer Funds interface allows you to select the direction of transfer. If you are using the RST SmartEGM with the RGS host tool, the valid WAT accounts are 1111, 2222, 3333, 4444, 5555, 6666, 7777, 8888, 9999, and 12345678, and the valid player IDs are 12345678, 11111111, and 22222222.
Progressives	The Progressives tab displays progressive information. The `setProgressiveValue` command updates the Progressives screen.
Data Model Viewer	Data Model Viewer displays the EGM's data model, allowing you to snapshot current values and to compare two snapshots. The comparison feature gives you a way to easily detect any changes that may have occurred in the data model.

5. **Object Window** - The object window displays content associated with the selected SmartEGM tab.
6. **EGM Status** - The EGM Status bar appears alongside all SmartEGM screens. This screen displays current EGM settings. As you use player verbs, device events and WAT transfers to send messages, the EGM Status information is updated.
7. **Objects** - Contain a single function (or group of functions) that you work with in the tool.

   Objects are populated from the tool's data model. The data model reflects all of the data that has been captured by the tool and any updates that are received while the tool is running.

   When you first start the tool, all of the objects are empty. As messages are received by the tool, the appropriate objects are updated automatically. New objects are updated based on what's in the tool's data model. As a result, objects are immediately populated when dragged onto a layout, as long as the tool has been running for a while and has received the applicable command. The same behavior holds true when switching between desktops. If the command is in the data model, the object is automatically populated.

   The RST data model can be viewed on the SmartEGM layout by selecting the Data Model Viewer object tab.

8. **Player Display** - The Player Display shows messages sent from the host to the EGM, including welcome messages, award messages, session messages, and card-out ("goodbye") messages.

   The host has the option to use substitution *tokens*, special characters that display as pre-defined information (for example, player name or EGM ID) at the EGM. If the EGM is not carded on a player-required token, the actual token text displays. For example, if you use the player account number token (%a), you would see "%a" display instead of a player number. Substitution tokens can be used for any message type, and are described in Appendix E of the *G2S Message Protocol* document.

   To assist testing efforts, player information is displayed in the left-hand corner of the screen.

   - **Card Reader Bezel** displays the ID number of the inserted player card. A green border indicates that a player card is inserted; a red border indicates that there is no player card inserted at the EGM.
   - **Hot Player Level** displays the current hot player level.
     - **(Highest)** is the highest hot player level the player has obtained. This field resets at card-in and card-out, and does not require a player card-in for hot player determination to occur.
   - **Total Points** displays the total player bonus points (initial point balance + current session points). This field is initially populated by the `playerSessionStartAck` command.

     Note that if you send a `setPointBalance` command after a player session has started, the initial point balance is adjusted to the new value; the number sent is not added to the bonus point total.

     - **(Session Points)** indicates the bonus points accrued for the current session. The current session includes: base point awards, player point awards, generic override points awarded, and points awarded by the host (The individual values are in the player log record). This field is only applicable to carded players.

- **Countdown** displays a player's count until earning a specified number of bonus points.
  - **(Target)** is the number that the player must reach to receive the bonus point(s). Since you have the option to count down or up, what you can expect to see in this field will change. This field is populated by the `player.setCountdownOverride` command.

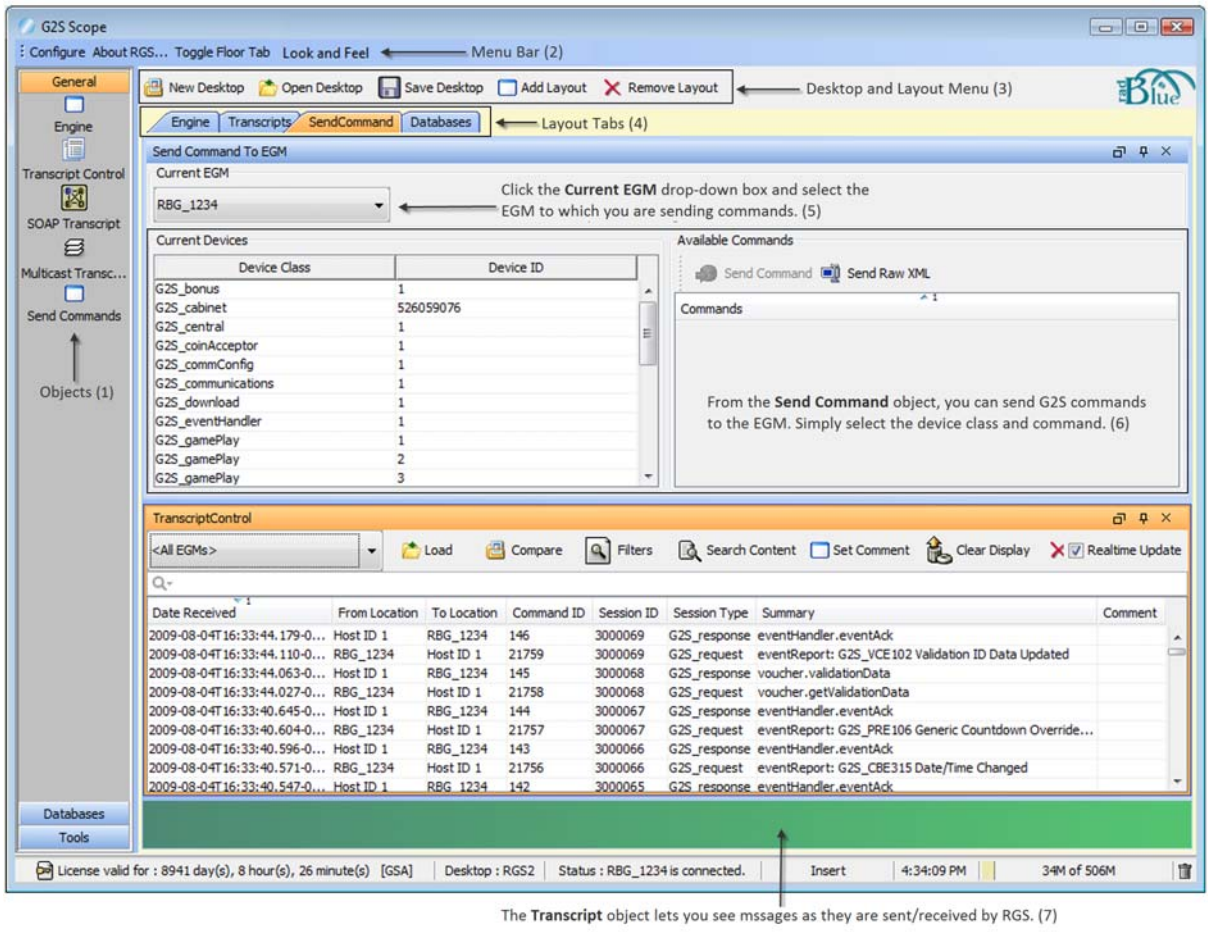
  Example

  - If you are counting up to 20 with one point earned, you would see: **Countdown (Target) 1 (20)**
  - If you are counting down from 20 with one point earned, you would see: **Countdown (Target) 19 (0)**
- **Award Program** displays the countdown parameters currently in effect (base, player or override).

Review the RGS Interface

From the RGS user interface, you can:

- select one of up to five EGMs and send G2S commands as a host.
- view all messages that are sent and received by RGS.
- compare two transcript message.
- view and modify the RGS voucher and progressive databases.
- create, delete and modify layouts and desktops.
- view SOAP and Multicast message transcripts.



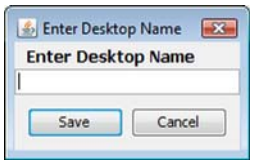
1. **Objects** - Contain a single function (or group of functions) that you work with in the tool. You can see all available objects on the floor tab. Click the **Toggle Floor Tab** button to show or hide the floor tab.

   Drag and drop an object onto the object window (area of screen where objects are displayed) when you want to create a custom desktop or layout.

2. **Menu Bar** - From the menu bar you can configure various tool options (**Configure**), view tool version information (**About RGS**), show/hide the floor tab (**Toggle Floor Tab**), and configure the interface's appearance (**Look and Feel**).

3. **Desktop and Layout Menu** - The **desktop** is a collection of objects and tabs that constitute the work area of the tool. The **layout** is a series of tabs on the desktop, used to organize objects by function. Within each layout, objects can be placed next to each other, or on top of each other (in which case they are accessed by object tabs). Use this menu to open, add and save custom application desktops, and add and remove layouts.

   To add a new desktop, click **New Desktop**.

   Type the name of the new desktop, and click **Save**.

4. **Layout Tabs** - A series of tabs on the desktop, used to organize objects by function. Within each layout, objects can be placed next to each other, or on top of each other (in which case they are accessed by object tabs).

5. **Current EGM** - Click the drop-down arrow and select the EGM with which you want to work.

6. **Send Command** - From the Send Command you can send G2S commands (simulating a G2S host). RGS allows you to send only one command at a time.

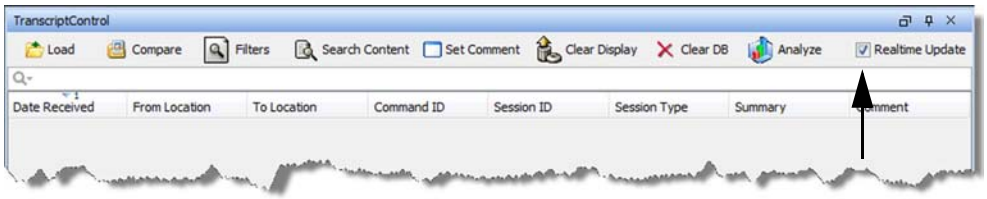
7. **Transcript** - The Transcript object appears at the bottom of the screen, so you can see the flow of messages between RGS and the RST SmartEGM as you use the Send Command.

Activity 4: Configure the Transcript for Real-Time Updates (RGS and RST)

The Transcript is a powerful tool for understanding how G2S works. We'll talk more about it in Activity 5, but for now, you just need to make sure it's configured correctly.

Do the following on both RST *and* RGS:

1. Click the **Transcript** tab.



2. Verify that the **Realtime Update** option is selected.

More Info

Once you select this option, messages appear in the Transcript *as they are received* by the tool.

If you want to see messages that were received *before* you selected Realtime Update, click **Load** to view the last *x* messages received by the tool (all messages received by the tool are stored in a database).

Changes to the Realtime Update setting are persisted, so if you disable this option, it will be disabled the next time you start the application.

Activity 5: Start the SmartEGM (RST)

The SmartEGM configuration file is the data model for the SmartEGM. By default, configuration file is loaded for you (either the standard configuration file *or* the last configuration used).

The communications endpoint is defined in the configuration file. Depending on the endpoint to which you are communicating, there are several configuration files available to you through the **Change SmartEGM Configuration** option. The SmartEGM file can also be customized to meet your development or testing needs. See the RST User Guide for more information.

1. Click the **Main** tab on the **SmartEGM** layout, if it is not already selected.

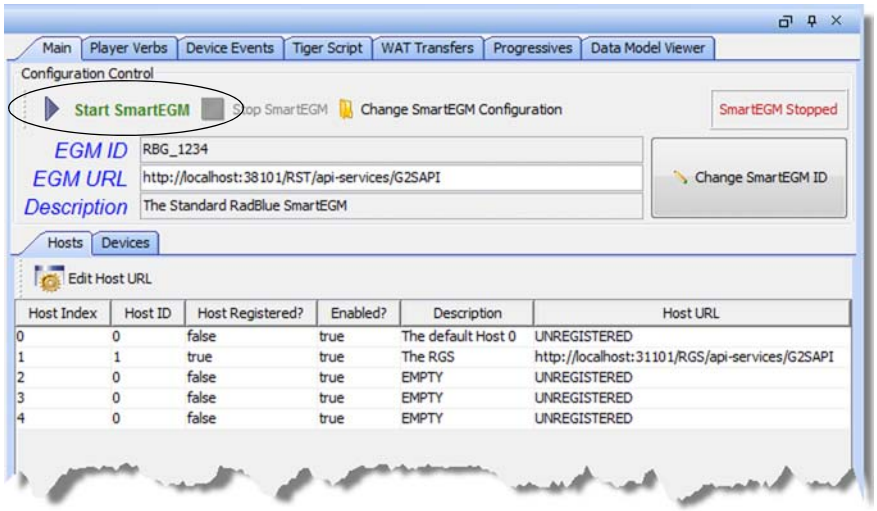
   The latest **smartegm-config.xml** file, which is configured to communicate with RGS, is automatically loaded for you. If you are using a student license, the **smartegm-config-student-edition.xml file** should be loaded.

2. Verify that the correct configuration file has been loaded by looking at the Description field.

   - If you are using a standard license, the Description field displays "The Standard RadBlue SmartEGM."
   - If you are using a student license, the Description field displays "The Student SmartEGM Config File for RGS."

3. If necessary, select the correct configuration file.

   - Click **Change SmartEGM Configuration**.
   - Select **smartegm-config.xml** to highlight it. If you are using a student license, select **smartegm-config-student-edition.xml**.
   - Click **Open**.



4. Click **Start SmartEGM**.

Activity 6: View Messages in the Transcript (RST and RGS)

The Transcript is available in RST, RGS and RPA. It is a feature you will use frequently when working with RadBlue tools. The Transcript lets you see messages as they flow back and forth between two endpoints. Thus, your basic workflow in RST and RGS will be:

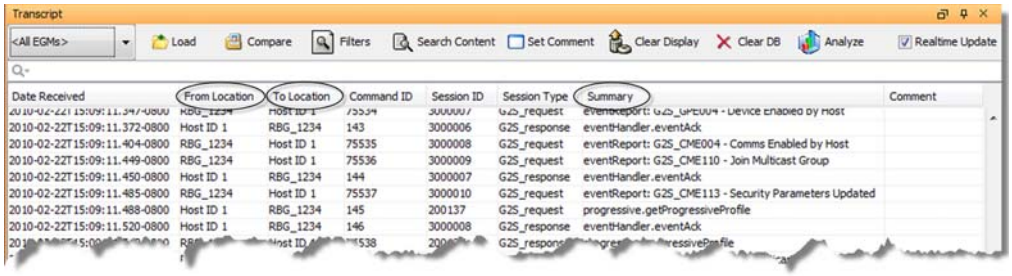
1. Do something.
2. Look at the Transcript.
3. Do something else.
4. Look at the Transcript.

