



# RadBlue System Tester Version 6

[Released: 09 DEC 2009]

Questions?

Log on to the RadBlue Forum: <http://radblue.mywowbb.com/>

## Release Notes

### Release Summary

In this release, added support for several unsupported events, addressed several issues and made usability improvements.

### Improvements

- The Transcript's **Compare** option has been enhanced to highlight differences in the two messages being compared.
- The Transcript has been modified for improved record processing performance.
- You can now send the `optionList` command by clicking **Send Option List** Device Events tab.
- The **New Desktop** and **Open Desktop** options have been removed from the Desktop and Layout menu.
- The list of game play devices in the Play Simple Central Game is now pulled from the central device's `centralGamePlay` table in the central's profile. The available denominations also comes from the central device's `centralGamePlay` table.
- The **smartegm-config-central.xml** file has been modified to include two central game play devices that correspond to game play devices on the EGM.
- The EGM now sends a `getValidationData` command to the host when it determines that there are no available validation IDs.
- The SmartEGM will only send an command ID not-in-order (G2S\_APX013) event when the command ID of the received message is less than or equal to the last processed command.
- RST now supports the following events:

**Note** RST lists each event as supported, but does not generate the event that contain in the event text description. We have added "(NS)" to the end of each event text description to differentiate which events are generated by RST and which are not

**Table 1** Coin Acceptor

Event	Description
G2S_CAE006	Device Configuration Changed by Operator (NS)

**Table 2** CommConfig

Event	Description	Event	Description
G2S_CCE001	commConfig Disabled by EGM (NS)	G2S_CCE003	commConfig Changed by Host (NS)
G2S_CCE002	commConfig Enabled by EGM (NS)	G2S_CCE004	commConfig Changed by Operator (NS)

**Table 3** Cabinet

Event	Description	Event	Description
G2S_CBE201	Comms Issue EGM Disabled (NS)	G2S_CBE313	Cabinet Tilt Cleared (NS)
G2S_CBE202	EGM Disabled - Operator Menu (NS)	G2S_CBE317	Cash-Out Button Pressed (NS)
G2S_CBE206	Operator Menu Activated (NS)	G2S_CBE318	Power Off - Logic Door Open (NS)
G2S_CBE207	Demo Mode Activated (NS)	G2S_CBE319	Power Off - Auxiliary Door Open (NS)
G2S_CBE208	Meters/Audit Mode Initiated (NS)	G2S_CBE320	Power Off - Cabinet Door Open (NS)
G2S_CBE209	EGM Locked - Operator Menu (NS)	G2S_CBE321	Operator Reset Cabinet (NS)
G2S_CBE210	Device Action Locked EGM (NS)	G2S_CBE322	Life-To-Date Meters Reset (NS)
G2S_CBE301	Service Lamp On (NS)	G2S_CBE323	NVM Cleared (NS)
G2S_CBE302	Service Lamp Off (NS)	G2S_CBE324	Backup Battery Low (NS)
G2S_CBE309	General Cabinet Tilt (NS)	G2S_CBE325	EGM Power Lost (NS)
G2S_CBE310	Video Display Error (NS)	G2S_CBE326	EGM Power Up (NS)
G2S_CBE311	NVM Failure (NS)	G2S_CBE327	Hard Meters Disconnected (NS)
G2S_CBE312	General Memory Failure (NS)	G2S_CBE316	Hard Meters Reconnected (NS)

**Table 4** Note Acceptor

Event	Description	Event	Description
G2S_NAE006	Device Configuration Changed by Operator (NS)	G2S_NAE115	Voucher Stacked (NS)

**Table 5** Event Handler

Event	Description	Event	Description
G2S_EHE001	Event Handler Disabled by EGM (NS)	G2S_EHE102	Event Handler Queue Overflow (NS)
G2S_EHE002	Event Handler Enabled by EGM (NS)	G2S_EHE103	Event Handler Queue Overflow Cleared (NS)
G2S_EHE006	Event Handler Configuration Changed by Operator (NS)		

**Table 6** GamePlay

Event	Description	Event	Description
G2S_GPE006	Device Configuration Changed at EGM (NS)	G2S_GPE104	Wager Changed (NS)
G2S_GPE099	Device Tilts Cleared (NS)	G2S_GPE107	Secondary Game Escrow (NS)
G2S_GPE102	Primary Game Failed (NS)	G2S_GPE108	G2S_GPE202 Device Tilt (NS)

**Table 7** OptionConfig

Event	Description	Event	Description
G2S_OCE001	optionConfig Disabled by EGM (NS)	G2S_OCE105	optionConfig Configuration Cancelled (NS)
G2S_OCE002	optionConfig Enabled by EGM (NS)	G2S_OCE107	optionConfig Configuration Aborted (NS)
G2S_OCE003	optionConfig Configuration Changed by Host (NS)	G2S_OCE108	optionConfig Error (NS)
G2S_OCE004	optionConfig Configuration Changed by Operator (NS)		

**Table 8** Hopper

Event	Description	Event	Description
G2S_HPE001	Device Disabled by EGM (NS)	G2S_HPE109	Extra Coins Paid (NS)
G2S_HPE002	Device Enabled by EGM (NS)	G2S_HPE110	Runaway Hopper (NS)
G2S_HPE006	Device Configuration Changed by Operator (NS)	G2S_HPE111	Dispenser Door Opened (NS)
G2S_HPE099	Device Faults Cleared (NS)	G2S_HPE112	Dispenser Door Closed (NS)
G2S_HPE101	Hopper Empty (NS)	G2S_HPE901	Device Disconnected (NS)
G2S_HPE102	Hopper Full (NS)	G2S_HPE902	Device Connected (NS)
G2S_HPE103	Hopper Jammed (NS)	G2S_HPE903	Device Firmware Fault (NS)
G2S_HPE104	Hopper Below High Water Mark (NS)	G2S_HPE904	Device Mechanical Fault (NS)
G2S_HPE105	Hopper Above High Water Mark (NS)	G2S_HPE905	Device Optical Fault (NS)
G2S_HPE106	Hopper Below Low Water Mark (NS)	G2S_HPE906	Device Component Fault (NS)
G2S_HPE107	Hopper Above Low Water Mark (NS)	G2S_HPE907	Device Non-Volatile Memory Fault (NS)
G2S_HPE108	Hopper Fault (NS)	G2S_HPE908	Illegal Activity Detected (NS)

