rad**blue**

Release Date: 09 December 2014

Release Summary

In this release of the RST, we made several improvements to the User Interface, made it easy to expand the view message and message compare screens in the G2S transcript, enhanced the pause and insertNote verbs in the tiger scripting language, and made extensive improvements to the message modification function in the RST Tester Toolkit module.

We hope you enjoy this new release. If you have any ideas for improvement that we may have missed, just let us know at support@radblue.com.

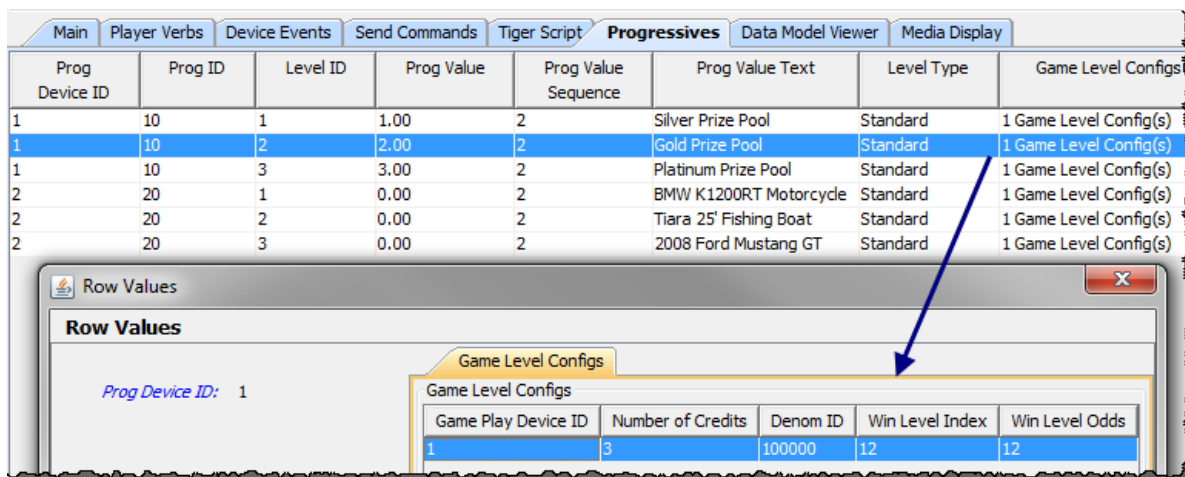The following sections provide details for each of the significant modifications and improvements.

SmartEGM Modifications

- **Added validation for the `commsOnLine`.*egmLocation* attribute** – When the Smart EGM is started, it now verifies that the URL in the *egmLocation* attribute of the `commsOnLine` is one on which the RST is listening, and also one that the G2S host can access. Subsequent commsOnline request by-pass this logic until the SmartEGM is restarted.

- **`commsDisabledAck` now updates the sync timer** – Previously, the SmartEGM engine gracefully accepted a `commsDisabledAck` from the host, and then happily ignored the *syncTimer* attribute in that message. In this new release of the SmartEGM, if a new *syncTimer* value is received, the SmartEGM uses that new value, starting a new timer for the new value. If the *syncTimer* value is less than 15 seconds, an error is logged, and the *syncTimer* is set to 15 seconds.

- **Configuration devices now use proper values for missing attributes** – commConfig, optionConfig and deviceConfig devices have no *egmEnabled*, *hostEnabled*, or *hostLocked* flags in their respective status objects. We recently learned the descriptor list for these devices MUST use the default values for the missing attributes, so the `descriptorList` now reflects that interpretation (*egmEnabled* =true, *hostEnabled* = true, *hostLocked*=false, *egmLocked*=false).
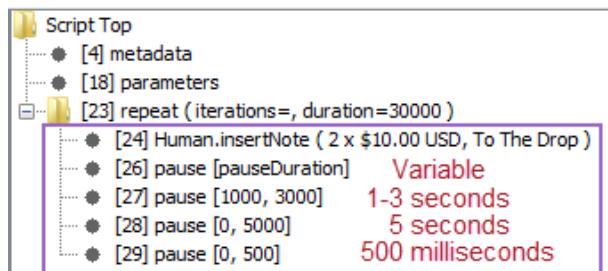
  Also, the deprecated *egmLocked* flag is now set to true only if the device ACTUALLY has the EGM locked