You get the idea.

The data displayed in the Transcript is extracted directly from the G2S messages being sent and received. As you work with Transcript messages, we'll show you how you can drill down to view each message's XML content.

Once you start the SmartEGM, messages begin flowing between the RST SmartEGM and RGS. In this activity, we'll explore the Transcript.

1. Click the **Transcript** tab in RST and RGS.



2. Look at the Transcript messages flowing in and out of each tool. Notice the direction (**To Location** and **From Location**) and the G2S command (displayed in the **Summary** column) contained in each message.

What Are You Looking for in the Transcript?

- ❑ Are the correct commands being sent?

  For example, is the `commsOnline` command being sent during startup?

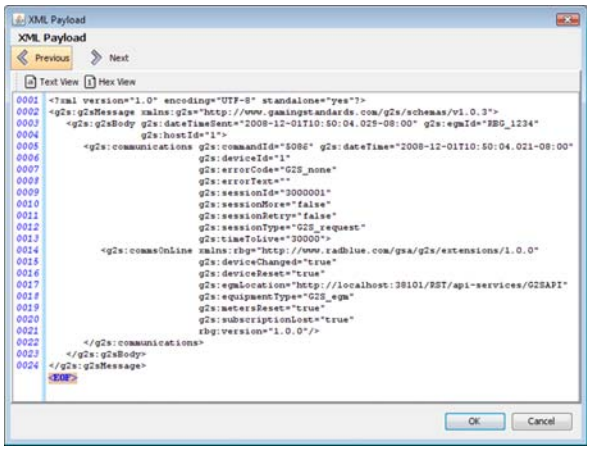
- ❑ Are messages being acknowledged (with a `G2SACK` as well as a message ACK)?

- ❑ Are there correct request-response pairs?

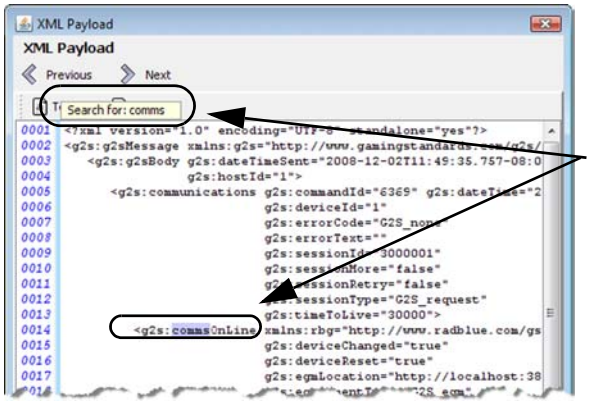
  For example, if a `communications.getDescriptor` command request is sent, a corresponding `communications.descriptorList` command response should be received. Note that the **Session ID** is the same for the request and response.

View Message XML in the RST Transcript

1. Select any message in the Transcript, and double-click to display the XML Payload screen.
2. Look at the message header and find the schema version being used.
3. Scroll down the screen, and look at the content of the message. Note that all details of the message are displayed in an easy-to-read format:



4. Click **Next** and **Previous** to scroll through the Transcript messages without leaving the XML Payload window.
5. Click **Hex View** to see the message content in hexadecimal format.
   Click **Text View** to toggle back to the XML version.
6. Click in the message window, and type **comms**.



The search utility is invoked automatically when you click in the XML Payload window and start typing.

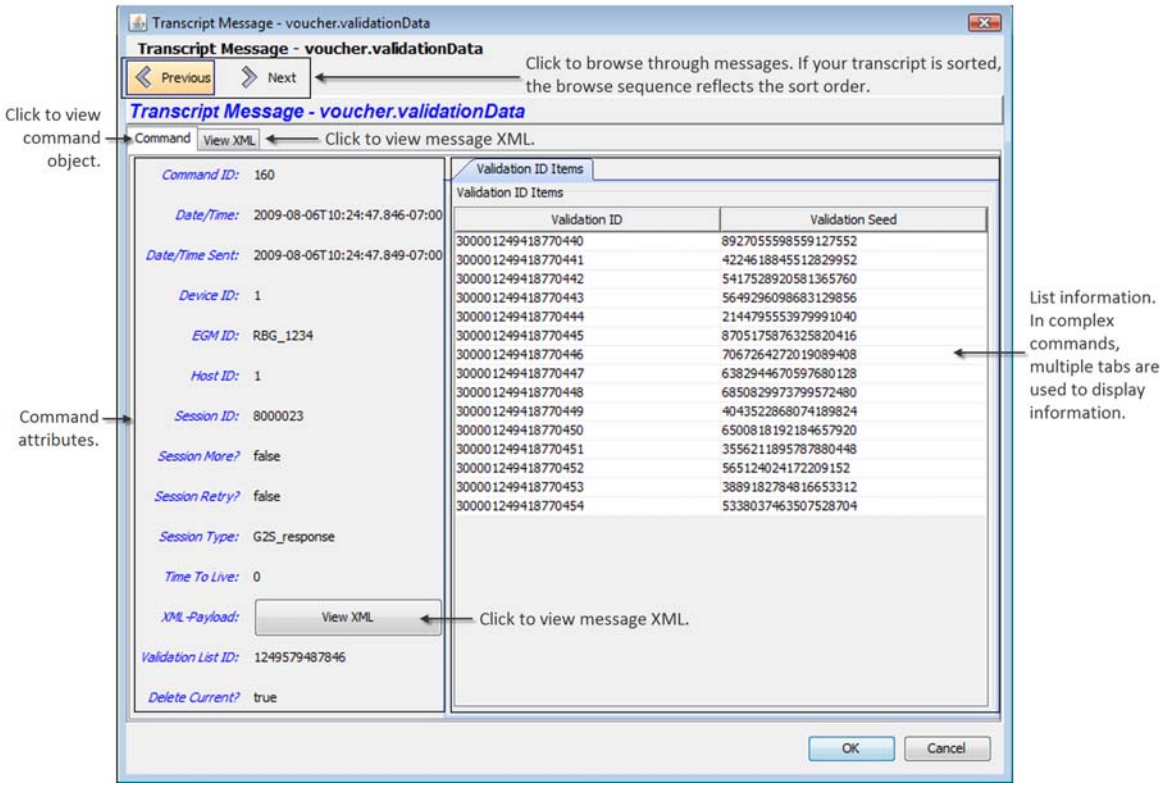
7. Use the down arrow to locate the next instance of the search string; use the up arrow to return to the prior instance of the search string.
8. Click **OK** to close the XML Payload window and return to the Transcript.

View Command Objects through the RGS Transcript

In addition to viewing a message's XML, you can also view the *command object*. A command object is a graphical representation of a command (as opposed to viewing the command in XML format). You can view command objects on the message details screen. List information for complex commands displays in tabs to the right of the command attributes.

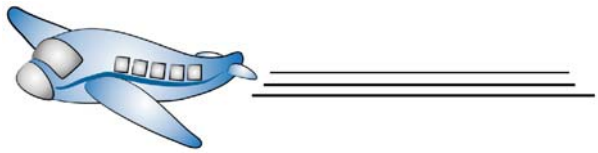
To display a message's command object:

1. Double-click the command (message) you want to view. In this example, we chose the `voucher.validationData` command.
2. Click the **Command** tab. While each command is unique, the layout is similar for each one: command attribute list on the left and list information on the right.

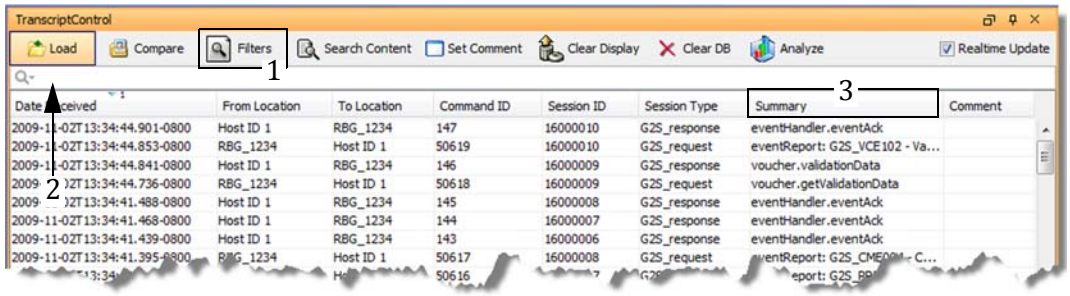


3. Click **Previous** and **Next** to navigate through the transcript list while in the message detail view.
4. Click **OK** to return to the transcript.

Flying Solo Activity 1



Now that you're familiar with the Transcript, use the Transcript filtering options to search for a specific message or message type.



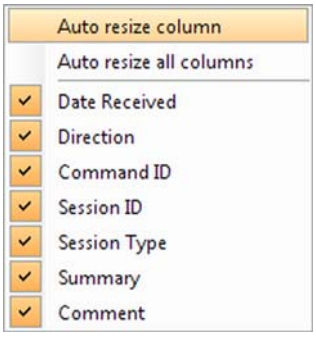
Task 1. Click **Filters**, clear the **G2SACK** command check box, and click **OK**. How does this change your view of the information?

Task 2. Click in the **Quick Search** dialog box, and type **comms**. What happens in the Transcript as you type? Click on the magnifying glass, and use the options in the menu to expand and narrow your search.

Task 3. All columns are sortable. Click in the **Summary** column header to sort the Transcript by command name.

Task 4. Sort on multiple columns using the CTRL key. Press **CTRL** and click on the **Session Type** column header. Then, click **Summary**. How does this affect how the information sorts?

Task 5. You can choose to show or hide Transcript columns. Right-click in any column header. Click **Summary** to hide the Summary column. Right-click again, and click Summary. What happens?



Task 6. Right-click in any column header, and resize one or all columns to fit the content.

Task 7. Rearrange column order by clicking on a column header and dragging it to a new location.

Activity 7: Look at the Debug Log (RST and RGS)

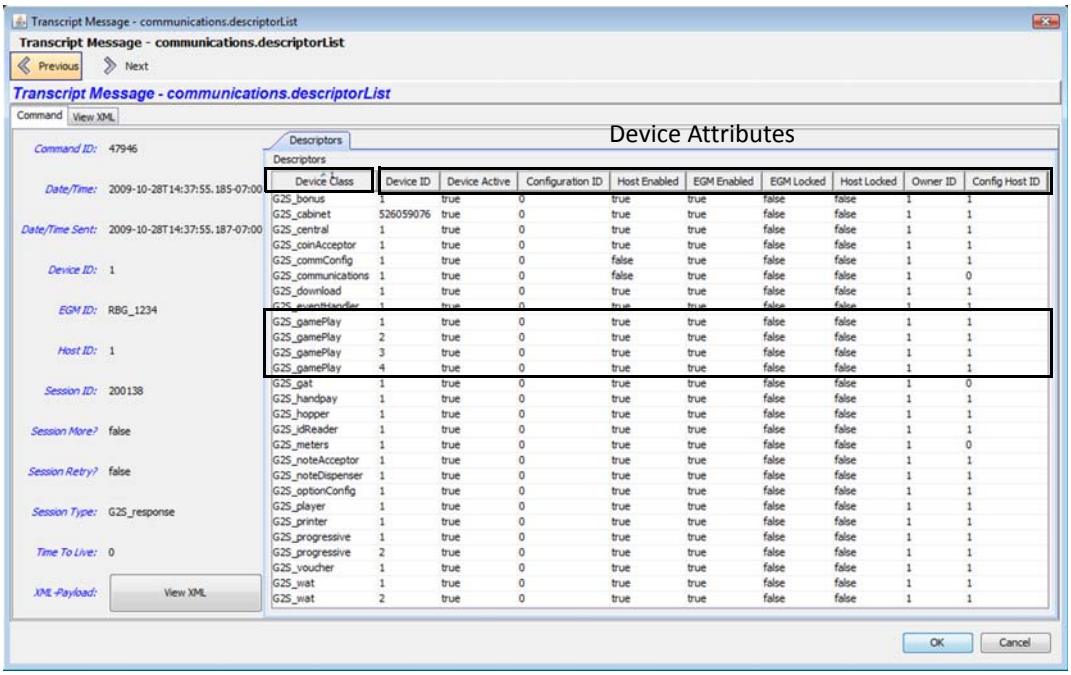
The Debug Log displays communications activity between RST and RGS. Go to the Debug Log to look for errors encountered in messages handled by the tool. Message categories are:

- **INFO** - Messages that do not impact the system, but may be useful to know. INFO messages appear in **black** text.
- **DEBUG** - Fine-grained informational events that are useful in troubleshooting. DEBUG messages appear in **black** text.
- **WARN** - Messages that indicate potentially harmful situations. WARN messages appear in **blue** text.
- **ERROR** - Messages related to program errors. ERROR messages appear in **red** text.
- **FATAL** - Designates a severe error events that will presumably lead the application to abort. FATAL messages appear in **red** text.
- **UNKNOWN** - Messages that have not been assigned a logger designation. UNKNOWN messages appear in **pink** text.

1. In RST, click the **Debug Log** tab.
2. In RGS, click the **Engine** tab.
3. To view just the error and debug messages, click **Filter**, clear the **INFO** option, and click **Apply**.
4. Do you see any red or blue text, which may indicate a communications issue or a problem with one of the messages?