**Table 9** Note Dispenser

Event	Description	Event	Description
G2S_NDE001	Device Disabled by EGM (NS)	G2S_NDE108	Dispenser Fault (NS)
G2S_NDE002	Device Enabled by EGM (NS)	G2S_NDE109	Dispenser Door Opened (NS)
G2S_NDE006	Device Configuration Changed by Operator (NS)	G2S_NDE110	Dispenser Door Closed (NS)
G2S_NDE099	Device Faults Cleared (NS)	G2S_NDE901	Device Disconnected (NS)
G2S_NDE101	Dispenser Empty (NS)	G2S_NDE902	Device Connected (NS)
G2S_NDE102	Dispenser Full (NS)	G2S_NDE903	Device Firmware Fault (NS)
G2S_NDE103	Dispenser Jammed (NS)	G2S_NDE904	Device Mechanical Fault (NS)
G2S_NDE104	Dispenser Below High Water Mark (NS)	G2S_NDE905	Device Optical Fault (NS)
G2S_NDE105	Dispenser Above High Water Mark (NS)	G2S_NDE906	Device Component Fault (NS)
G2S_NDE106	Dispenser Below Low Water Mark (NS)	G2S_NDE907	Device Non-Volatile Memory Fault (NS)
G2S_NDE107	Dispenser Above Low Water Mark (NS)	G2S_NDE908	Illegal Activity Detected (NS)

**Table 10** Download

Event	Descriptions	Event	Descriptions
G2S_DLE001	Device Disabled by EGM (NS)	G2S_DLE109	Package Failed Validation (NS)
G2S_DLE002	Device Enabled by EGM (NS)	G2S_DLE125	Package Upload Aborted (NS)
G2S_DLE006	Device Configuration Changed at EGM (NS)	G2S_DLE205	Script Time Ended (NS)
G2S_DLE007	EGM Locked by Device (NS)	G2S_DLE208	Script Waiting on Operator Initiation (NS)
G2S_DLE008	EGM Unlocked by Device (NS)	G2S_DLE305	Module Error (NS)
G2S_DLE104	Insufficient Storage for Package Download (NS)		

**Table 11** Printer

Event	Description	Event	Description
G2S_PTE006	Printer Configuration Changed by Operator (NS)	G2S_PTE103	Print Failed (NS)
G2S_PTE101	Print Transfer Failed (NS)		

**Table 12** Central

Event	Description	Event	Description
G2S_CLE001	Central Device Disabled by EGM (NS)	G2S_CLE006	Central Device Configuration Changed by Operator (NS)
G2S_CLE002	Central Device Enabled by EGM (NS)	G2S_CLE102	Central Device Outcome Failure (NS)

**Table 13** ID Reader

Event	Description	Event	Description
G2S_IDE001	Device Disabled by EGM (NS)	G2S_IDE902	Device Connected (NS)
G2S_IDE002	Device Enabled by EGM (NS)	G2S_IDE903	Device Firmware Failure (NS)
G2S_IDE006	Device Configuration Changed by Operator (NS)	G2S_IDE904	Device Mechanical Failure (NS)
G2S_IDE099	Device Faults Cleared (NS)	G2S_IDE905	Device Optical Failure (NS)
G2S_IDE104	ID Validated Offline (NS)	G2S_IDE906	Device Component Failure (NS)
G2S_IDE108	Invalid Validation Information Received (NS)	G2S_IDE907	Device Non-Volatile Memory Failure (NS)
G2S_IDE901	Device Disconnected (NS)		

**Table 14** Voucher

Event	Description	Event	Description
G2S_VCE001	Vouchers Disabled by EGM (NS)	G2S_VCE006	Voucher Configuration Changed by Operator (NS)
G2S_VCE002	Vouchers Enabled by EGM (NS)	G2S_VCE101	Validation ID Data Expired (NS)

**Table 15** WAT

Event	Description	Event	Description
G2S_WTE001	WAT Disabled by EGM (NS)	G2S_WTE006	WAT Configuration Changed by Operator (NS)
G2S_WTE002	WAT Enabled by EGM (NS)	G2S_WTE110	New Key Pair Failed - Invalid Key Value (NS)

## Corrections

- The Transcript has been updated to display the correct **To Location** and **From Location** for multicast messages.
- RGS has been modified to wait until the EGM is in an online state with the voucher host before sending the `getValidationData` command.
- Previously, the RST data model (as seen through the Data Model Viewer) did not always update after SmartEGM activity. The RST data model now updates correctly and consistently.
- If RST receives a `wat.error` command in response to an `initiateTransfer` request, it now commits the transfer.
- Previously, if the EGM was not idle and a `commConfig.enterCommConfigMode` command was received, the RST returned an APX011 (Time-to-Live Expired) error. Now, RST correctly returns a CCX003 (EGM Must Be Idle) error.
- The SmartEGM now stops communications immediately when the EGM receives an MSX003 (Communications Not Online) on a `commsOffLine` command.
- An issue with permissions for the Vista 7 operating system has been resolved. The user may need to perform an additional procedure. [\\_\\_\\_\\_\\_](#)



# RadBlue System Tester Version 5.0

[Released: 04 NOV 2009]

Questions?

Log on to the RadBlue Forum: <http://radblue.mywowbb.com/>

## Release Notes

### Release Summary

In this release, we made switching from a student license to a standard license easier, made usability changes to the Engine Options screen, and corrected some issues in the tool.

### Improvements

- When the user switches from a student license to a standard license, the following changes are made automatically:
  - ♦ The **My URL** option, under **Configure > Engine Options**, is set to a non-SSL location.
  - ♦ The SmartEGM configuration file is changed to the standard configuration file (**smartegm-config.xml**).
  - ♦ SSL features are reset to the standard (non-SSL) settings.
- The **Engine Options** screen on the Configure menu has been updated with consistent wording and a more user-friendly option order.
- A **Use original SmartEGM configuration file on startup** option has been added to **Configure > Desktop Options**. If this option is selected, RST ignores the updated configuration file (with the previous data model updates) and loads the smartegm-config.xml file each time the tool is started.

### Corrections

- If the G2S host returns an MSX003 error to a `commsClosing` command gets an MSX003, the SmartEGM immediately closes the communications channel instead of waiting 30 seconds for the `commsClosingAck` response command.
- If the `playSimpleGame Tiger` verb's *handpay-action* attribute is set to "handpay," the `playSimpleGame Tiger` verb does not send the `keyedOff` command. This requires the user to send a `keyOff Tiger` verb to clear the EGM.
- RST now returns a **G2S\_CME010** error if the security parameters in the MTPCoordination is invalid.





# RadBlue System Tester Version 4.0

[Released: 06 OCT 2009]

Questions?