- In the past, the SmartEGM attempted to send an unacknowledged `gat.verificationResult` request to the host while it was still in the sync state (before communications were enabled by the G2S Host). It now waits until communications are enabled before retrying the request.

User Interface Modifications

- **Simplified Progressive Table Display** – The game level configuration columns were removed from the RST's progressive panel to simplify the table. You can see the game level configuration details of each progressive level by double clicking the row and examining the resulting row values display (see below).

- **Tiger Script Tree Display Enhancements** – The Tiger Script tree provides an overview of what is going on in an EGM Script. This display was enhanced to include the insertNote count attribute and to show the duration of a pause statement (variable or fixed). See the screenshot, below, for an example of these improvements:

Transcript Modifications

- The **G2S transcript** was enhanced by adding **Max** and **Restore** buttons to the View Transcript Message window (launched when you double click a message in the transcript). Since these buttons are located in the top frame of the control, they are available in the Command View and the XML View of the message. This feature is most useful when you encounter a really large message.

Max – expands the screen to the full size of the home screen of the computer.

Restore – Restores the window to its previous size and location.

- The same **Max** and **Restore** button functionality was added to the **Compare** button in the transcript. This feature is used to compare two similar messages in the transcript, so it was also a good candidate for a maximize feature.

Tiger Scripting Modifications

- **Human.insertNote verb now includes a *count* attribute** – The **tiger:Human.insertNote** verb simulates the player inserting a note into a note acceptor device. On the RST GUI, you can specify how many notes should be inserted, so a new *count* attribute has been to the insertNote verb so you can specify the count. For backwards compatibility, the default value for this new attribute is "1". In the following example, five $10 notes are inserted into the EGM:

```
<tiger:Human.insertNote tiger:count="5" tiger:denom-id="1000000" />
```
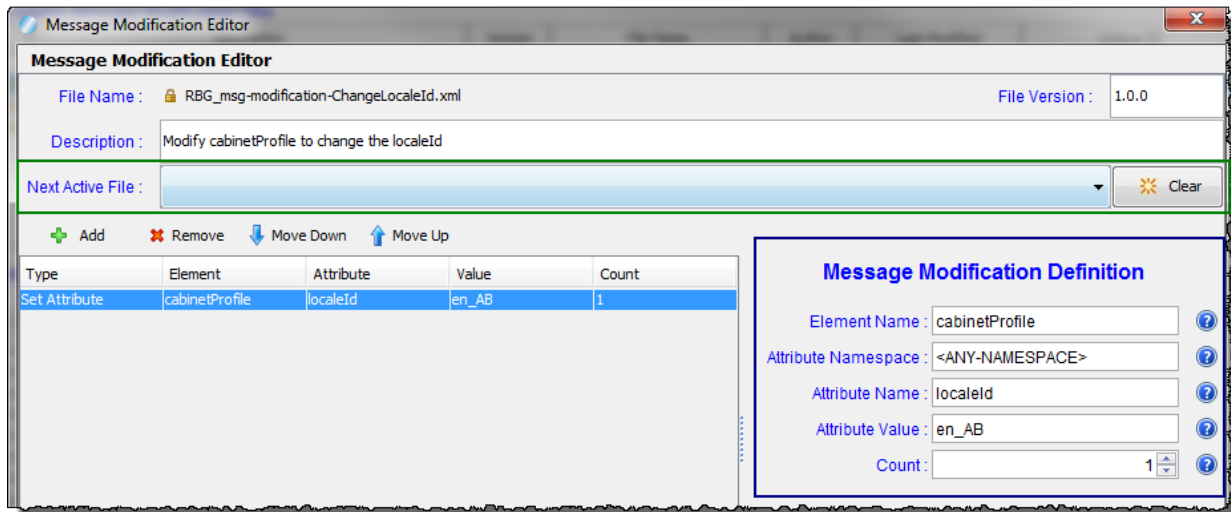
- **The core:pause verb now accepts a variable duration** – The **core:pause** verb is used to add fixed or random pauses as needed in a script. When creating large scripts to test systems, it became useful to be able to tune how long the script paused between each operation, via an *integerParameter* that can be changed by the tester at run-time without having to modify hundreds of pause statements in the script. In this new release of the scripting engine, you can now define an integer parameter (such as *pauseDuration*) that can then be modified as an execution parameter for the script at run-time. In the following example, a pauseDuration parameter is defined with a default value of 500 milliseconds, and then the new parameter is used in a pause statement:

```
<core:parameters>
   <core:integerParameter core:name="pauseDuration" core:default-value="500" />
</core:parameters>

<core:pause core:duration="pauseDuration"/>
```