Activity 8: Look at the Descriptor List (RGS)

Through the Transcript, you can view the `communications.descriptorList` command to view current classes and devices for the EGM (in this case, the RST SmartEGM).



1. Go to the RGS Transcript.
2. Use the Quick Search feature to search for **communications.descriptorList**.
3. Double-click the `communications.descriptorList` command to launch the detail view.
4. Look at the **Device Class** column to see which device classes are currently supported by RST.
5. Notice how many devices are available for each class. (For example, in the figure above, there are four gamePlay devices.)
6. Review the displayed attributes.
7. Double-click any device to view all available attributes for that device.
8. Click **Back** to close the Row Values screen.
9. Click **OK** to close the communications.descriptorList detail view and return to the Transcript.

Congratulations! You've completed Module 2. Close both tools and go have a coffee.

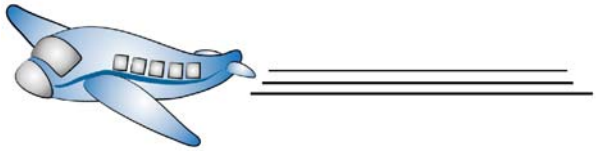
Module 3

Creating Activity

In this module you will learn to:

- subscribe to events.
- subscribe to meters.
- send G2S commands through the RGS Send Command layout.
- use the RST SmartEGM Player Verbs to create EGM activity.
- view eventHandler and eventReport information.

Flying Solo Activity 1



Now that you've completed Module 1, let's test your knowledge to get up and running for Module 2.

Task 1. Launch RGS.

Task 2. Launch RST.

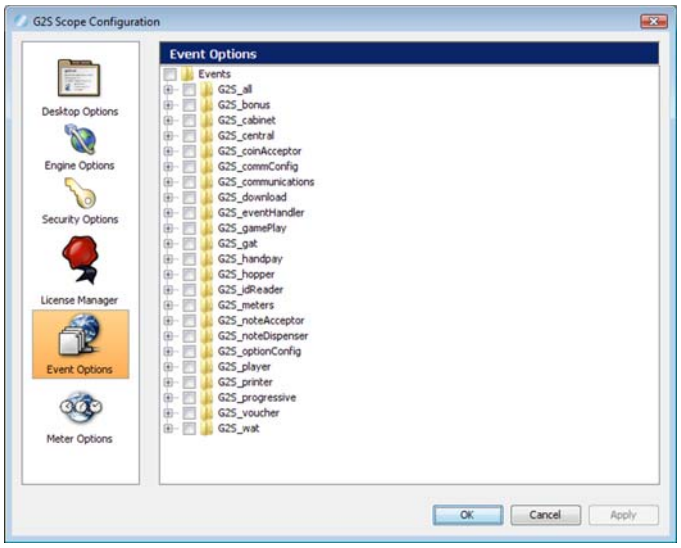
Activity 1: Subscribe to Events and Meters (RGS)

Both event and meter subscriptions are set by the RGS during the start-up algorithm that is executed whenever a `commsOnline` command is received. Options on the configure menu are used to configure these initial subscriptions.

Event subscriptions allow you to be notified by the EGM when the subscribed-to event occurs. You must subscribe to an event in order to be notified when that event occurs. Meter subscriptions allow you receive an EGM's meters at the intervals you specify.

This activity shows you how to set the initial event and meter subscriptions.

1. Select **Configure** from the menu bar.
2. Select **Event Options** from the left-hand menu to configure event subscriptions.

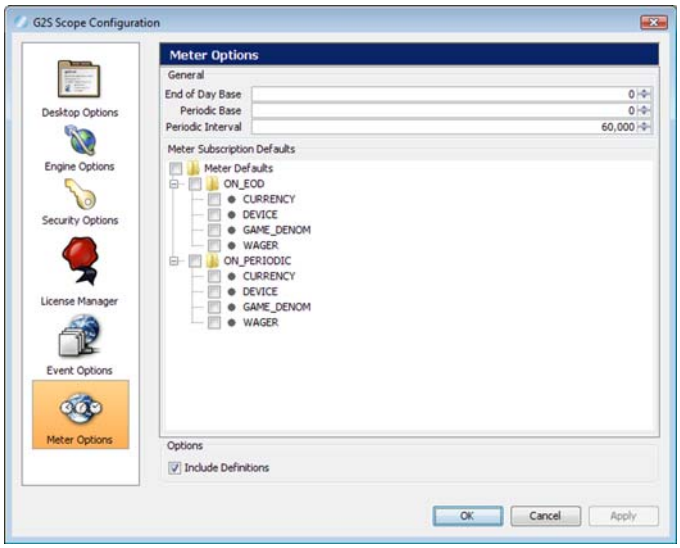


Event Options is used to configure the event subscriptions sent by the RGS during the start-up algorithm (triggered whenever a `commsOnline` command is received from an EGM). In this tree-structured control, selecting or clearing any box automatically causes any options beneath that box to be affected. The organization of the tree is by Class, then by a listing of all events within the class, followed by the six possible attributes for the control. These attributes control what associated data is to be included with the `eventReport`, whether only updatable meters are sent, and whether the `eventReport` is to be persisted until an `eventReportAck` is received by the EGM.

For simplicity, the selected event subscriptions are applied during the start-up algorithm to every device within the class, reported by the EGM in its `descriptorList`. This can be refined later using the Send Command control to set/clear subscriptions to events for a specific device within a class.

3. For this activity, select **Events** (at the top of the tree) to subscribe to all events.
4. Click **Apply**.

5. Select **Meter Options** from the left-hand menu to configure meter subscriptions.



In the General section of the Meter Options screen, you can specify the following:

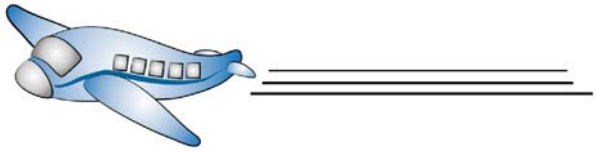
- **End-of-Day Base** -Indicates when the daily EOD meters are to be sent, expressed as the number of milliseconds after midnight.
- **Periodic Base** - Indicates when the daily Periodic Meter report schedule begins, expressed as the number of milliseconds after midnight.
- **Periodic Interval** - Indicates how frequently Periodic Meter reports are to be generated (in milliseconds).

In the Meter Subscription Defaults section, you can specify the meter subscription that is sent to the EGM during the start-up algorithm for every device in the `descriptorList` command (by definition, the EGM ignores subscriptions for devices for which it does not maintain meters). The available meters sets (currency, device, game denomination, and wager meters) can be selected for the default end-of-day and/or periodic meter subscriptions.

Finally, you can indicate whether the meter subscription should also request that the EGM include definitions of each meter in the reports that are generated by the EGM as a result of these subscriptions. Including the meter definitions causes three additional attributes to be included with each meter - *meterType* (count or value meter), *meterIncreasing* (Does meter always increase? Or does it go up and down like a credit meter?), and *meterRollover* (the maximum value for this meter after which it resets to zero).

6. For this activity, select **Meter Defaults** to subscribe to all meters.
7. Click **Apply**.
8. Click **OK**.

Flying Solo Activity 2



Now that you've configured event and meter subscriptions, start the tools.

Task 1. (RST) Verify that the **smartegm-config.xml** file is loaded by checking that the Description field displays **The Standard RadBlue SmartEGM**. If it does not, click **Change SmartEGM Configuration**, select the file, and click **Open**.

Task 2. (RST) Start the SmartEGM.

Task 3. (RGS) Look at the Transcript and verify that the SmartEGM is now sending events.

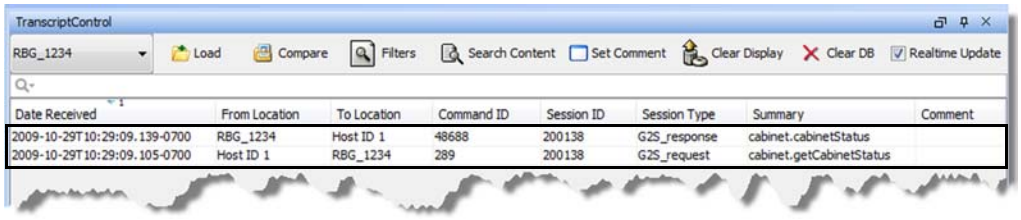
Activity 2: Send G2S Commands to the EGM (RGS)

The SendCommand layout lets you send G2S commands to an EGM and view the responses. This layout contains the sendCommand object and the Transcript.

1. Click the **SendCommand** layout tab.
2. Select the **TranscriptControl** object (lower section), and click **Clear Display** to clear the Transcript.
3. Select the **Send Command to EGM** object (top section). Notice that the **Current EGM** drop-down lets you select one of up to five EGMs that may be configured to communicate with RGS at once. Any commands you send will be sent to the EGM identifier displayed under Current EGM. Since you are only communicating with RST, no additional EGM identifiers are displayed.
4. Select **G2S_cabinet** from the Current Devices list. The **Available Commands** list automatically populates with all commands associated with the selected device.
5. Select **Cabinet - Get Cabinet Status** from the Available Commands list.

Note If the selected command has any configurable attributes, you are able to modify those attributes here.

6. Click **Send Command**.
7. Look at the messages as they appear in the Transcript. Are the G2SACK messages displaying? Use the **Filters** option to remove them from view.



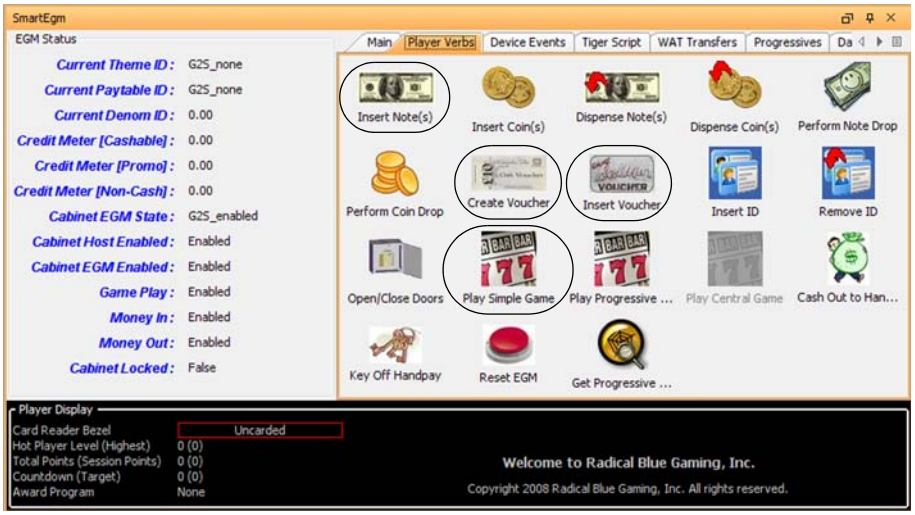
8. Double-click on the `cabinet.cabinetStatus` response, and review the content.

Activity 3: Create EGM Activity (RST)

Use the RST SmartEGM Player Verbs to simulate EGM activity. Each Player Verb button push represents an *action* at an EGM and may generate multiple messages to RGS. This activity shows you how to simulate simple game play. *Be sure to review the Transcript after you send each Player Verb, so you can see the flow of messages*.

The EGM Status bar appears alongside all SmartEGM screens. The EGM Status displays current EGM settings. As you use player verbs, device events and WAT transfers to send messages, the EGM Status information is updated.

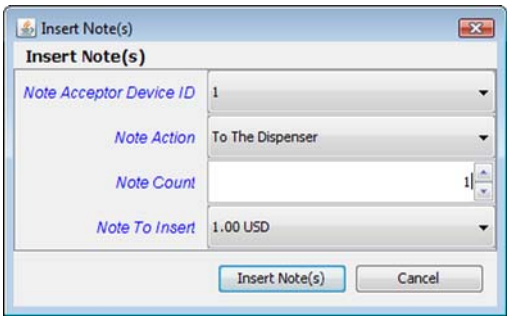
1. Click **Player Verbs**.



For this activity, you will simulate the following actions at an EGM:

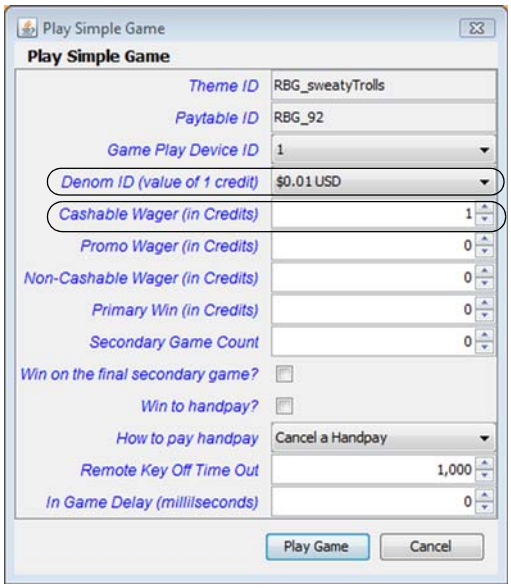
- insert a note
- play a simple game
- create a voucher
- insert a voucher

2. Click **Insert Note(s)**.

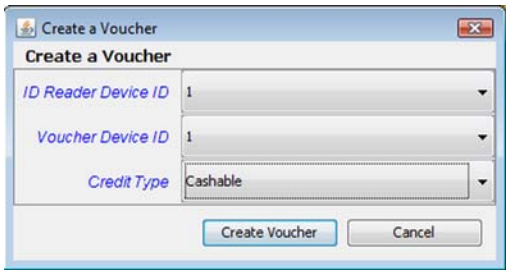


3. Change the **Note Count** to **10** and the **Note To Insert** to **10.00 USD**.

4. Click **Insert Note(s)**. Notice that the **Credit Meter [Cashable]** field in the EGM Status panel changes from zero (0) to **100.00**.
5. Click **Play Simple Game**.