Log on to the RadBlue Forum: <http://radblue.mywowbb.com/>

## Release Notes

### Release Summary

In this release, we added support for the financial class in Edge and Central. In addition, RST now supports OCSP.

### Improvements

- The S2S 1.3.1 financial class (manufacturer extension 1b1) has been added to RST Edge and Central. In addition, the following commands from the infoUpdate class have been added to support the financial class:

- ♦ setInfoUpdateSubList (client side)
- ♦ infoUpdateData (host side)

See [About the Financial Class](#).

- Online Certificate Status Protocol (OCSP) is now supported and can be configured through **Configure > Security Options > OCSP Options**. The OCSP options are applicable to both S2S and G2S.

A screenshot of the "OCSP Options" configuration window. It features a checkbox labeled "Enable OCSP" which is checked. Below this are three input fields: "gsaOO" with a value of 1, "gsaOR" with a value of 600, and "gsaOA" with a value of 630. Each field has a small up/down arrow icon. At the bottom is a text field labeled "Manual Access Location" which is currently empty.

- ♦ **Enable OCSP** - Select to enable Online Certificate Status Protocol (OCSP), which allows RST to determine the state of encryption certificates.
- ♦ **gsaOO** - Type or select the minimum period, in minutes, that RST will attempt to authenticate a certificate from an OCSP server. Zero (0) disables this setting.
- ♦ **gsaOR** - Type or select the maximum time, in minutes, that RST can use a certificate without re-authenticating it.
- ♦ **gsaOA** - Type or select the maximum time, in minutes, that RST can use a good certificate when OCSP servers are offline. Note that the gsaOA value should be greater period than the gsaOR value; The difference between gsaOR and gsaOA is the "accept offline" period.
- ♦ **Manual Access Location** - Type the URL location of the OCSP responder.

## Corrections

- (G2S) If the ID reader device is not specified, *idNumber* is set to "", *playerId* is set to "" and *idReaderType* is set to "G2S\_none" in the *issueVoucher* command.
- (G2S) Previously, when two event subscriptions were set with different values for the same event code, the values were not merged. The last value would overwrite the previous value. This was an issue because, according to the G2S protocol, subscriptions are additive. This issue has been corrected.

## About the Financial Class

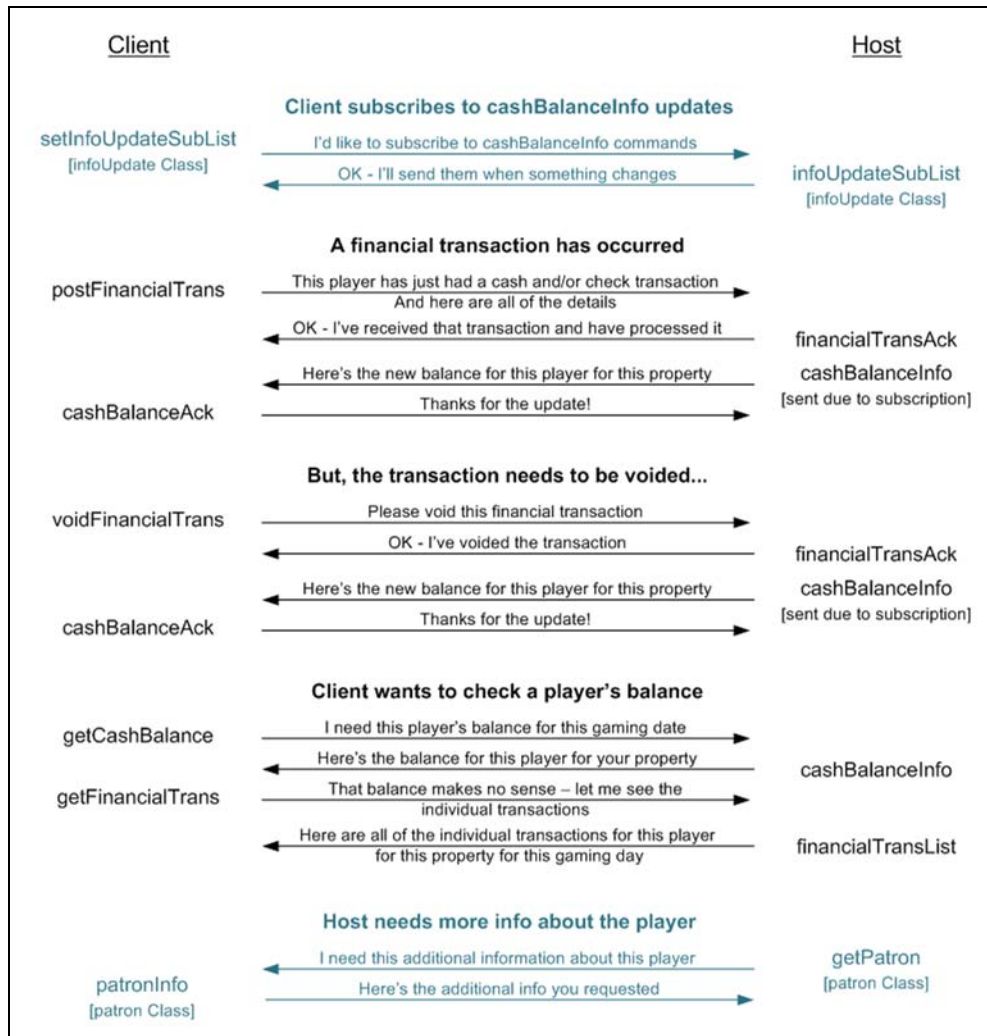
The financial class was created to address regulatory requirements for cash and check transaction tracking. This class is intended to be implemented with a central financial tracking system (acting as the host) with the following responsibilities:

- Receive information on cash transactions from multiple client systems that tallies cash received from and disbursed to the player. Each of these financial transactions indicates the source of the transaction. Multiple currency types can be included in a single transaction.
- Provide the current cash balance for each player identifier by means of subscription or user query.
- Receive information on check transactions associated with each player ID in order to produce a report summary.
- Accommodate multiple properties simultaneously.
- Aggregate daily cash transactions as they pertain to each client system.