Tester Toolkit Modifications

The Message Modification engine in the RST Tester Toolkit is used to modify G2S commands that are created by the RST, before they are sent to the G2S Host for whom they are intended. In this release, we made a number of modifications to the Message Modification Editor and Engine functionality, making the Message Modification tool much more useful.

1) In the Message Modification Editor, you can now specify the **Next Active File** (green box above). This feature lets you chain together a series of Message Modification files to create extended patterns – possibly modifying a couple of commands, then letting a few go through unmodified, then modifying a few more, etc.

2) To allow commands to pass through the Modification Engine without being affected, a new modification type of **NO_MODIFICATION** was created.

3) The Message Modification fields that are displayed are now tied to the Type of modification being performed.

    a) NO_MODIFICATION – Just specify how many times to ignore the command [COUNT]

    b) REMOVE_ATTRIBUTE – Specify Element, Attribute Name to remove (in that element), and Count

    c) SET_ATTRIBUTE – Adds the attribute if it's not present in the identified element, modifies the attribute if it is present. All Message Modification fields are used with the SET_ATTRIBUTE function.

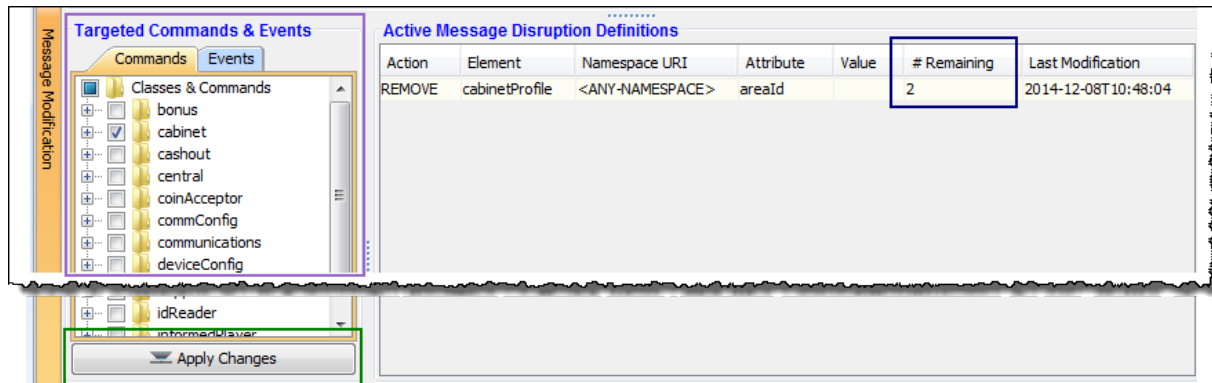4) Once a message modification file is activated, it will act on any command that matches a new **Targeted Commands & Events pre**-filter that was added to the run-time section of the Message Modification Control. Just select the commands and/or events that you want to affect, and then press the **Apply Changes** button to tell the engine to read in the new filter set. By default, no events or commands are selected, so make sure you select something!