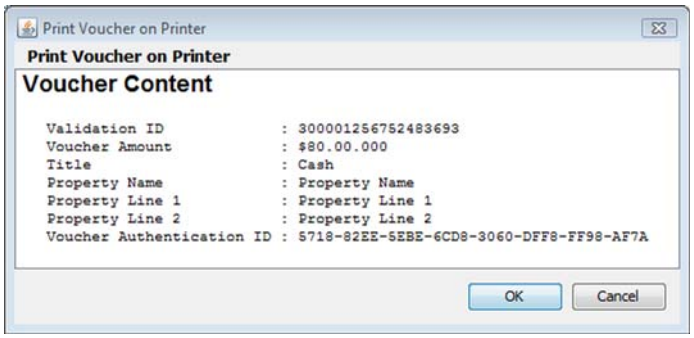


6. Change the **Denom ID** to $**5.00 USD** and **Cashable Wager (in Credits)** to **4**.
7. Click **Play Game**.
8. Look at the EGM Status panel. The **Current Denom ID** field is now $**5.00** and the **Credit Meter [Cashable]** field is **80.00**.
9. Click **Create Voucher**.



Since the EGM funds in this example are cashable, leave the voucher **Credit Type** as **Cashable**. If the EGM credit meter also has promotional and/or non-cashable, you could select one of those credit types for your voucher. When you select a credit type, the entire amount shown on the EGM credit meter prints to the voucher, and that meter changes to zero (0.00) in the EGM Status panel.

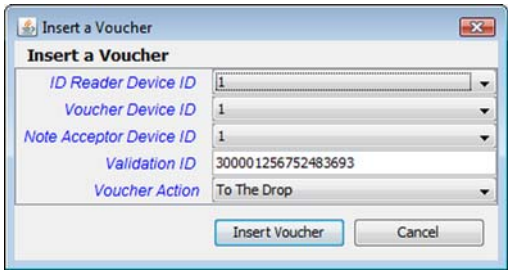
10. Click **Create Voucher**.



11. Review the content of the (simulated) voucher, and click **OK**.

The validation ID is stored in the RST's voucher database and becomes available for the **Insert Voucher** verb.

12. Click **Insert Voucher**.



The **Validation ID** field is automatically populated with an ID from the voucher database. You can use this ID or enter one of your own.

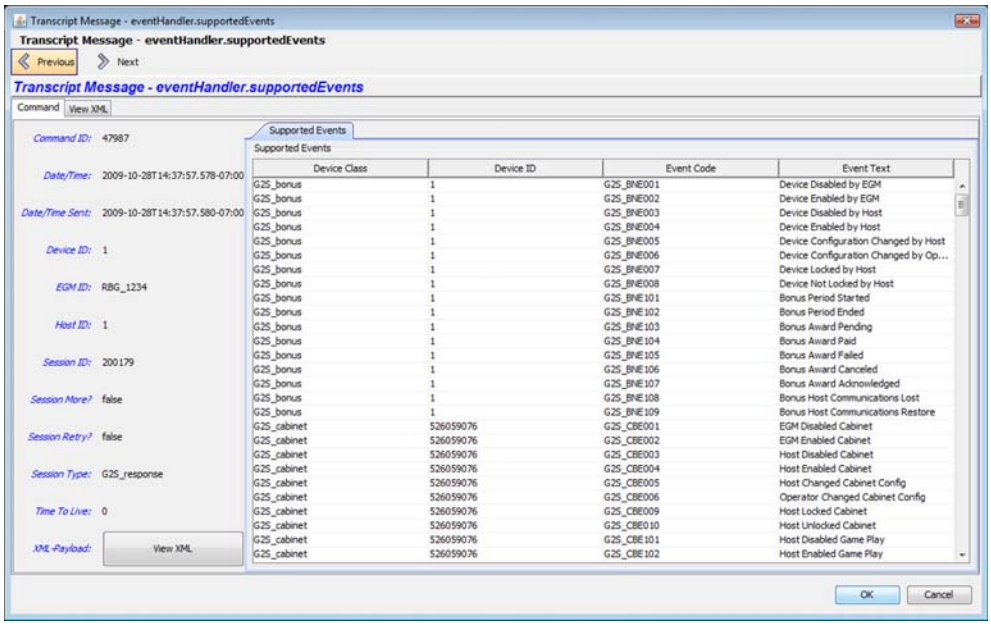
13. Click **Insert Voucher**.

Notice that the Credit Meter (Cashable) field in EGM Status panel changes to **80.00**.

Activity 4: View Events as Command Objects

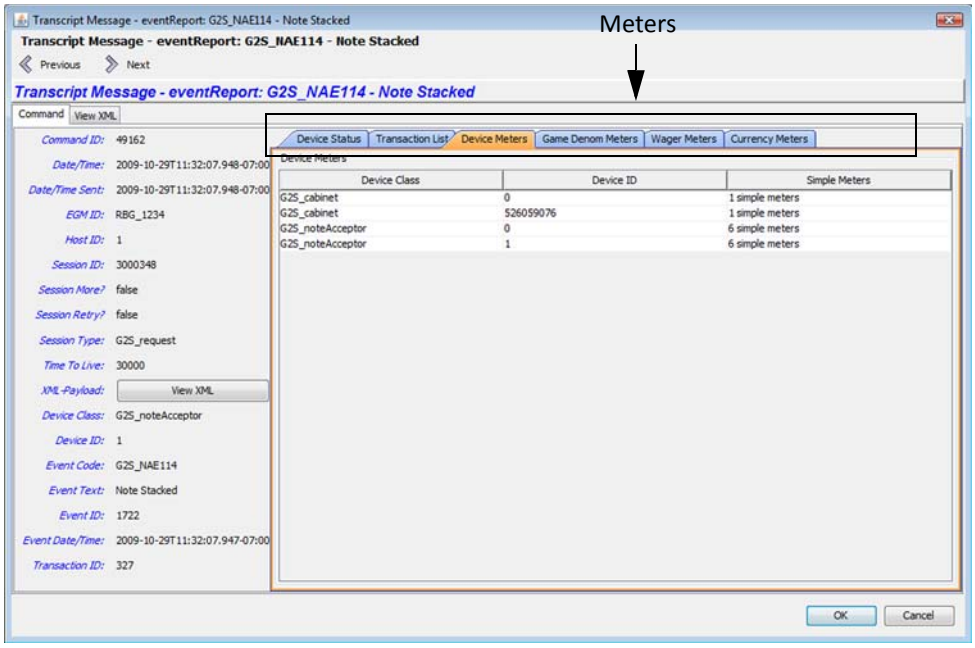
You can view the command object for event information through the Transcript by looking at event commands (eventHandler and eventReport). In this exercise, you will view events supported by the EGM and event reports generated by the EGM when a supported event occurs.

1. Go to the RGS Transcript.
2. Use the Quick Search feature to search for **supportedEvents**.
3. Double-click the `eventHandler.supportedEvents` command to launch the detail view.



4. Notice that the **Device Class** column lists, by device class, all events currently supported by the EGM.
5. Scroll down to the **G2S_noteAcceptor** device class and verify that the G2S_NAE114 (Note Stacked) event is supported by the EGM.
6. Click **OK** to return to the Transcript.
7. Use the Quick Search feature to search for **eventReport**. All event reports received by RGS should be displayed.

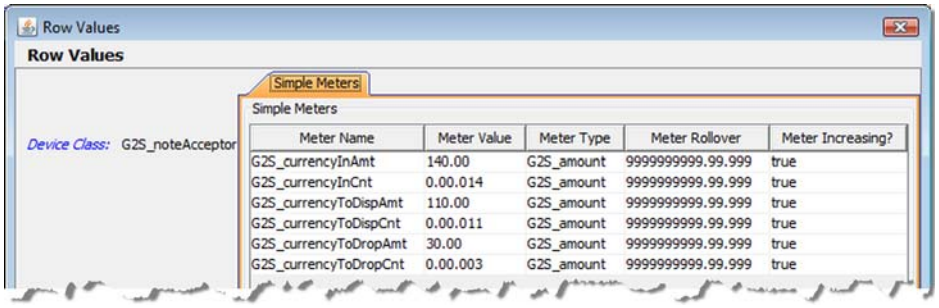
8. Scroll down until you find **eventReport: G2S_NAE114 - Note Stacked**, and double-click the row. Notice that the "Meters" tabs display all of the meters that were updated by the event.



A Device ID of zero (0) denotes class-level meters (all devices). Non-zero device identifiers denote the meters for each device within the class. This is most relevant for the gamePlay class where there may be dozens or hundreds of games within the cabinet.

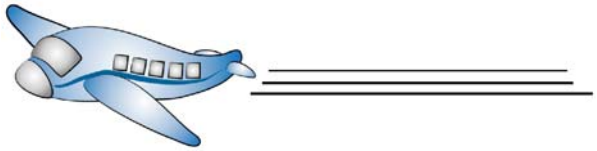
9. Click the **Device Meters** tab.

10. Double-click the **G2S_noteAcceptor** device with **6 simple meters** to view information on each of the six meters.



11. Click **Back** and **OK** to return to the Transcript.

Flying Solo Activity 3



The table below describes each Player Verb. Use it to explore more EGM actions and their subsequent message flows.

Player Verb	Description
Cash Out to Handpay	This verb simulates a player pushing the cash out button, and putting the EGM into a handpay situation.
Create Voucher	This verb simulates a cashout to a voucher.
Dispense Coin(s)	This verb simulates a cashout using coins.
Dispense Note(s)	This verb simulates a cashout to bills.
Insert Coin(s)	This verb simulates the insertion of coin into the EGM coin acceptor.
Insert ID	This verb simulates the insertion of the player ID (for example, a player card) into the EGM's ID reader.
Insert Note(s)	This verb simulates the insertion of bills into the EGM's note acceptor.
Insert Voucher	This verb simulates the insertion of vouchers into the EGM's note acceptor.
Key Off Handpay	This verb simulates the clearing of a handpay condition on the EGM.
Open/Close Doors	This verb simulates the opening or closing of EGM doors.
Perform Coin Drop	Simulates the removal of coin from the EGM.
Perform Note Drop	Simulates the removal of the stacker from the EGM note acceptor.
Play Simple Central Game	Simulates game play using a central determination system.
Play Simple Game	Simulates normal EGM game play.
Remove ID	Simulates the removal of the player ID from the EGM.
Reset EGM	Resets the cabinet status for lock states that cannot otherwise be cleared through G2S commands or the SmartEGM user interface.

Module 4

Configure the Start-Up Algorithm

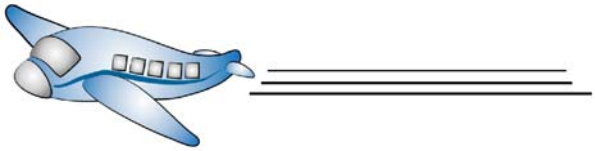
The startup algorithm contains the commands sent to an EGM by RGS when the EGM first comes online (triggered when a `commsOnline` command is received from the EGM).

Normally, RGS sends all commands in this list to the EGM for each device in the `descriptorList` command. If your EGM is having problems with the commands in a particular class, configuring the start-up algorithm provides an easy way to have RGS omit commands from its start-up sequence.

In this module you will learn to:

- start the EGM with communications disabled.
- view messages with communications disabled.
- enable communications using the RGS Send Command feature.

Flying Solo Activity 1



Task 1. Launch RGS.

Task 2. Launch RST.

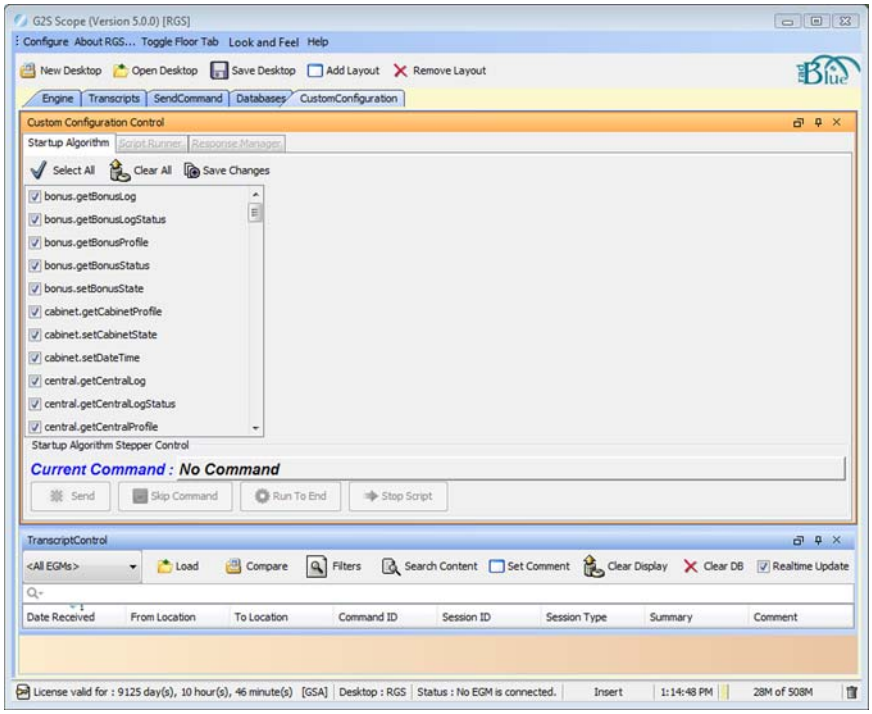
Note Do **not** start the RST SmartEGM engine yet.

Activity 1: Start EGM with Disabled Communications (RGS)

In G2S, the EGM cannot send unsolicited messages to the host until its communication device has been enabled by the host (through a `communications.setCommsState` command). This command is generally sent at the end of the host's startup algorithm when the host has finished querying the EGM and setting event and meter subscriptions.