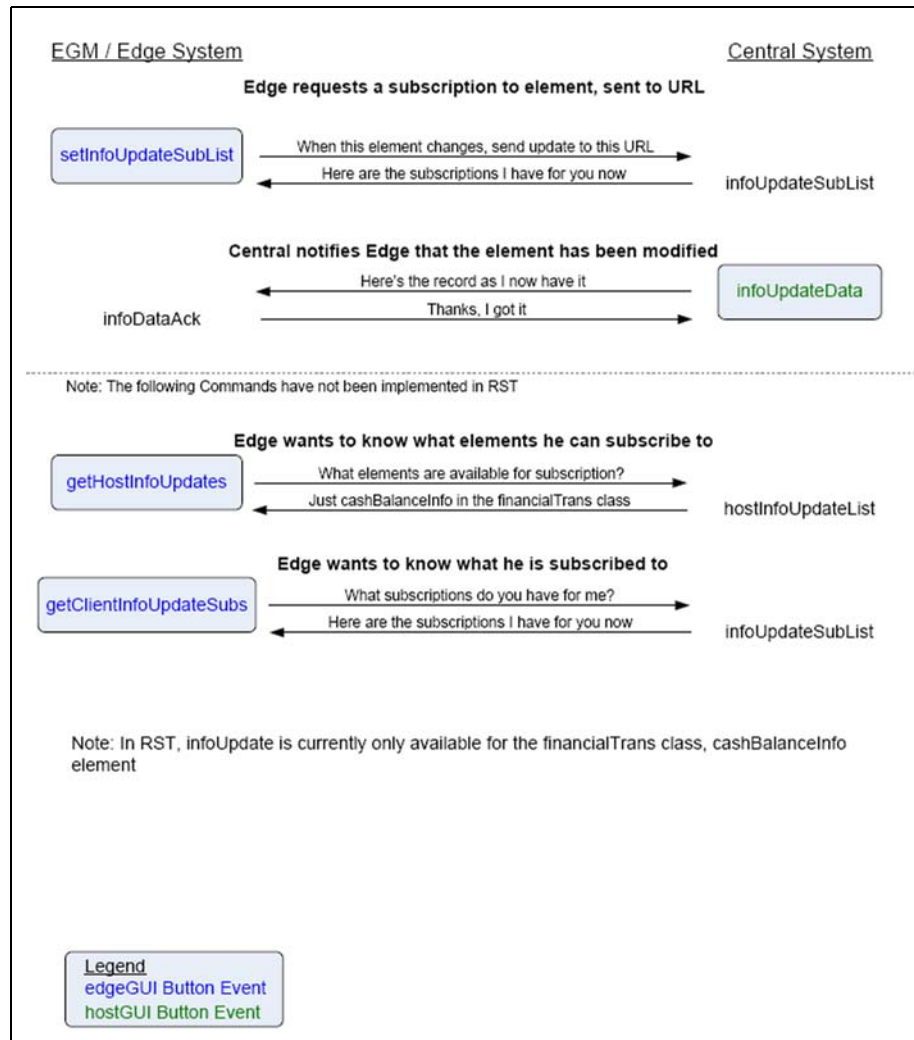
The financial transaction class provides client systems with the following commands:

- *postFinancialTransaction* - Sends details of a cash or check transaction to the host, including the gaming date to which the transaction is to be attributed.
- *voidFinancialTransaction* - Cancels a transaction sent in error (each transaction has a unique identifier created by the client creating the transaction).
- *getCashBalance* - Requests the cash balance for a particular player for the specified property and gaming date.
- *getFinancialTransactions* - Requests all individual financial transactions for this player for the specified property and gaming date.
- *infoUpdate* class - Through the *infoUpdate* class, the client can subscribe to the *cashBalanceInfo* command, to be notified whenever an update occurs for a specific player, at a specific property.

## Financial Class Message Flow



## InfoUpdate Class Message Flow





# RadBlue System Tester Version 3.0

[Released: 01 SEP 2009]

Questions?

Log on to the RadBlue Forum: <http://radblue.mywowbb.com/>

## Release Notes

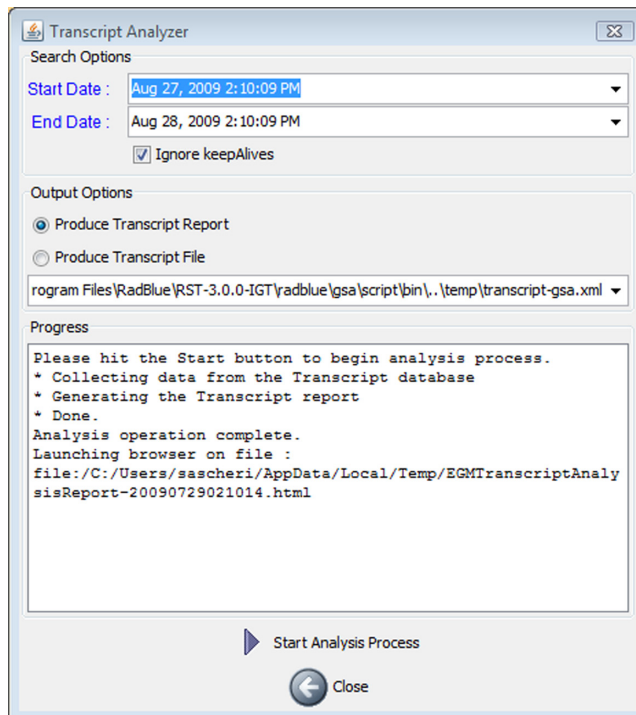
### Release Summary

In this release, several improvements and corrections, including a change that speeds up processing times for eventReport commands.

Note that we have changed the way RST is versioned. The first digit of the version number will increment each month (for example, this month is 3.0 and next month will be 4.0).

### Improvements

- A **Help** option has been added to the RST menu bar.
- You can now use license files with or without the MAC address in the file name.
- The voucher log *validationId* in the eventReport log is now masked with asterisks except for the last four digits.
- The Progress section on the Transcript Analysis report's Transcript Analyzer screen has been expanded to include report generation information.



## Corrections

- `keepAlive` commands are no longer sent as notifications.
- The **From Location** and **To Location** columns now display the actual source and destination for both request and response messages in the Transcript Control.



# RadBlue System Tester

## Version 1.15

[Released: 03 AUG 2009]

Questions?

Log on to the RadBlue Forum: <http://radblue.mywowbb.com/>

# Release Notes

## Release Summary

In this release, several improvements and corrections, including a change that speeds up processing times for eventReport commands.

## Improvements

- All attributes in the `setIdValidation` command have been added to the **id-database.xml** file.
- In order to speed up the RST processing time for eventReport commands, the eventReport engine was rewritten to improve performance. A by-product of this is that optional attributes are not included if the value of that attribute is equal to the default value defined in the schema.
- An **Unsupported Events** element has been added to the **smartegm-config.xml** file to specify the event codes that are not supported by the SmartEGM. See [Using the Unsupported Events Element](#).