5) As matches are acted upon by rules in the Active Message Modification Definitions, the # Remaining counter decrements for that rule until it reaches zero. If there are multiple rules in the Definition file that match the message being considered, they will all act on the message. Once all rules in the file reach 0, the Next Active File (see above) becomes active.

Release Date: 26 August 2014

Release Summary

In this release, we did quite a bit of work on the data model snapshot comparison utility – revisiting it to clean up a number of items that we discovered during our rewrite of the Quick Start Guide – our primer for G2S and our tools.

In the SmartEGM engine, we modified our support for the keepAlive timer to correspond with rules introduced in new versions of the protocol, tightened up our validation of progressive configurations, and added a new Tiger verb that allows you to cause an EGM to hit multiple progressive jackpots during a single game cycle.

We hope you enjoy this new release. If you have any ideas for improvement that we may have missed, just let us know at support@radblue.com.

The following sections provide details for each of the significant modifications and improvements.

Installer Modifications

- **All tools have been updated to use Java 1.7.0_65** – To keep up with the latest security (and other) improvements, all tools have been updated to Java 1.7.0_65. When installing this release, the Windows installer will download Java as part of the installation. If you are installing multiple tools on your computer, this Java upgrade is only done once. The Linux installer bundles the JRE in the installer.

SmartEGM Modifications

- **Updated keepAlive timer logic** – In the SmartEGM engine, we modified our support for the keepAlive timer to correspond with rules introduced in new versions of the protocol. Previously, the keepAlive timer was reset when any G2S message was sent to the host OR any G2S message was received from the host. Now, the keepAlive timer is only reset when messages occur on both channels (and also whenever a keepAlive is sent to the host). For more details on these new rules, please see the keepAlive command description in G2S protocol document for versions 2.x and greater.

- **Improved handpay behavior when device is hostdisabled** – The SmartEGM was properly disabling the handpay device on the `setHandpayState` command from the host, but it was not allowing any handpay functionality when the device was disabled. It now honors the `handpayProfile` "disabled" attributes that control which handpay functions are allowed when the device is disabled (for the functions that it supports).

SmartEGM Modifications (continued)

- Notifications from the SmartEGM were being sent when the communications channel is not enabled (EGM in sync state). Now they're just discarded if they can't be sent to the host.

- Average Payback Percentage was not being calculated properly, but was always being set to the average payback percentage for the current game cycle. This has been corrected.

- There was an occasional issue where the SmartEGM would not exit option configuration mode when an `enterOptionConfigMode` command with *enable*=**false** (disable option config mode) was received, due to a race condition in the code. This has been corrected.

- The SmartEGM configuration would allow a progressive device to be configured with progressive win levels that do not exist or that don't allow progressives.

  The Smart EGM configuration progressive validation has been updated to check the following:

  1. The game play device must have *gamePlayProfile.progressiveAllowed* set to **true**

  2. The gamePlay win level being referenced must have *progressiveAllowed* set to **true**

  3. The *numberOfCredits* value in the progData table must be less than *gamePlayProfile.maxWagerCredits* for the referenced gamePlay device.

  4. The *denomId* in the progData table has to be in the list of denominations for the gamePlay device (though it does not have to be active)

Note: If you are using a standard RadBlue configuration file, you may get an error when first starting the RST after doing this upgrade. Just accept the error, and then change the SmartEGM configuration file to use the new version that is distributed with the tool, and all will be good. This new progressive validation caught an error in some of our old SmartEGM configuration files.

DMV Compare modifications

The RST maintains a copy of the EGM data model used in the tool, updating all information that is available through the G2S protocol as activities occur in the SmartEGM (either through actions at the EGM, or via G2S commands from the host). The data model can be viewed on the Data Model Viewer (DMV) tab on the SmartEGM layout in the RST – just expand the tree to drill down to the details of any device in the data model.