In this activity, you'll customize the RGS startup algorithm by disabling the `communications.setCommsState` command.

1. Click the **CustomConfiguration** tab.



2. Clear the `communications.setCommsState` checkbox.
3. Click **Save Changes**.
4. Click **OK** on the **Settings Saved** dialog box.

Activity 2: View Messages with Communications Disabled (RST)

Now that the RGS startup algorithm is configured, start RST and see what happens.

1. Verify that the **smartegm-config.xml** file is loaded by checking that the Description field displays **The Standard RadBlue SmartEGM**. If it does not, click **Change SmartEGM Configuration**, select the file, and click **Open**.
2. Start the SmartEGM.
3. Look at the Transcripts for RST and RGS. Each command in the algorithm is sent and RST sends a corresponding response.

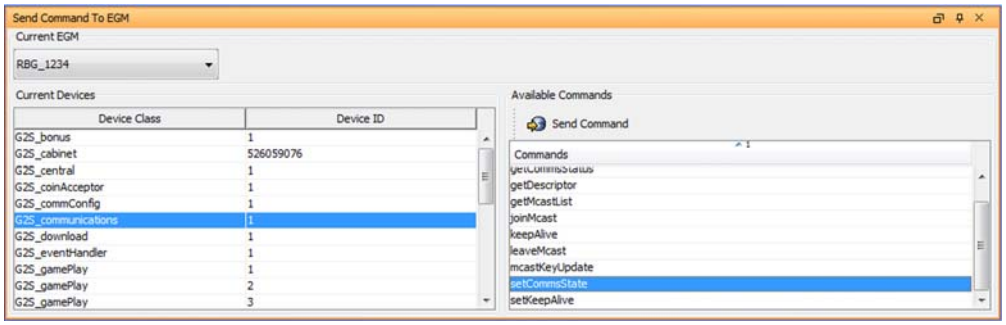
   When the start-up algorithm ends, note that no events are generated by the EGM. Instead, RST sends a `commsDisabled` command every 30 seconds, for which RGS sends a `commsDisabledAck` response.

   Note The *syncTimer* interval, which is the frequency at which the `commsDisabled` commands are sent as a reminder to the host, can be changed in RGS by going to **Configure** > **Engine Options** > **Communications - Sync Timer**.

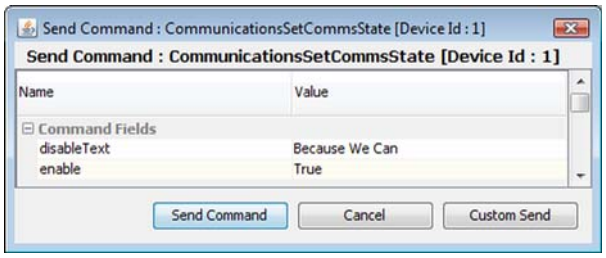
Activity 3: Enable EGM Communications (RGS)

Now, enable the communications with the EGM by sending a `communications.setCommsState` command.

1. Click the **Send Command** layout.

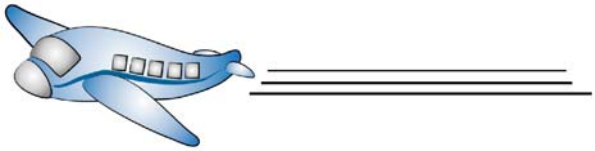


2. Select the **G2S_communications** device class, and then select **setCommsState** from the Available Commands list.
3. Click **Send Command**.



4. Click in the Value column for the **enable** field, and select **True**.
5. Click **Send Command**.
6. Look at the Transcript. Now that the communications device is enabled, normal messaging begins, so all subscribed events are sent and the `commsDisabled` command is no longer generated.

Flying Solo Activity 2



Task 1. Use the Send Command and Configure option to enable/disable various G2S devices. Use the Transcript to see the effect of each enable/disable action.

Task 2. Stop RST and close both tools.

Module 5

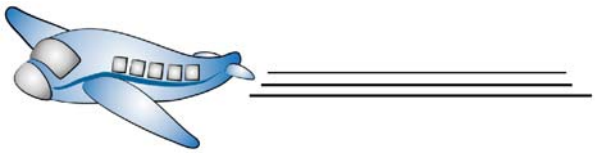
Insert RPA Between RST and RGS

Now that you know how to get RST and RGS communicating, let's insert RPA between those two endpoints.

In this module you will learn to:

- navigate the RPA user interface.
- configure RPA to communicate with RST and RGS.

Flying Solo Activity 1

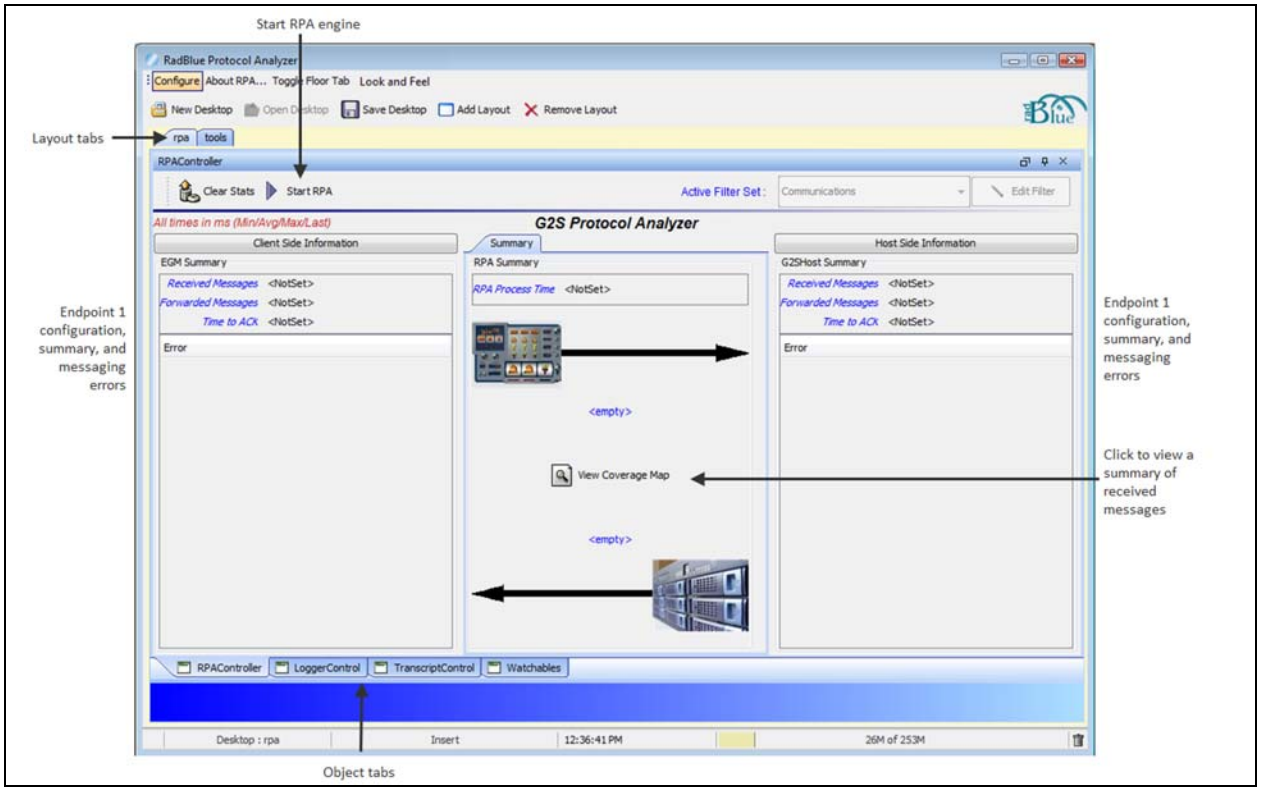


Task 1. Launch RPA by double-clicking the RPA icon.

Activity 1: Getting Around RPA

Take a few moments to familiarize yourself with the RPA interface. You'll find that the general layout is similar to RST and RGS. The menu bar, including the placement of the configuration (Configure) options, is the same. Your old friend the Transcript is back as well.

The RPA Controller layout changes slightly depending on whether or not you are using disruptive filters. For the purposes of this guide, we're going to assume that you are not using disruptive filters. For more information on disruptive filters, see the RPA User Guide.



There are three main sections on the RPA layout:

- **EGM Summary** - This section displays message processing information for commands sent by the EGM as well as any messages with errors.
- **RPA Summary** - This section displays the current command being handled by RPA. Click the **View Coverage Map** to see a list of all commands and events and the number of times RPA has received each.
- **G2S Host Summary** - This section displays message processing information for commands sent by the host as well as any messages with errors.

Activity 2: Create the RPA "From URL" (RPA)

In this activity, you will create the RPA location that appears in the *fromUrl* attribute in messages sent from RPA to the host system.

1. Select **Configure** from the menu bar.
2. Select **Engine Options**.
3. Set the **IP Address and SOAP Port** number.

   The **Bind To** field is IP address that RPA uses for communications. The **SOAP Port** and **SSL SOAP Port** are the ports that RPA uses for communications. In most cases, you will use the default port numbers.

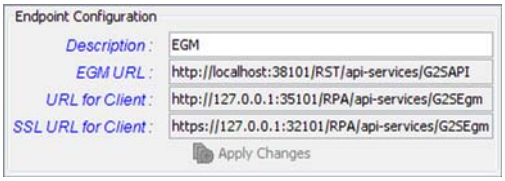
   - Click the drop-down arrow, and select **127.0.0.1** (localhost) since RGS and RST are on the same computer as RPA.
4. Set the **Protocol Options**.

   This option lets you define the GSA protocol RPA uses. The **Protocol Name** and **Version** values are used to create the **Schema Location**.

   - Verify that the **Protocol Name** is **G2S** and the **Version** is **1.0.3**.
5. Click **OK**.
6. On the RPA layout, click **Host Side Information**.
7. Look at the **RPA's From Address** field and verify that it displays the following URL: **http://[local IP address]/RPA/api-services/G2SHost**

Activity 3: Configure Client-Side (RST) Information (RPA)

1. On the RPA Controller screen, click **Client Side Information**.



2. Enter a description for the client (for example, EGM).

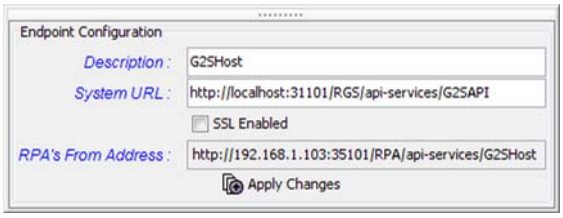
Note The **URL for Client** and **SSL URL for Client** are automatically calculated based on the information you enter in step 2.

Be sure that the EGM uses the **URL for Client** *or* **SSL URL for Client** for this host in its registered host list, for the Protocol Analyzer network address. *If the EGM does* **not** *use this address, it will not be able to communicate with the Protocol Analyzer.*

3. Click **Apply Changes**.

Activity 4: Configure RGS Information (RPA)

1. On the RPA Controller screen, click **Configure Host Side**.



2. Enter a description for the host (for example, G2SHost).
3. Enter the URL for the G2S host from step 1.
4. Click **Apply Changes**.

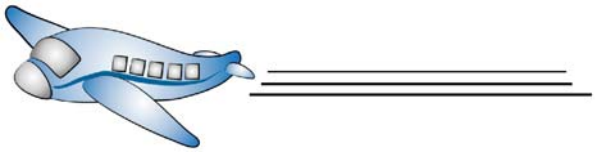
Module 6

View Message Data in RPA

In this module you will learn to:

- change the SmartEGM configuration file.
- view RPA data.
- use RST to create a message error that you can see in RPA.
- view message errors in RPA.

Flying Solo Activity 1



Task 1. RPA should already be running. If not, launch RPA.

Task 2. Launch RGS.

Task 3. Launch RST.

Activity 1: Change the SmartEGM Configuration File (RST)

1. Click the **Main** tab on the **SmartEGM** layout, if it is not already selected.
2. Click **Change SmartEGM Configuration**.
3. Select **smartegm-config-rpa.xml** to highlight it. If you are using a student license, select **smartegm-config-student-edition-rpa.xml**.
4. Click **Open**.
5. Click **Start SmartEGM**.

Activity 2: View Data (RPA)

1. Go to the RPA application, and start the RPA engine by clicking **Start RPA**.
2. Look at the three summary sections to view messages flowing through RPA. You can see the number of messages received from the EGM and host as well as the current messages being handled by RPA (under RPA Summary).
3. Click **View Coverage Map** to see the number of messages received for each message and event. Notice that commands and events that have not been received by RPA appear in red.
4. Use the Quick Search feature to search for the **cabinet.setCabinetState** command.
5. Now search for **eventReport**, and use the scrollbar to see the events that have been sent between the two endpoints.

Activity 3: Create a Message Error (RST)

1. From the SmartEGM layout, click the **TigerScript** tab.
2. Click **Select Script**.
3. Select **smartegm-example-send-raw-001.xml**, and click **Open** to load the selected script.
4. Click **Start Script**.
5. Click **OK** when the successful completion message displays.

Activity 4: View Message Error (RPA)

1. From the RPA layout, double-click the error that appears under the Error section in the EGM Summary.
2. Explore the Error Browser data. Note the error request comment, telling you what is wrong with the selected message (Cannot find the declaration of 'element')
3. Click **View XML**. Can you see the error in the XML to which the error request comment refers?

Module 7
Advanced Skills

In this module, you will learn to:

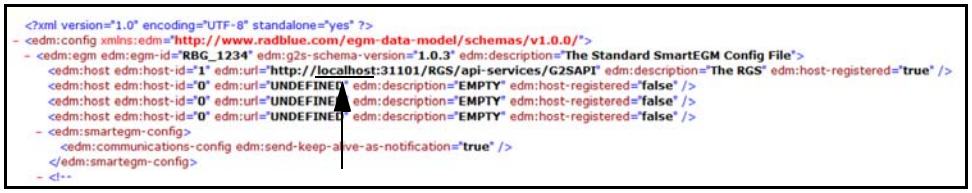
- configure RST and RGS to run on two different computers.
- add RPA between RST and RGS.
- generate the Transcript Analysis report.
- view changes to the EGM's Data Model.