## Corrections

- The Transcript has been modified to compare the selected two messages regardless of the current sort view.
- When loading a SmartEGM configuration file containing devices with an owner host ID of zero (0) (device is owned by the EGM), an **Unable to load the config file: Unable to load SmartEGM Configuration** error was reported. The SmartEGM now correctly loads **smartegm-config.xml** files where the owner host ID is zero.
- RST now handles the handpay sequence for the **Play Simple Game** verb correctly. The sequence is:
  - ♦ `send: handpay.handpayRequest`
  - ♦ `wait for: handpay.handpayAck`
  - ♦ `send: handpay.remoteKeyOff`
  - ♦ `wait for: handpay.remoteKeyOffAck`

## Using the Unsupported Events Element

The Unsupported Events element lets you specify which events are *not* supported by the SmartEGM for the host subscription. By default, the SmartEGM supports all events. The Unsupported Event element should be included in each eventHandler device that you want to affect.

Note that forced event subscriptions supersede unsupported events.

Name	Type	Use	Restrictions	Description
unsupported-events	element	optional	minOcc: 0 maxOcc: 1	Unsupported events element. This element can contain multiple unsupported-event elements.  Each eventHandler device may contain only <i>one</i> unsupported-events element.
unsupported-event	element	required	minOcc: 1 maxOcc: unbound	Single unsupported event element. This element can contain only one <i>event-code</i> attribute. However, depending the attribute value may specify multiple events, depending on how it is expressed.
event-code	attribute	required	xs:string	Event type or specific event to be excluded. Regular expressions are permissible.

### Unsupported Events Element Example

In this example, all gamePlay events are unsupported.

```
<edm:event-handler edm:device-id="1" edm:device-active="true" edm:configuration-id="0"
edm:host-enabled="true" edm:egm-enabled="true" edm:egm-locked="false"
edm:host-locked="false" edm:chatty="true" edm:owner-id="1" edm:config-id="1"
edm:vendor-id="Unknown" edm:product-id="Undefined" edm:release-number="Undefined"
edm:vendor-name="Undefined" edm:product-name="Undefined" edm:serial-number="Undefined">
  <edm:unsupported-events>
    <edm:unsupported-event edm:event-code="G2S_GPE" />
  </edm:unsupported-events>
</edm:event-handler>
```





# RadBlue System Tester Version 1.14

[Released: 29 JUN 2009]

Questions?

Log on to the RadBlue Forum: <http://radblue.mywowbb.com/>

## Release Notes

### Release Summary

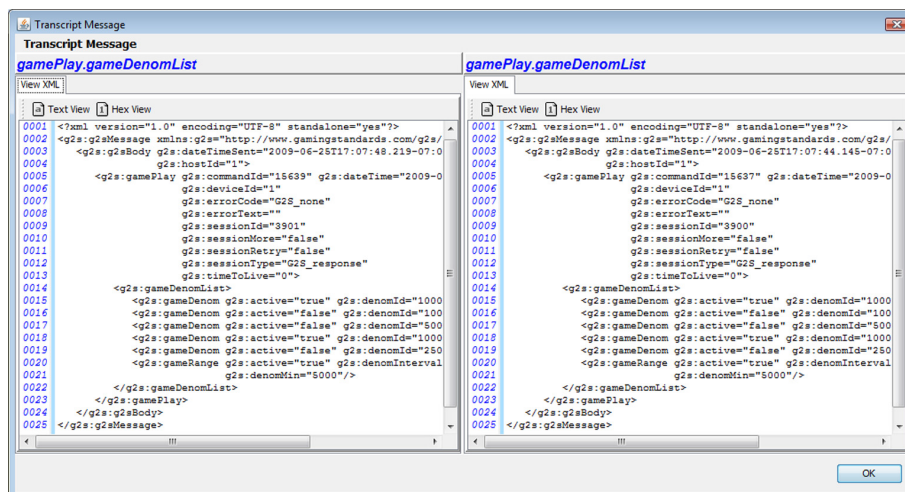
In this release, we added a mechanism to ensure stronger encryption ciphers are used and added the ability to compare the XML view of two commands in the transcript. We've also added support for several multicast events.

### Improvements

- RST can now force stronger encryption ciphers first in its cipher list. However, host systems must also support this methodology for that feature to work.

To use stronger encryption ciphers, you must take the following steps:

- i. Go to: <http://java.sun.com/javase/downloads/index.jsp>
  - ii. Scroll down to the **Additional Resources > Other Downloads**.
  - iii. Download **Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 6**.
  - iv. Follow the installation instructions in the README document, located in the download's .zip file.
- You can now compare the details of two commands in the transcript.



To use the compare feature:

- i. Hold down **CTRL** and click the commands you want to view.
- ii. Click **Compare**.

At this time, the Compare feature should not be used while transcript content is filtered.

- Event report command entries in the transcript now include the description associated with the event (for example: **eventReport:G2S\_EHE101 Event Subscription Changed**).
- The following multicast events are now supported in RST:
  - ♦ CME110 - Join Multicast Group
  - ♦ CME111 - Leave Multicast Group
  - ♦ CME112 - Multicast Message Error
  - ♦ CME113 - Security Parameters Updated

## Corrections

- The command summary in the transcript is now calculated correctly when the XML does not include the namespace prefix.
- The *validationIdList* attribute in the `VoucherStatus` command is now updated when a new list of validation numbers is received.



# RadBlue System Tester Version 1.13

[Released: 01 JUN 2009]

Questions?

Log on to the RadBlue Forum: <http://radblue.mywowbb.com/>

## Release Notes

### Release Summary

For release 1.13, we added support for the S2S handpay class as well as the ability to send a single `configuration.getInformation` message for all supported configuration elements. We also updated our S2S support to use the 1.2.6 and 1.3.1 schemas. In addition, all game play player verbs in the SmartEGM now support the in-game delay feature.

### Improvements

- The application splash screen has been modified to display startup messaging information.
- (G2S) The lockouts in all classes are now working properly. SmartEGM functionality is disabled.
- (G2S) In-game delays have been added to the player verbs **Play Progressive Game** and **Play Central Game** screens.

Prog ID	Prog Value	Game Play Device Id	# of Credits	Denom ID	Win Level Index	Prog Value ...
10	0.00	1	3	100000	1	UNDEFINED
10	0.00	1	3	100000	12	UNDEFINED

The delay is executed just before the generation of event G2S\_GPE105 (Primary Game End). A value of zero (0) denotes no delay.