A powerful feature of the DMV is the ability to snapshot a copy of the data model, and to compare any two snapshots to review the changes that occurred in the EGM between any two instants in time. Recently, while updating the Quick Start Guide, we did a detailed review of the DMV Compare functions in the RST and RGS and found a number of important improvements that were made in this release. Specifics on these changes follow:

- The comparison statement in the DMV Compare report is now a bit friendlier on the screen and on the report: "Comparing RBG_1234-snap1 [Older Value] to RBG_1234-snap2 [Newer Value]"

- The columns that can be used when Quick Filtering the reports have been improved to use the most relevant columns in each type of comparison (meters, profile, status, etc.)

- The DMV Compare Filter GUIs got some updated labels (these let you reduce the results by filtering out unwanted sets of data).

- Negative percentage values in DMV were not formatted properly (this can occur when comparing the average payback percentages between data model snapshots).

- The DMV Snapshot Comparison now correctly compares wager category, game denomination, and currency meters. A new "Group" column contains the denomination or wager category description making the result screen much easier to use.

- For reporting in this new Group column, Game denominations do not specify a currencyId as they use the currencyId of the cabinet. Currency meters now use the format "5.00 USD" so they are now relevant for all cases.

For more details on using this powerful utility – check out the Advanced Skills module (Module 6) in the Quick Start Guide.

Tiger Scripting Modifications

- **Multiple Progressive Hits per Game Cycle** – The SmartEGM now supports multiple progressive hits in one game cycle via a new `playSimpleGameWithProgressiveHit` verb. This new verb allows a series of **progressive-hit-data** elements, each containing a valid *win-level-index*, which will cause multiple progressive hits in one gamePlay cycle. Each element must be valid for the selected gamePlay device. If desired, a particular **progressive-hit-data** element can be repeated multiple times. The following is an example of how the verb is used:

```
<tiger:Human.playSimpleGameWithProgressiveHit tiger:device-id="3"
    tiger:denom-id="100000" tiger:credits-to-wager-cashable="5"
    tiger:primary-win="0">
        <tiger:progressive-hit-data tiger:win-level-index="1" />
        <tiger:progressive-hit-data tiger:win-level-index="1" />
        <tiger:progressive-hit-data tiger:win-level-index="2" />
</tiger:Human.playSimpleGameWithProgressiveHit>
```

In this example, *device-id* refers to the gamePlay device being played, as do the other attributes of the `playSimpleGameWithProgressiveHit` element. Once the primary game result has been determined, three progressive hits will occur – two for win level index 1, and then one for win level index 2.

The following validations are performed when this new Tiger verb is executed:

1. The referenced game play device's profile must allow progressives.

2. Each win level referenced by the progressive-hit-data element must allow progressives

3. For each referenced win level, the engine will verify that the combo of gamePlayId, winLevelIndex, denomId, numberOfCredits MUST be an entry in the associated progressive device's progressive data table.

Remote Control Modifications

For the specific details of each of these calls, please see the RST scratchpad. Just start the RST, and then access the scratchpad using a browser and the following URL:

http://localhost:38501/RST/pages/remotecontrol/ScratchPad.html

Note: The RST Remote Control is an optional REST based interface that lets you control the RST from your program. If this option is not included in your license, but would be useful, please contact us at support@radblue.com.

- The RST's Remote Control interface now includes support for all of the SmartEGM device events (those seen on the Device Events tab on the SmartEGM layout of the RST).

- You can now change the SmartEGM configuration file that is currently being used by the tool via a new method that was added to the Remote Control interface (of course this only works if the SmartEGM is not running). Using this new method, you can easily reset the EGM back to any known state (not necessarily all 0's.)