Activity 1: Run RST and RGS on Two Different Computers

Once you have mastered running RST and RGS on the same computer, you may want to run them on two different computers to get a better understanding of network communications. Using two different computers provides a more realistic messaging simulation.

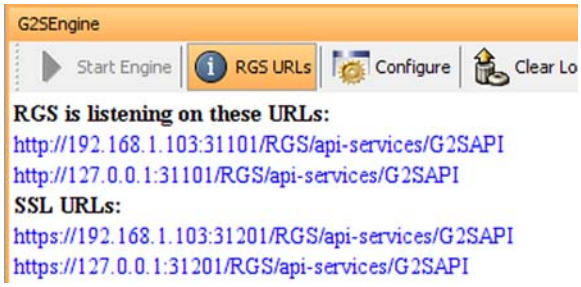
1. Run the RGS installer on computer 1.
2. Launch RGS with the desktop of your choosing.

   Note The RGS needs no special configuration because it gets the EGM's URL from the `commsOnline` message.
3. Run the RST installer on computer 2.
4. Still on computer 2, make a copy of the RST **smartegm-config.xml** file, located in **RadBlue System Tester** > **radblue** > **gsa** > **script** > **smart-conf** > **smart-egm**.
5. Give the new file a unique file name.
6. Open the new file with an XML editor or Wordpad.



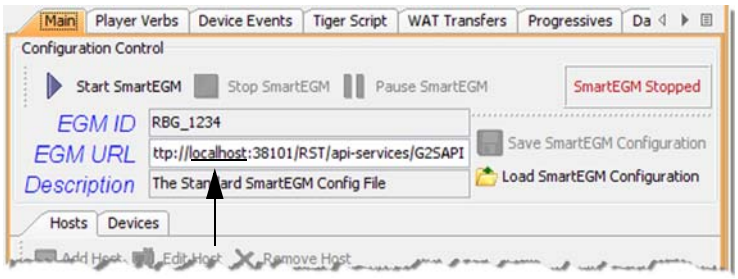
7. Change the URL of **host 1** from "localhost" to the IP address of the computer *you want to connect with* (in other words, where RGS is installed). *Do not* change the port number of the web address.

   To find the IP address of the computer that RGS is installed on, click **RGS URLs** on the Engine layout of the RGS application. For example:



8. Save the file.
9. Launch RST with the **SmartEGM** desktop.

10. Load the new configuration file into the SmartEGM.



11. Change the "localhost" of the **EGM URL** field entry with the IP address of the computer on which RST *is installed.*

To find the IP address of the computer that RST is installed on, do one of the following:

a. Go to a command line and type **ipconfig**.

b. Go to **Configure** > **Engine Options**, scroll down to the **IP Address & SOAP Port** section, and click the **Bind To** drop-down arrow. The computer's IP address appears in the drop-down. Note that "127.0.0.1" is the localhost.

You can make the EGM URL change permanent by going to **Configure** > **Engine Options** and modifying the **My URL** field. Simply replace "localhost" with the IP address of the computer on which RST is installed.

12. Start the SmartEGM engine.

If all changes were made correctly, you should now see communications flowing between RST and RGS. Review the Debug Log in each tool for any errors.

Activity 2: Add RPA Between RGS and RST

Whether RPA is installed on the same computer as RGS and/or RST, the configuration procedure is the same: configure the From URL in RPA and load the RPA version of the SmartEGM configuration file in RST.

1. Launch RPA.
2. Select **Configure** from the menu bar.
3. Select **Engine Options**.
4. Set the **IP Address and SOAP Port** number.

   The **Bind To** field is IP address that RPA uses for communications. The **SOAP Port** and **SSL SOAP Port** are the ports that RPA uses for communications. In most cases, you will use the default port numbers.

   - Click the drop-down arrow, and select the **IP Address** that you want RPA to use for communications.
5. Set the **Protocol Options**.

   This option lets you define the GSA protocol RPA uses. The **Protocol Name** and **Version** values are used to create the **Schema Location**.

   - Verify that the **Protocol Name** is **G2S** and the **Version** is **1.0.3**.
6. Click **OK**.
7. On the RPA layout, click **Host Side Information**.
8. Look at the **RPA's From Address** field and verify that it displays the following URL: **http://[IP address]/RPA/api-services/G2SHost**
9. Navigate to the RST installation directory and go to: radblue > gsa > script > smart-conf > smart-egm.
10. Right-click the **smartegm-config-rpa.xml** file (for standard licenses) or the **smartegm-config-student-edition-rpa.xml** file (for student licenses) and select **Edit**.
11. Change the following line to reflect the RPA IP address:

```
<edm:host edm:host-id="1" edm:url="http://[RPA IP Address]:35101/RPA/api-services/G2SEgm"
                edm:description="RadBlue Protocol Analyzer" edm:host-registered="true" />
```

12. **Save** and **close** the file.
13. Launch RST.

14. Click the **Main** tab on the **SmartEGM** layout, if it is not already selected.
15. Click **Change SmartEGM Configuration**.
16. Select **smartegm-config-rpa.xml** to highlight it. If you are using a student license, select **smartegm-config-student-edition-rpa.xml**.
17. Click **Open**.
18. Click **Start SmartEGM**.
19. Go to the RPA, and click **Start RPA**.

Activity 3: Generate the Transcript Analysis Report (RGS, RST or RPA)

The EGM Transcript Analysis report provides information about messages that were sent from and received by the application for the period requested. The purpose of this report is to provide the user with a user-friendly summary of G2S messages. The EGM Transcript Analysis report should be the first place you look when you want to know whether a test worked or not.

In addition, the EGM Transcript Analysis report can be sent to RadBlue to facilitate troubleshooting. Save the file from your browser, or use the browser's e-mail feature to e-mail the report directly to RadBlue.

The report is divided into several sections to assist your analysis of the information:

- **Transcript Summary** - Information related to the computer running the installed application.
- **Transcript** - Transaction log of sent and received G2S and S2S messages. G2S acknowledgements and, optionally, `keepAlive` messages are filtered out.
- **Sessions** - Transcript messages grouped by session ID.
- **Device Commands** - Transcript messages grouped by affected G2S device.
- **Device States** - Status of each device and any device status changes for the requested time period.
- **Events** - Events generated by the EGM.
- **G2S ACK Errors** - G2S acknowledgement messages containing errors.
- **Meters** - Meter values.
- **Messages** - G2S XML messages from the transcript.

The EGM Transcript Analysis report can be output to an .html page or to an .xml file.

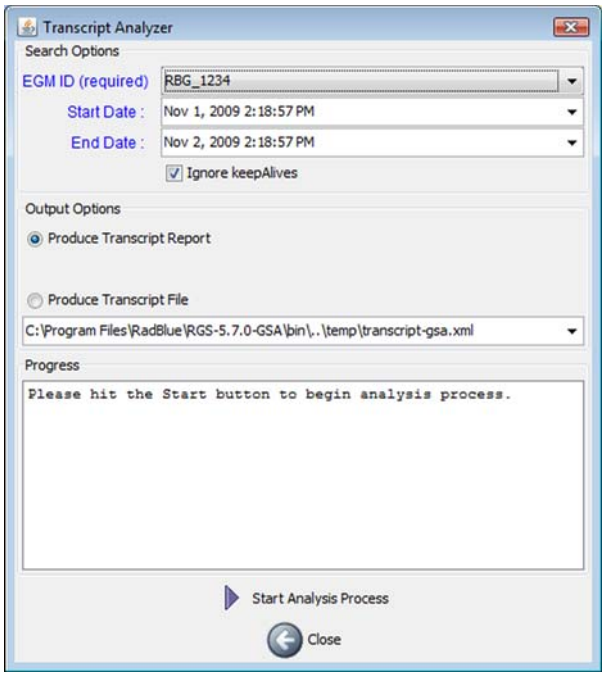
Navigating the Report File

The top of the report contains links to each section.

- Click a link to move through the file.
- Click the browser's "Back" button to return to the previous location in the file.

Generate the Report

1. Select the Transcript Control in RGS, RST *or* RPA.
2. Click **Analyze**.



- **EGM ID (required)** - *RGS and RPA only*. Click the drop-down arrow, and select the EGM on which you want to report.
- **Start Date** - Click the drop-down arrow to select the beginning date and time of the reporting period.
- **End Date** - Click the drop-down arrow to select the ending date and time of the reporting period.
- **Ignore keepAlives** - Select if you do not want to see `keepAlive` and `keepAliveAck` commands on your report.
- **Output Options** - Select **Produce Transcript Report** to export the Transcript Analysis report directly to an.html page, *or* select **Produce Transcript File** to export the Transcript Analysis report to an .xml file. If you export the report to an .xml file, you can accept the default output location or browse to the location of your choice.

3. Click **Start Analysis Process**.

Sample Report

Transcript Summary Section

The **Transcript Summary** section shows information related to the computer running the installed application.

EGM ID	
OS Architecture	x86
OS Name	Windows XP
OS Version	5.1
OS Patch Level	Service Pack 2
CPU List	pentium_pro+mmx pentium_pro pentium+mmx pentium i486 i386 i86
MAC Addresses	• 00-0F-FE-95-8C-2C
Java Version	1.6.0
JVM Version	1.6.0-b105
Java Home	C:\Program Files\RST-1.1.14-igt\radblue\jre\jre
Username	radblue
Started On	2008-04-24T16:09:07.572-07:00

Transcript Section

The **Transcript** section shows sent and received messages. Note that since the date is from the G2S message (sent either by the EGM or the host), the date may be incorrect. There are a number of reasons why this occurs. For example:

- the PC clock is incorrect.
- the Network Time Protocol (NTP) is not functioning correctly.
- there are programming errors.

Serial Number	Date/Time	Date/Time Sent	Direction	Command ID	Session ID	Session Type	Retry?	Device ID	Summary
1	2008-04-24T09:18:52.853-07:00	2008-04-24T09:18:52.853-07:00	OUTBOUND	2237	4000001	G2S_request		communications[1]	communications.commsOnLine
3	2008-04-24T09:18:54.978-07:00	2008-04-24T09:18:54.963-07:00	INBOUND	821	4000001	G2S_response		communications[1]	communications.commsOnLineAck
5	2008-04-24T09:18:55.463-07:00	2008-04-24T09:18:55.463-07:00	OUTBOUND	2238	4000002	G2S_request		communications[1]	communications.commsDisabled
8	2008-04-24T09:18:55.525-07:00	2008-04-24T09:18:55.525-07:00	INBOUND	822	4000002	G2S_response		communications[1]	communications.commsDisabledAck
9	2008-04-24T09:18:55.541-07:00	2008-04-24T09:18:55.541-07:00	INBOUND	823	723	G2S_request		communications[1]	communications.getDescriptor
11	2008-04-24T09:18:55.666-07:00	2008-04-24T09:18:55.666-07:00	OUTBOUND	2239	723	G2S_response		communications[1]	communications.descriptorList
14	2008-04-24T09:18:56.244-07:00	2008-04-24T09:18:56.244-07:00	INBOUND	824	724	G2S_request		eventHandler[1]	eventHandler.getEventHandlerStatus
15	2008-04-24T09:18:56.322-07:00	2008-04-24T09:18:56.322-07:00	OUTBOUND	2240	724	G2S_response		eventHandler[1]	eventHandler.eventHandlerStatus
18	2008-04-24T09:18:56.416-07:00	2008-04-24T09:18:56.416-07:00	INBOUND	825	725	G2S_request		eventHandler[1]	eventHandler.getSupportedEvents
19	2008-04-24T09:18:56.463-07:00	2008-04-24T09:18:56.463-07:00	OUTBOUND	2241	725	G2S_response		eventHandler[1]	eventHandler.supportedEvents
22	2008-04-24T09:18:57.713-07:00	2008-04-24T09:18:57.713-07:00	INBOUND	826	726	G2S_request		eventHandler[1]	eventHandler.getEventHandlerProfile
23	2008-04-24T09:18:57.744-07:00	2008-04-24T09:18:57.744-07:00	OUTBOUND	2242	726	G2S_response		eventHandler[1]	eventHandler.eventHandlerProfile
26	2008-04-24T09:18:57.838-07:00	2008-04-24T09:18:57.838-07:00	INBOUND	827	727	G2S_request		eventHandler[1]	eventHandler.getEventSub
27	2008-04-24T09:18:57.885-07:00	2008-04-24T09:18:57.885-07:00	OUTBOUND	2243	727	G2S_response		eventHandler[1]	eventHandler.eventSubList
29	2008-04-24T09:18:58.041-07:00	2008-04-24T09:18:58.041-07:00	INBOUND	828	728	G2S_request		eventHandler[1]	eventHandler.setEventHandlerState
31	2008-04-24T09:18:58.229-07:00	2008-04-24T09:18:58.229-07:00	OUTBOUND	2244	728	G2S_response		eventHandler[1]	eventHandler.eventHandlerStatus
34	2008-04-24T09:18:58.307-07:00	2008-04-24T09:18:58.307-07:00	INBOUND	829	729	G2S_request		meters[1]	meters.getMeterSub
35	2008-04-24T09:18:58.353-07:00	2008-04-24T09:18:58.353-07:00	OUTBOUND	2245	729	G2S_response		meters[1]	meters.meterSubList

The **Serial Number** is assigned by the tool. This number corresponds to the order in which the message was received by the application. Use this value to compare the arrival order between any two messages.

The **Date/Time Sent** field is the date and time message that is sent to the host or EGM.

The **Date/Time** field is the date and time message was constructed.