This delay can be used to simulate delays in real-world EGMs. The delay makes the game cycle more realistic.

- (All Desktops) The Transcript now sorts records from newest to oldest, so the newest records appear at the top of the screen.
- (S2S) The RST-S2S tool has been updated to support S2S 1.2.6 and S2S 1.3.1.
- (S2S) A **Clear Browser History** button has been added to RST-S2S (Edge and Central). This button removes all RST-created cookies from the cache on your PC, and is used to clear all previously stored data for that instance of the tool.

- (S2S-Edge) RST S2S - Edge versions 1.2.6 and 1.3.1 now support sending a single `configuration.getInformation` command for all supported elements. Simply complete the required field(s) with the appropriate information. Fields left blank are not included in the request command.

**Send configuration.getConfiguration - All**

Fields marked with the asterisk control that section. If the field is blank then the corresponding S2S element will not be transmitted.

Employee ID \* : 11111111

Job Code ID \* : MKT05

Account ID \* : 12345678

Stop ID \* : 2

Chip Set ID \* : 5000

Game ID \* : RBD\_0025

Location ID \* : reno01

Pit ID \* : blkjk

Calculation ID \* : 1212

Comp Item ID \* : rart

Send Command Cancel

The host system should return a `configuration.configurationInfo` response command with the requested information.

- (S2S-Edge) RST S2S Edge now supports the following handpay commands:
  - ♦ `handpay.getEmployeeInfo`

**Send handpay.getEmployeeInfo**

☒ Magnetic Card  
☐ Key Pad  
ID Type : ☐ Finger Scan  
☐ Retina Scan  
☐ Prox Card

ID Number : 11111111

Send Command Cancel

- ◆ `handpay.getEmployeePin`

**Send `handpay.getEmployeePin`**

☒ Magnetic Card  
☐ Key Pad  
 ID Type : ☐ Finger Scan  
☐ Retina Scan  
☐ Prox Card

ID Number :

Pin Number:

- ◆ `handpay.disburseRequest`

**Send `handpay.disburseRequest`**

☐ EGM  
☒ KIOSK  
☐ CASHIER  
 Request Type :

Request ID :

Machine ID :

Machine Location :

☒ Large Win  
☐ Cancel Credit  
 Handpay Type :

Transaction ID :

Handpay Date/Time :

Handpay Amount :

Currency ID :

Game Win :

Bonus Win :

Progressive Win :

Machine Paid :

Last Wager :

Max Bet Value :

Winning Combination :

Disburse ID :

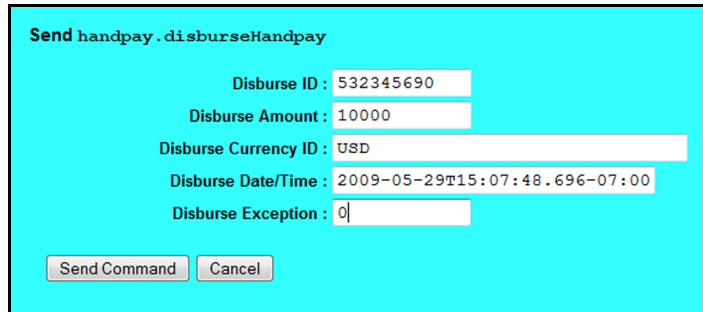
Disburse Amount :

Disburse Currency ID :

Job Code ID 1 :

Employee ID 1 :

- ♦ `handpay.disburseHandpay`



**Send handpay.disburseHandpay**

Disburse ID : 532345690

Disburse Amount : 10000

Disburse Currency ID : USD

Disburse Date/Time : 2009-05-29T15:07:48.696-07:00

Disburse Exception : 0

## Corrections

- (G2S) Previously, the note dispenser status reported "false" for *egmEnabled* and *hostEnabled*, regardless of its actual state. The note dispenser status is now reported correctly.
- (G2S) The *changeStatus* attribute in the `commConfig.commChangeStatus` command now updates correctly when a pending change is cancelled.
- (G2S) Progressive device meters are now saved in the **smartegm-config-updated.xml** file.
- (G2S) An issue has been corrected in which progressive device meters were not being included in the GPE112 (Game End) event.
- (G2S) RST now returns the `commChangeLogList` when requested from the G2S host.



# RadBlue System Tester Version 1.12

[Released: 29 APR 2009]

Questions?

Log on to the RadBlue Forum: <http://radblue.mywowbb.com/>

## Release Notes

### Release Summary

For release 1.12, we have added shortcuts for each RST desktop and removed the Initial Desktop Setup screen. In addition, the SmartEGM configuration file is now automatically loaded for you. Finally, the **smartegm-config.xml** file has been updated to support denomination ranges.

### Improvements

- The Linux version of the RST installer now uses the file name:  
**radblue-system-tester-license.xml**  
See [Bulletin 02: Installing RadBlue Tools on Linux](#).
- (S2S) RST Edge and RST Central now support the `playerRating.activeRatingReq` command in S2S version 1.3.1.
- The **Initial Desktop Setup** screen has been removed, so you will no longer be asked to select a desktop before RST launches. Instead, when RST is installed, a shortcut is created for each default desktop (SmartEGM, Edge and Central). These icons are located under **Start > All Programs > RadBlue System Tester**. To create desktop icons, simply drag and drop the shortcut(s) you want onto your computer desktop.

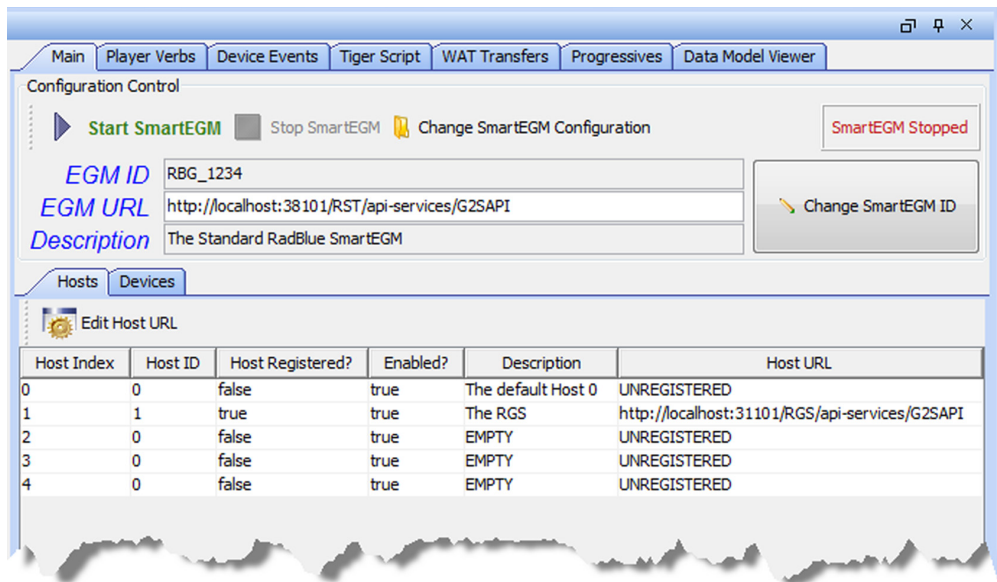
Note that if you are running Edge and Central on the same computer, each desktop is configured for running Edge and Central on the same computer. However, you must still install two instances of RST. See [Bulletin 04 - Installing S2S Edge and S2S Central on the Same PC](#) for more information.