Release Date: 28 MAY 2014

Release Summary

In this release, we added quite a few usability improvements to the transcript control – color-coding the XML documents, letting you save changes that you make to the transcript layout and improving quick filter results so the display now includes the selected record when you clear the filter.

More specifically to the RST and the SmartEGM engine (the EGM simulation engine common to the RST and RLT), we added support for the master reset extension (1e), added retry logic for handpay requests, and added the ability to…. If you have licensed the optional Remote Control module, you'll be happy to discover that we've spent considerable time reworking the interface so it now functions as an API for any action available through the User Interface (or via a Tiger script)

We hope you enjoy this new release. If you have any ideas for improvement that we may have missed, just let us know at support@radblue.com.

The following sections provide details on each of the significant modifications and improvements.

Installer Modifications

- **The RST installer is now for a single vendor per installed instance** – This is only an issue if you are running multiple custom versions of the tool (very few are). You can now run a GSA version of the RST plus a one company's custom extensions per installed instance. If you run into issues, we can give you two separate licenses - one for each custom extension, and then you can install two separate instances of the tool. Let Russ (russ@radblue.com) know if you think you need something like this.

- **More Complete Error Reporting** – If the MAC address in the license file from RadBlue is not found in the PC, the resulting error message now displays a list of the MAC addresses that are seen by the installer for speedy solutions to installation problems.

- **All tools have been updated to use Java 1.7.0_51** – in an effort to stay current, all tools are being updated to a new version of Java. Java 8 is almost ready for prime-time, so we'll probably move to that new major version in the next release of our products.

Transport and Security Modifications

- **Proper handling of SCEP certificate chains** – The RadBlue SCEP client was not properly handling a certificate chain from an SCEP server. The tool was properly storing the first certificate, but then overwrote that certificate with the second certificate in the chain. Resolution - we now verify that the expected distinguished name matches the certificate before we store the certificate. (This is a common security routine used in all products).

- **Import/export self-signed certificates** – When the self-signed certificate option in the tool is used to create a signed certificate, that certificate has to be shared with other end-points so they trust the connection. New export and import options have been added to the Key Store tab in all tools so you can easily export a certificate from, and import a certificate into the tool's keystore file.
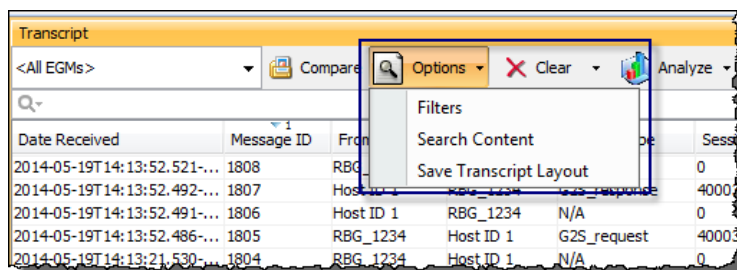
User Interface Modifications

- **Game play device events have been added to the RST** - A new gamePlay device event panel was added to the RST so you can cause a General gamePlay Tilt (causing events GPE202 (Device Tilt) and GPE001 (Device Disabled by EGM)) and you can Clear All Faults (causing events GPE099 (Device Tilts Cleared) and GPE002 (Device Enabled by EGM)). This new functionality was added to the Device Events Tab (accessed as follows):