The **Direction** is relative to the application. *Inbound* means a message that came from the other end. *Outbound* is a message that was generated by the application you ran the report from.

The **Retry?** column is blank for first attempts and filled in for retries. This column corresponds to the *sessionRetry* attribute in the G2S message.

Sessions Section

The **Sessions** section groups sent and received messages. Messages appear in ascending order, by **Session ID**.

A Session ID is a number that is set by the sender of a request, which allows the response to be tied to the original request. While session ID numbers should increase, they don't have to - they must be unique for the number of outstanding messages.

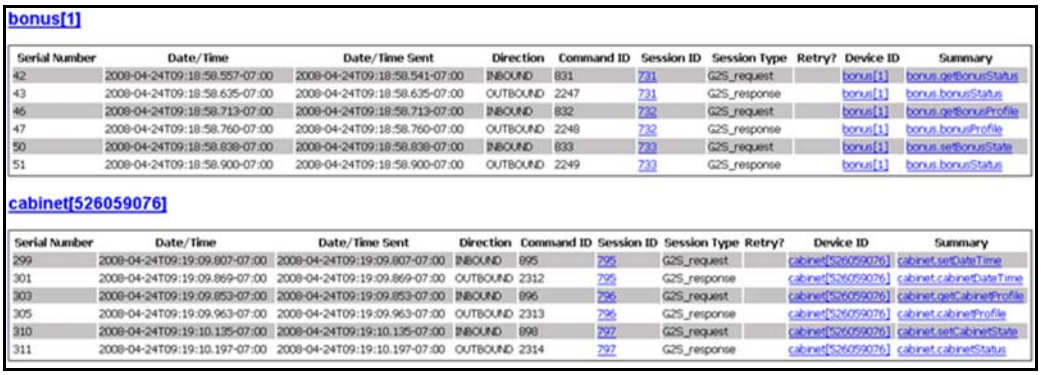
Entries in red indicate incomplete sessions (either no response or too many responses) and are often errors.

Session ID	Messages
723	communications.getDescriptor communications.descriptorList
724	eventHandler.getEventHandlerStatus eventHandler.eventHandlerStatus
725	eventHandler.getSupportedEvents eventHandler.supportedEvents
726	eventHandler.getEventHandlerProfile eventHandler.eventHandlerProfile
727	eventHandler.getEventSub eventHandler.eventSubList
728	eventHandler.setEventHandlerState eventHandler.eventHandlerStatus
729	meters.getMeterSub meters.meterSubList
730	meters.getMeterSub meters.meterSubList
731	bonus.getBonusStatus bonus.bonusStatus

Device Commands Section

The **Device Commands** section contains sent and received transcript messages grouped by individual G2S devices.

If Session ID numbers are reused, each row will contain all of the messages with the same Session ID. Currently, the report does not display legal pairs.



bonus[1]

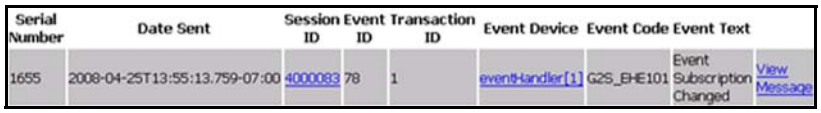
Serial Number	Date/Time	Date/Time Sent	Direction	Command ID	Session ID	Session Type	Retry?	Device ID	Summary
42	2008-04-24T09:18:58.557-07:00	2008-04-24T09:18:58.541-07:00	INBOUND	831	731	G2S_request		bonus[1]	bonus.getBonusStatus
43	2008-04-24T09:18:58.635-07:00	2008-04-24T09:18:58.635-07:00	OUTBOUND	2247	731	G2S_response		bonus[1]	bonus.bonusStatus
46	2008-04-24T09:18:58.713-07:00	2008-04-24T09:18:58.713-07:00	INBOUND	832	732	G2S_request		bonus[1]	bonus.getBonusProfile
47	2008-04-24T09:18:58.760-07:00	2008-04-24T09:18:58.760-07:00	OUTBOUND	2248	732	G2S_response		bonus[1]	bonus.bonusProfile
50	2008-04-24T09:18:58.838-07:00	2008-04-24T09:18:58.838-07:00	INBOUND	833	733	G2S_request		bonus[1]	bonus.setBonusState
51	2008-04-24T09:18:58.900-07:00	2008-04-24T09:18:58.900-07:00	OUTBOUND	2249	733	G2S_response		bonus[1]	bonus.bonusStatus

cabinet[526059076]

Serial Number	Date/Time	Date/Time Sent	Direction	Command ID	Session ID	Session Type	Retry?	Device ID	Summary
299	2008-04-24T09:19:09.807-07:00	2008-04-24T09:19:09.807-07:00	INBOUND	895	795	G2S_request		cabinet[526059076]	cabinet.setDateTime
301	2008-04-24T09:19:09.869-07:00	2008-04-24T09:19:09.869-07:00	OUTBOUND	2312	795	G2S_response		cabinet[526059076]	cabinet.cabinetDateTime
303	2008-04-24T09:19:09.853-07:00	2008-04-24T09:19:09.853-07:00	INBOUND	896	796	G2S_request		cabinet[526059076]	cabinet.getCabinetProfile
305	2008-04-24T09:19:09.963-07:00	2008-04-24T09:19:09.963-07:00	OUTBOUND	2313	796	G2S_response		cabinet[526059076]	cabinet.cabinetProfile
310	2008-04-24T09:19:10.135-07:00	2008-04-24T09:19:10.135-07:00	INBOUND	898	797	G2S_request		cabinet[526059076]	cabinet.setCabinetState
311	2008-04-24T09:19:10.197-07:00	2008-04-24T09:19:10.197-07:00	OUTBOUND	2314	797	G2S_response		cabinet[526059076]	cabinet.cabinetStatus

Events

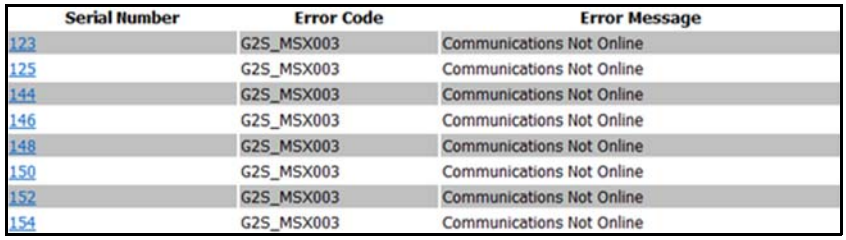
The **Events** section displays information about events generated by the EGM.



Serial Number	Date Sent	Session ID	Event ID	Transaction ID	Event Device	Event Code	Event Text	
1655	2008-04-25T13:55:13.759-07:00	4000083	78	1	eventHandler[1]	G2S_EHE101	Event Subscription Changed	View Message

G2S ACKs That Have Errors

The **G2S ACKs That Have Errors** section shows G2S acknowledgement messages containing errors.



Serial Number	Error Code	Error Message
123	G2S_MSX003	Communications Not Online
125	G2S_MSX003	Communications Not Online
144	G2S_MSX003	Communications Not Online
146	G2S_MSX003	Communications Not Online
148	G2S_MSX003	Communications Not Online
150	G2S_MSX003	Communications Not Online
152	G2S_MSX003	Communications Not Online
154	G2S_MSX003	Communications Not Online

Meters

The **Meters** section shows the values of the following meters:

- PCA - Player Cashable Amount (G2S_playerCashableAmt)
- PPA - Player Promo Amount (G2S_playerPromoAmt)
- PNCA - Player Non-Cashable Amount (G2S_playerNonCashAmt)
- GSI - Games Since Initialization (G2S_gamesSinceInitCnt)

Note that this is not a comprehensive list of all possible meters.

Serial Number	Date Sent	Session ID	PCA	PPA	PNCA	GSI	
395	2008-04-25T13:15:17.131-07:00	1006	20.00.000	0.00.000	0.00.000	0	View Message
401	2008-04-25T13:16:18.165-07:00	1007	13.00.000	0.00.000	0.00.000	1	View Message

Messages

The **Messages** section displays the actual XML code for messages that appear in the transcript.

The report may rewrite the XML content while formatting this section of the report. In particular, the report may change the XML namespace prefix. However, the XML namespace URI is maintained. Note that 11 MAR 2010 - R4unused namespace declarations may not be displayed.