- Once you have launched RST, you can still configure the startup desktop through the **Configure > Desktop Options** screen.
- The `getTransportOptions` WSDL call, which is executed automatically at start-up, is now *optional*. Use **Configure > Engine > Enable Get Transport Options** to toggle this behavior.

- In this release, we added denomination range support in the SmartEGM. The gamePlay devices 1 and 3 contain denomination ranges, expressed as follows in the **smartegm-config.xml** file:

```
<edm:game-denoms>
  <edm:game-denom edm:denom-id="1000" edm:active="true" />
  <edm:game-denom edm:denom-id="5000" edm:active="true" />
  <edm:game-denom edm:denom-id="10000" edm:active="true" />
  <edm:game-denom edm:denom-id="15000" edm:active="true" />
  <edm:game-denom edm:denom-id="20000" edm:active="true" />
  <edm:game-denom edm:denom-id="25000" edm:active="true" />
  <edm:game-denom edm:denom-id="100000" edm:active="true" />
  <edm:game-denom edm:denom-id="500000" edm:active="true" />
  <edm:game-denom edm:denom-id="1000000" edm:active="true" />
  <edm:game-denom edm:denom-id="2500000" edm:active="true" />
  <edm:game-range edm:denom-min="5000" edm:denom-max="25000"
edm:denom-interval="5000" edm:active="true" />
</edm:game-denoms>
```

- (G2S) The **smartegm-config.xml** file is now automatically loaded when you start the RST. After the initial startup, a checkbox allows you to select whether you load the original configuration or the latest (-update) configuration.



When RST SmartEGM launches, simply click **Start SmartEGM**.

If you want to use a different SmartEGM configuration file, click **Change SmartEGM Configuration** and select a new file.

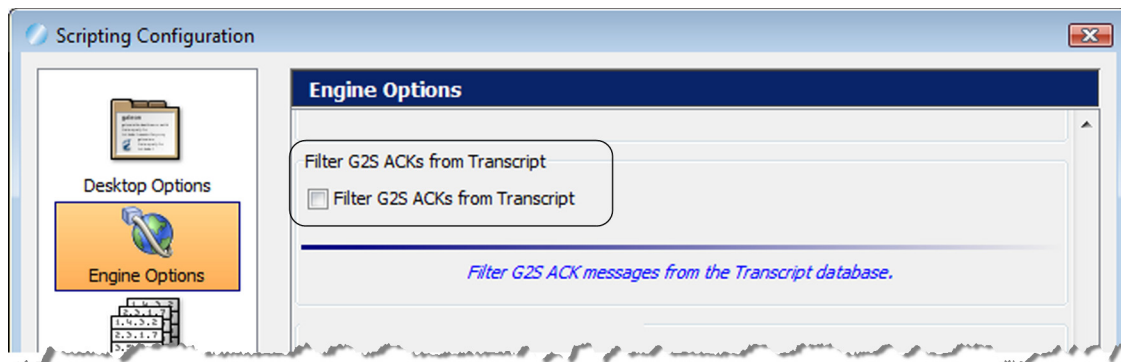


## Corrections

- (G2S) Previously, bonus awards were not starting properly after a restart of the EGM. A check has been added to ensure that bonusing is started, if it has not already done so.
- *On Linux versions of the tool only*, an error was caused when the Transcript object was dragged and dropped onto an RST layout. As a result, the Transcript object did not display. This issue has been corrected.
- (G2S) The `uploadPackage` command now goes through the states as outlined in Figure 10.2 of the G2S Message Protocol.
- (S2S) An issue with the S2S user interface not displaying has been corrected.
- (G2S) RST now handles the `progressiveError` from the host by logging the message and not sending the `APX008` (Command Not Supported) error back. Previously, the `APX008` was being sent, in violation of the G2S protocol.

## Filtering G2S ACK Messages from the Transcript

The **Configure > Filter G2S ACKs from Transcript** option lets you remove `g2sAck` messages from the Transcript. *If you change this option, you must restart RST for the change to take effect.*





# RadBlue System Tester Version 1.11

[Released: 30 MAR 2009]

Questions?

Log on to the RadBlue Forum: <http://radblue.mywowbb.com/>

## Release Notes

### Release Summary

In this release, we added support for S2S 1.3.1 as well as addressing some minor issues.

### New Features

- RST now supports versions 1.2.6 and 1.3.1 of the S2S protocol. Both versions now support the following classes.

**Note** In all cases, the other entity (either Edge or Central) will respond to a request in these classes.

**Table 1** S2S classes supported by RST-S2S for versions 1.2.6 and 1.3.1.

Class	You can send commands from...	Purpose
communications	Edge Central	Edge server initiates communications with the central server, and then sends a variety of commands to report status changes in the communications class.
comp	Edge Central	The Edge server automatically sends a <code>getConfiguration</code> command to the Central server upon startup.
configuration	Edge Central	The Edge server can request configuration information from the Central server for several classes.
egmRegistration (S2S 1.2.6 only)	Edge	The Edge server replicates Electronic Gaming Machines (EGM) data to the Central server. <b>Note:</b> This class has been deprecated by GSA.
fillCredit	Edge Central	Send a <code>fillCredit</code> issuance command from the Edge server, or send a <code>fillCreditInfo</code> message from the Central server.
marker	Edge Central	Issue a marker or voucher from the Edge server, or send an update from the Central server.
openClose	Edge Central	Open, post and close a table rating from the Edge server.