Transcript Modifications

- **Minor reorganization of Transcript Controls** – Since UI changes are usually the most torturous, we'll start the Transcript section with this one. In this new release of the tools, the Search and Filter options have now been grouped with the new Save Transcript Layout option in a new group call **Options** (so clever, right?):
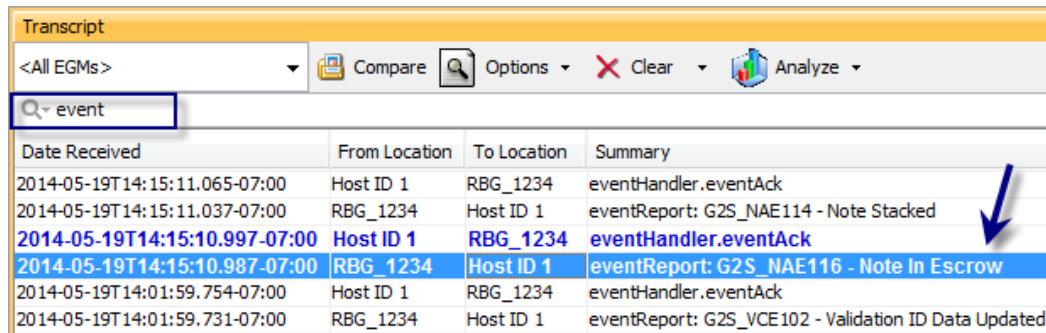

- **Column order, size, and selected columns are now persisted** – Make any changes to the layout of one instance of the transcript, then select **Options** > **Save Transcript Layout** to save your changes as the default layout. This new format is then used in all Transcript instances *in that tool as soon as* you restart the tool.

- **View XML is now color-coded!** We found a cool control that allows us to color code the XML document seen in the View XML object for Transcript Messages, using a unique color for elements, attributes and text strings. This makes the XML MUCH easier to read:

- **Improved reporting when real-time updates is disabled** – When the real-time update of the transcript is disabled, the **Realtime Update** label now goes to Red and Bold, and a message is displayed on the transcript table to make it obvious that the table is no longer being updated in real time.

- **Improved Quick Filter results display** – If you use the Quick Filter control in the transcript to search for a particular type of record and then select a specific message in the transcript table, that record will now always be in the active window when you clear the filter (so you can easily see the G2S messages around the selected message).

- **No more "Not Set" tool tips** – In the Transcript command view, there were lots of instances of tool tips that displayed as "Not Set" (because they weren't defined). These undefined tool tips no longer display in the tools.

- **More readable denominations** – In the command view for the meterInfo command, denominations are now shown in decimal form, rather than in millicents for currency meters and game denomination meters.

Tiger Scripting Modifications

- **The `core:exit` verb was not working properly at times** – The Tiger verb `<core:exit />` was not working when it was used within a `<catch />` block inside of a `<repeat />` loop. The verb can now be used anywhere in the script to end the script if a specified condition is encountered.

SmartEGM Modifications

- **gamePlay Primary Game End event was sent at the wrong time** – The SmartEGM was sending the G2S_GPE105 event (Primary Game End) at the wrong time - The reporting of the primary game end was occurring after the processing of the secondary game was completed, but has now been moved so the event is now generated before the check for a secondary game occurs.

- **Improved gamePlay device behavior** – If an EGM's progressive device is disabled by the G2S Host, the EGM now "egm disables" the affected game play devices, instead of incorrectly "host disabling" them.

- **Missing Meter definitions -** Meter definitions were not being included with currency and game denomination meters when requested by the host.

- **Forced event subscription Improvement #1** – The SmartEGM did not handle a forced event subscription that contained event code "G2S_all ", but now it does.

- **Forced event subscription Improvement #2** – The SmartEGM was not allowing the host to clear the forced event subscription table. Now, if the host sends an empty forced event subscription table in a `setOptionChange` command, the SmartEGM will clear the table of forced events.

- **The GTECH Master Reset (gtkMR) extension (1e) has been implemented** – This cabinet class extension is used by the host to direct the EGM to return to its factory settings. Upon receiving a `cabinet.masterReset` command, the SmartEGM will now generate all expected commands and events in the master reset sequence (though it doesn't actually reset the EGM;''s data model to the factory state).

- **Retry logic has been added to the handpay class** – The `handpayRequest` will be retried until the state of the handpay transaction changes to *keyedOff*. The `keyedOff` message will be retried until an acknowledgement is received. In both cases the `timeToLive` value in the handpay device's profile sets the retry frequency. If the `timeToLive` value is less than 5 seconds then 5 seconds is used.