Message #1 communications.commsOnLine

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<g2s:g2sMessage xmlns:g2s="http://www.gamingstandards.com/g2s/schemas/v1.0.3">
   <g2s:g2sBody g2s:dateTimeSent="2008-04-24T09:18:52.853-07:00" g2s:egmId="RBG_1234"
                g2s:hostId="1">
      <g2s:communications g2s:commandId="2237" g2s:dateTime="2008-04-24T09:18:52.853-07:00"
                          g2s:deviceId="1"
                          g2s:errorCode="G2S_none"
                          g2s:errorText=""
                          g2s:sessionId="4000001"
                          g2s:sessionMore="false"
                          g2s:sessionRetry="false"
                          g2s:sessionType="G2S_request"
                          g2s:timeToLive="30000">
         <g2s:commsOnLine g2s:deviceChanged="true" g2s:deviceReset="true"
                          g2s:egmLocation="http://localhost:38101/RST/api-services/G2SAPI"
                          g2s:equipmentType="G2S_egm"
                          g2s:metersReset="true"
                          g2s:subscriptionLost="true"/>
      </g2s:communications>
   </g2s:g2sBody>
</g2s:g2sMessage>
```

Activity 4: Review Changes in the EGM's Data Model (RST)

Snapshot and Compare lets you capture the current EGM data model and compare the options, meter, and status values from two snapshots. This allows you to see data model changes that occur during testing.

When you compare two snapshots, the information is displayed in a separate Compare Snapshot Deltas tab and displayed by information type (options, meters and status). You can add comments to the comparison output and print the current view of the active tab.

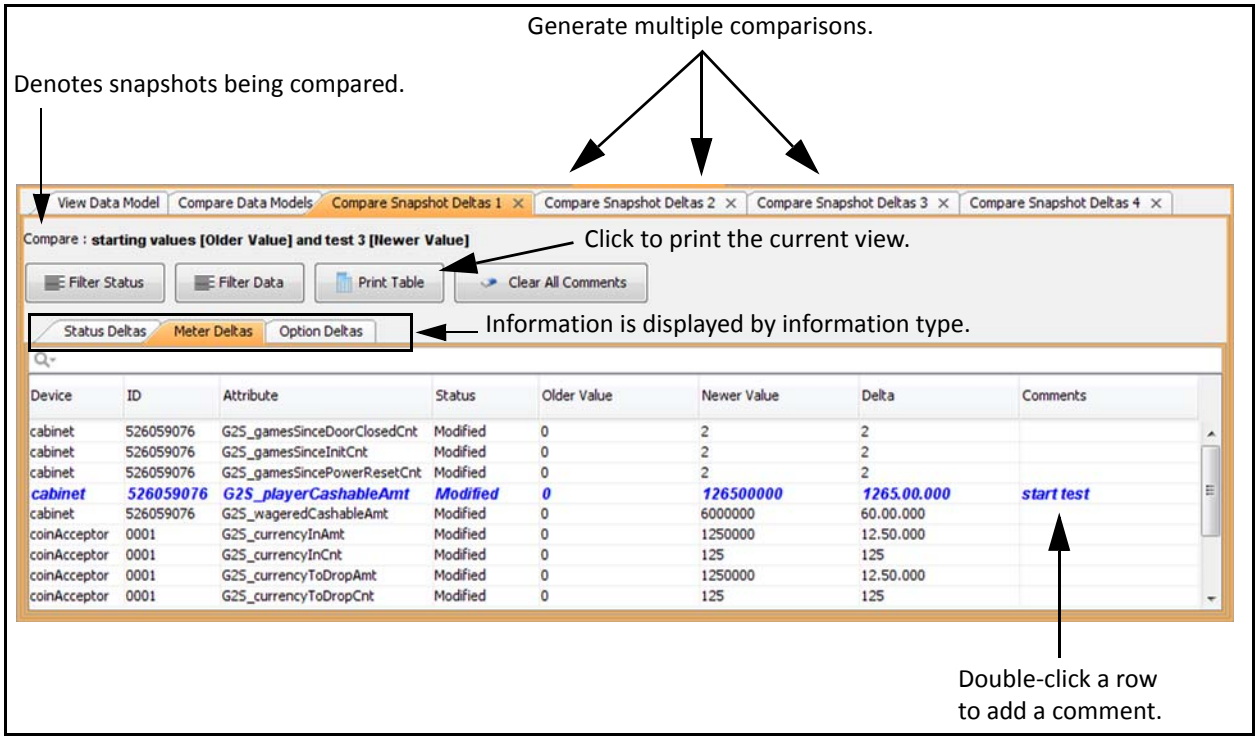
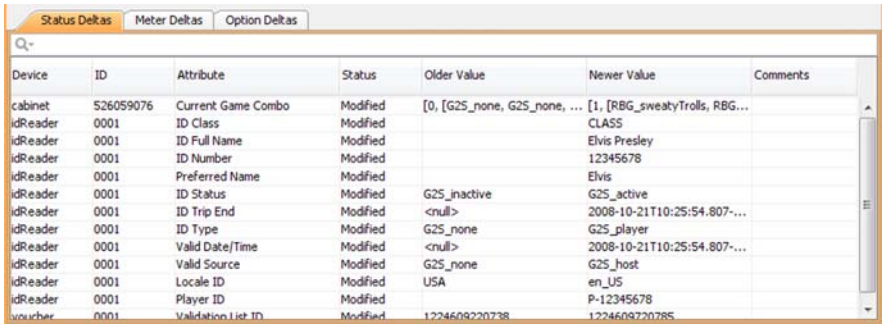


Figure 1: Snapshot and Compare comparison report.

Each snapshot comparison generates three separate comparisons: Status Deltas, Meter Deltas and Option Deltas. The information displayed on each comparison includes: device name, device identifier, attribute name, comparison status, attribute value from the older snapshot, attribute value from the newer snapshot, and user-entered comments.

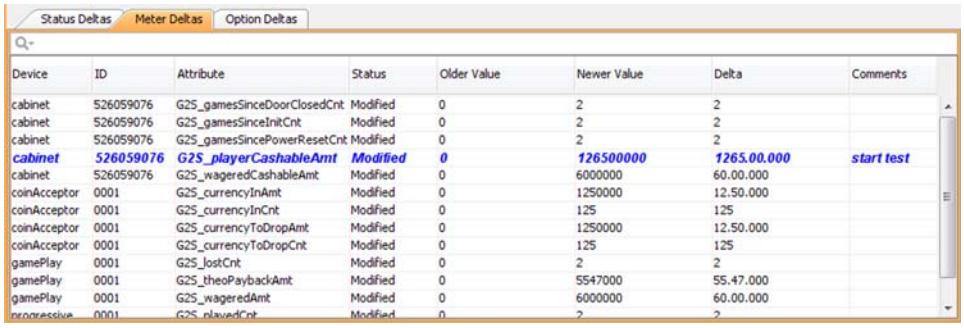
The Status Deltas comparison displays status parameters from the selected EGM snapshots.



Device	ID	Attribute	Status	Older Value	Newer Value	Comments
cabinet	526059076	Current Game Combo	Modified	[0, [G2S_none, G2S_none, ...	[1, [RBG_sweatyTrolls, RBG...	
idReader	0001	ID Class	Modified		CLASS	
idReader	0001	ID Full Name	Modified		Elvis Presley	
idReader	0001	ID Number	Modified		12345678	
idReader	0001	Preferred Name	Modified		Elvis	
idReader	0001	ID Status	Modified	G2S_inactive	G2S_active	
idReader	0001	ID Trip End	Modified	<null>	2008-10-21T10:25:54.807-...	
idReader	0001	ID Type	Modified	G2S_none	G2S_player	
idReader	0001	Valid Date/Time	Modified	<null>	2008-10-21T10:25:54.807-...	
idReader	0001	Valid Source	Modified	G2S_none	G2S_host	
idReader	0001	Locale ID	Modified	USA	en_US	
idReader	0001	Player ID	Modified		P-12345678	
voucher	0001	Validation List ID	Modified	1224609220738	1224609720785	

Figure 2: Status Deltas comparison.

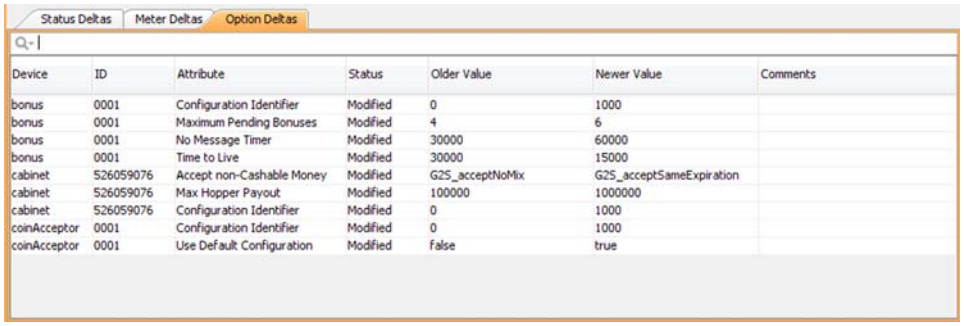
The Meter Deltas comparison displays meter parameters from the selected EGM data models. The Meter Deltas comparison contains an additional Delta field that shows the meter difference between the older and newer snapshots. In the Delta field, amount differences are displayed with the appropriate decimals to make entries easier to read.



Device	ID	Attribute	Status	Older Value	Newer Value	Delta	Comments
cabinet	526059076	G2S_gamesSinceDoorClosedCnt	Modified	0	2	2	
cabinet	526059076	G2S_gamesSinceInitCnt	Modified	0	2	2	
cabinet	526059076	G2S_gamesSincePowerResetCnt	Modified	0	2	2	
cabinet	***526059076***	***G2S_playerCashableAmt***	***Modified***	***0***	***126500000***	***1265.00.000***	***start test***
cabinet	526059076	G2S_wageredCashableAmt	Modified	0	6000000	60.00.000	
coinAcceptor	0001	G2S_currencyInAmt	Modified	0	1250000	12.50.000	
coinAcceptor	0001	G2S_currencyInCnt	Modified	0	125	125	
coinAcceptor	0001	G2S_currencyToDropAmt	Modified	0	1250000	12.50.000	
coinAcceptor	0001	G2S_currencyToDropCnt	Modified	0	125	125	
gamePlay	0001	G2S_lostCnt	Modified	0	2	2	
gamePlay	0001	G2S_theoPaybackAmt	Modified	0	5547000	55.47.000	
gamePlay	0001	G2S_wageredAmt	Modified	0	6000000	60.00.000	
progressive	0001	G2S_playedCnt	Modified	0	2	2	

Figure 3: Meter Deltas comparison.

The Option Deltas comparison displays option parameters from the selected EGM data models.

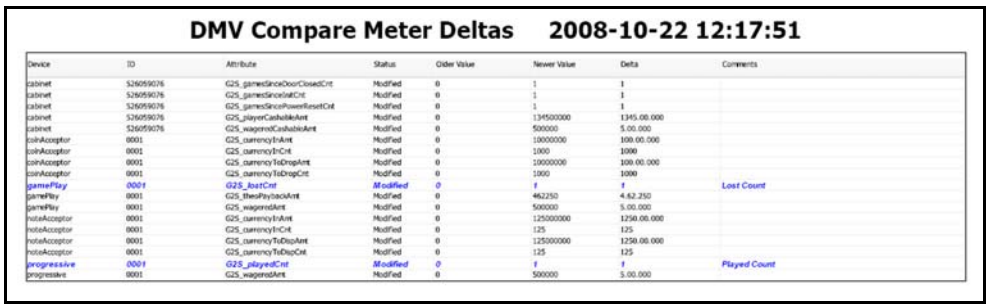


Device	ID	Attribute	Status	Older Value	Newer Value	Comments
bonus	0001	Configuration Identifier	Modified	0	1000	
bonus	0001	Maximum Pending Bonuses	Modified	4	6	
bonus	0001	No Message Timer	Modified	30000	60000	
bonus	0001	Time to Live	Modified	30000	15000	
cabinet	526059076	Accept non-Cashable Money	Modified	G2S_acceptNoMix	G2S_acceptSameExpiration	
cabinet	526059076	Max Hopper Payout	Modified	100000	1000000	
cabinet	526059076	Configuration Identifier	Modified	0	1000	
coinAcceptor	0001	Configuration Identifier	Modified	0	1000	
coinAcceptor	0001	Use Default Configuration	Modified	false	true	

Figure 4: Option Deltas comparison.

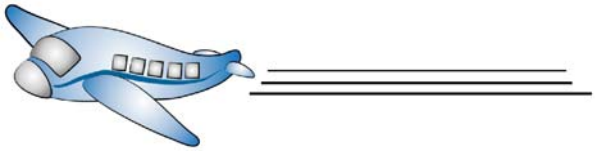
1. Take an initial snapshot of the EGM's data model.
   a. Select the **View Data Model** tab from the SmartEGM Data Model Viewer.
   b. Click **Snapshot**.
   c. Enter a name for the snapshot (for example, "test 1"), and click **Capture**.
   d. Read the confirmation message, and click **OK**.
2. Go to **Compare Data Models** tab, and verify that your snapshot appears under **Available Snapshots**.
3. Use the SmartEGM **Player Verbs** to create activity.
   a. Insert notes.
   b. Play a couple simple games.
   c. Create a voucher.
4. Take another snapshot of the EGM's data model.
   a. Select the **View Data Model** tab from the SmartEGM Data Model Viewer.
   b. Click **Snapshot**.
   c. Enter a name for the snapshot (for example, "test 2"), and click **Capture**.
   d. Read the confirmation message, and click **OK**.
5. Generate a comparison of your snapshots.
   a. Select the **Compare Data Models** tab from the Data Model Viewer.
   b. Hold down **CTRL**, and click the two snapshots you want to compare.
   c. Click **Compare** to generate the **Compare Snapshot Deltas** screen.
6. Print a comparison report.

   Note that you can only print the active Deltas tab of the current Compare Snapshot Detail.
   a. Right-click in the header of the comparison table you want to print, and select **Auto Resize All Columns**.
   b. Click **Print Table**.
   c. Set the print properties as needed, and click **Print**.


DMV Compare Meter Deltas 2008-10-22 12:17:51

Device	ID	Attribute	Status	Older Value	Newer Value	Delta	Comments
cabinet	526059076	G2S_gamesSinceDoorClosedCnt	Modified	0	1	1	
cabinet	526059076	G2S_gamesSinceInitCnt	Modified	0	1	1	
cabinet	526059076	G2S_gamesSincePowerResetCnt	Modified	0	1	1	
cabinet	526059076	G2S_playerCashableAmt	Modified	0	134500000	1345.00.000	
cabinet	526059076	G2S_wageredCashableAmt	Modified	0	500000	5.00.000	
coinAcceptor	0001	G2S_currencyInAmt	Modified	0	10000000	100.00.000	
coinAcceptor	0001	G2S_currencyInCnt	Modified	0	1000	1000	
coinAcceptor	0001	G2S_currencyToDropAmt	Modified	0	10000000	100.00.000	
coinAcceptor	0001	G2S_currencyToDropCnt	Modified	0	1000	1000	
gamePlay	0001	G2S_lostCnt	Modified	0	1	1	Lost Count
gamePlay	0001	G2S_thesPaybackAmt	Modified	0	462250	4.62.250	
gamePlay	0001	G2S_wageredAmt	Modified	0	500000	5.00.000	
noteAcceptor	0001	G2S_currencyInAmt	Modified	0	125000000	1250.00.000	
noteAcceptor	0001	G2S_currencyInCnt	Modified	0	125	125	
noteAcceptor	0001	G2S_currencyToDispAmt	Modified	0	125000000	1250.00.000	
noteAcceptor	0001	G2S_currencyToDispCnt	Modified	0	125	125	
progressive	0001	G2S_playedCnt	Modified	0	1	1	Played Count
progressive	0001	G2S_wageredAmt	Modified	0	500000	5.00.000	

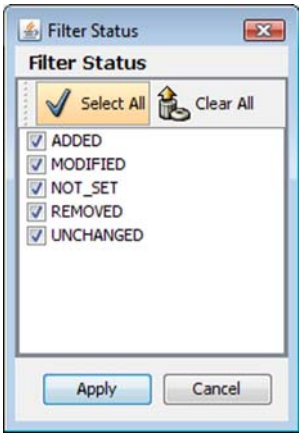
Flying Solo Activity 1



There are three ways to filter comparison data in Snapshot and Compare - Filter Status, Filter Data and Quick Filter. Here are some tasks to familiarize you with each filter.

Task 1. Use **Filter Status** to filter information *by comparison status*.

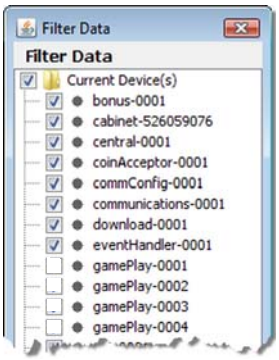
Click **Filter Status**. Click **Clear All** to clear all status filters, and click **Apply**. How does this change what you see in the comparison?



Task 2. Now, use the following Filter Status option definitions to see how each affects your view of the comparison information:

- **Select All** - Click to select all statuses. Selected statuses are included in the comparison.
- **Clear All** - Click to clear all statuses. Cleared statuses are filtered out of the comparison.
- **Added** - Select to show device parameters with information that was added from the older snapshot to the newer snapshot.
- **Modified** - Select to show device parameters with information that was modified from the older snapshot to the newer snapshot.
- **Not Compared** - Select to show device parameters that were not compared. This status may indicate that an error occurred during comparison or that the values could not be compared.
- **Removed** - Select to show device parameters that have been removed from the data model from the older snapshot to the newer snapshot.
- **Unchanged** - Select to show device parameters with no changes from the older snapshot to the newer snapshot. By default, this status is cleared.

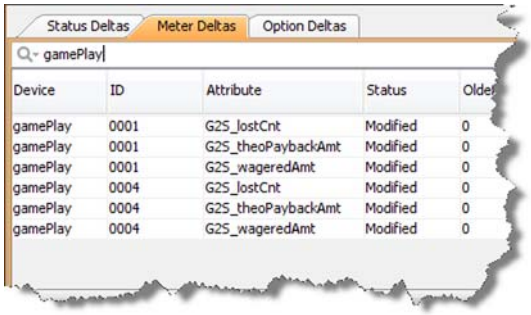
Task 3. Use the **Filter Data** option to filter information *by device*.



Click **Filter Data**, clear the check boxes for all gamePlay devices, and click **Apply**. Look at how the comparison report has changed.

Task 4. Use the **Quick Filter** option to dynamically filter comparison data *by keyword(s)*.

Task 5. Click inside the **Quick Filter** box, and type a keyword(s).



Task 6. Click  to refine your filter. Select one of the following filter options, and then type your keyword(s):

- **All** - Select to filter on all columns.
- **Device** - Select to filter on devices only.
- **Attribute** - Select to filter on attributes only.
- **Case sensitive** - Select to consider title case in your filter entry.
- **Case insensitive** - Select to disregard title case in your filter entry.
- **Use wild cards** - Select to allow the use of wild cards in your filter entry. Acceptable wild cards are "?" (match any letter or digit) and "*" (match several letters or digits).
- **Match from start** - Select to filter from the start of your entry.
- **Match anywhere** - Select to filter on any part of your entry.