Remote Control Modifications

- **RST Remote Control has been overhauled** – The RST Remote Control functionality has been completely updated so it now functions as an API for any button available through the User Interface (or via a Tiger script). The web-based scratchpad has also been updated to show the complete REST call and response, so the interface is self-documenting. ;o)
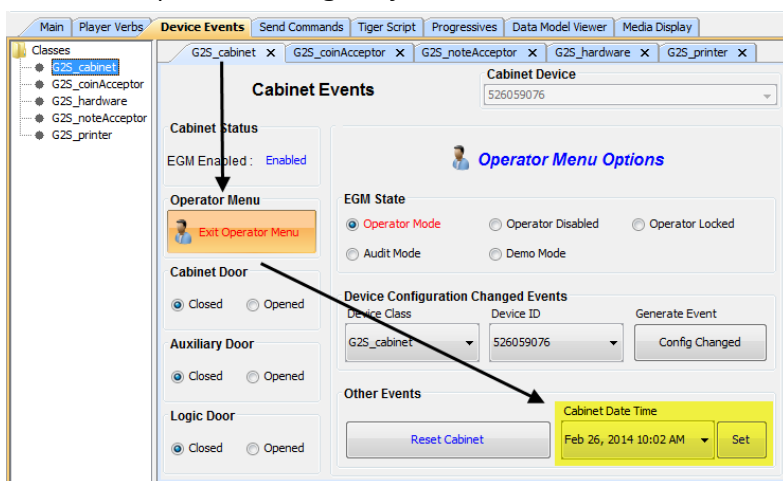
  The optional Remote Control module for RST allows your program to remotely drive the RST's SmartEGM, grab sections of the transcript for analysis, and do other cool things. If you are interested, let us know and we'll help you take it for a test drive.

Release Summary

In this release, we updated the Remote Control reference implementation, added the ability to reset the cabinet device date/time, removed obsolete options and corrected several issues.

New Features

- You can now reset the cabinet date and time through the **Operator Menu** (**SmartEGM > Device Events > G2S_cabinet > Enter Operator Menu**). A `G2S_CBE315` event is generated if the cabinet date/time is changed by more than 5 seconds.

Click **Set** to reset the cabinet date and time.

Engine Modifications

- Previoulsy, RST sent the `g2sAck` command using UTF-16 encoding. RST now sends the `g2sAck` command using UTF-8 encoding, as expected.
- The `eventHandler.eventLogList` command generated by RST is now correctly setting the affected meter device list in all cases.
- An issue in which the `mediaDisplay.hideMediaDisplay` command was returning an error when there was no active content running on the `mediaDisplay` device has been corrected.
- An issue in which RST was incorrectly requiring OWNER permission to process a `storage.getStorageInfo` request command has been corrected.

- Previously, RST was not using the `eventHandler.eventHandlerProfile` command's *timeToLive* attribute value as the retry frequency for `eventHandler.eventReport` commands. This issue has been corrected.

- Previously, when there were multiple logs in one class (gamePlay, download, etc.), RST used a single counter for the sequence number, resulting in skipped values in each log. RST now uses a separate sequence number for each log.

User Interface Modifications

- The following obsolete options have been removed from the **File** menu, on the menu bar:
  - New Desktop
  - Open Desktop
  - Save Desktop
  - Add Layout
  - Remove Layout

Remote Control Modifications

- An updated [RemoteControl.zip](#) reference implemantation file is now available. This file contains a full .NET project showing how to connect to the RST through a REST interface and issues sample calls.

- An issue in which Remote Control errors were not displayed in the Debug Console has been corrected.

Transcript Modifications

- An issue in which the data displayed incorrectly in the `eventHandler.eventHandlerLogList` details screen (accessed by double-clicking the message row in the Message Transcript) has been corrected.

- The **Load** option has been removed from the **Multicast Transcript** and **SOAP Transcript** objects because it is obsolete.