**Table 1** S2S classes supported by RST-S2S for versions 1.2.6 and 1.3.1.

Class	You can send commands from...	Purpose
patron	Edge Central	Player management messages sent between a server and a patron data warehouse system.
player	Edge	Communication of carded player gaming activity to central player tracking system.
playerRating	Edge Central	Start a rating, update a rating, close a rating and void a rating from the Edge server. Use the Central server to send the <code>ratingInfo</code> command.
registerClient	Edge	Edge server replicates data about any of its clients (including EGMs) to the central server.
voucher	Edge	Communication to central voucher system to add and redeem gaming vouchers.
wat	Edge	From the Edge server, you can request WAT funds, get an account balance and transfer WAT funds.

## Improvements

- (S2S) The required fields for the `configuration.configurationInfo` command are now enforced if an required section of the command is sent
- (G2S) All `optionGroup` elements of the same device identifier are now grouped under the same `deviceOption`. This eliminates redundant device classes with the same device ID.

## Corrections

- (G2S) An issue has been corrected with a `G2S_APX013 (Command Not In commandId Order [Command ID: x, expected x])` error generating when a `commsOnLine` is sent
- (S2S) An issue that involved a DWR error appearing unnecessarily when launching RST Edge and RST Central has been corrected.



# RadBlue System Tester

## Version 1.10

[Released: 02 MAR 2009]

Questions?

Log on to the RadBlue Forum: <http://radblue.mywowbb.com/>

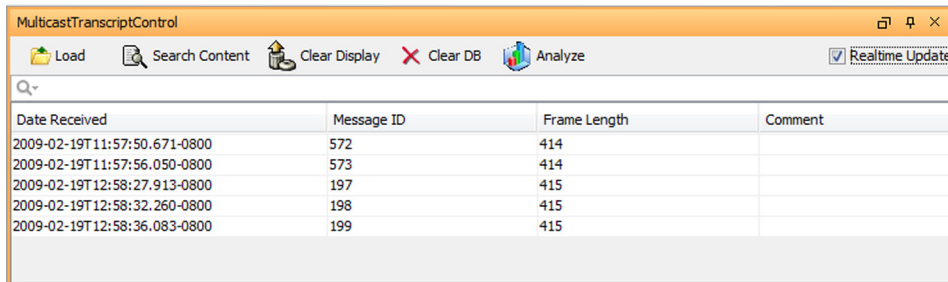
# Release Notes

## Release Summary

In release 1.10, we added WAT support, as well as additional patron support, in the S2S 1.2.6 scripts for RST Central and RST Edge. A new Multicast Transcript lets you view multicast messaging details.

## New Features

- A new **Change SmartEGM ID** button has been added to the SmartEGM Main interface. This option allows you to change the ID number of the EGM running on the SmartEGM. The Change SmartEGM ID button is enabled once a SmartEGM configuration file has been loaded, and the SmartEGM is stopped.
- A new Multicast Transcript has been added to RST. See [Using the Multicast Transcript](#) for more information.



Date Received	Message ID	Frame Length	Comment
2009-02-19T11:57:50.671-0800	572	414	
2009-02-19T11:57:56.050-0800	573	414	
2009-02-19T12:58:27.913-0800	197	415	
2009-02-19T12:58:32.260-0800	198	415	
2009-02-19T12:58:36.083-0800	199	415	

The Multicast Transcript object has been added to the All Commands, My Desktop and G2S Developer desktops.

- RST Edge now supports four new WAT requests for S2S 1.2.6:
  - ♦ `getWatAccounts`
  - ♦ `getWatBalance`
  - ♦ `requestTransfer`
  - ♦ `commitTransfer`
- RST Central now supports four new WAT responses for S2S 1.2.6:
  - ♦ `watAccountList`
  - ♦ `watBalance`
  - ♦ `authorizeTransfer`
  - ♦ `ackTransfer`

- RST Edge now supports four new requests for the patron class when using the S2S 1.2.6 script:
  - ♦ addPatron
  - ♦ updatePatron
  - ♦ deletePatron
  - ♦ queryPatron
- RST Central now provides a modifyPatronInfo response for the above four patron class commands in S2S 1.2.6 script.

## Improvements

- A new **smartegm-config-mult-events-per-msg.xml** file changes the eventHandler device to include up to five eventReport commands in each G2S message.
- The SmartEGM Config File Loader now displays any errors in the SmartEGM configuration file being loaded.
- Jetty (the web server used in RadBlue tools) has been updated from version 5 to version 6. RadBlue recommends that you use valid certificates when using the tools.
- Prior to version 1.10, RST did not invoke the getTransportOptions command in the G2S WSDL. RST now invokes the command prior to sending the commsOnLine command. If the **Enable GZIP** option (under **Configure > Engine Options**) is selected, RST reports that it supports GZIP when asks for its transport options by a host system.
- Under **Configure > Engine Options > Enable GZIP**, the “Enable GZIP compression on all outbound requests” informational text has been changed to “Enable GZIP compression on all requests.”
- In-game delays can now be specified through the user interface or by running a Tiger/XML script. The delay is actually executed twice: just before the generation of event G2S\_GPE105 (Primary Game End) and just before the generation of event G2S\_GPE110 (Secondary Game End). A value of zero (0) denotes no delay.

This delay can be used to simulate delays in real-world EGMs - The delay makes the game cycle more realistic.

- The Direction and Line Number columns have been removed from the Transcript object
  - A **From Location** column and a **To Location** column have been added to the Transcript object
- For G2S*, if the sender of the message is an EGM, the From Location is an EGM ID (for example, RBG\_1234) and the To Location is a host ID (for example, host ID 7).

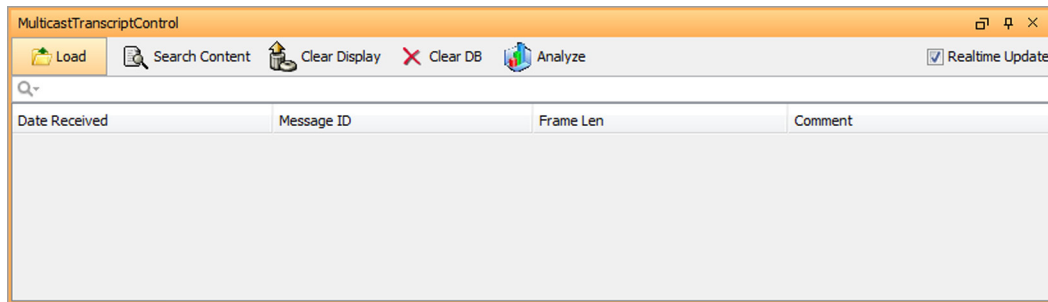
If the sender of the message is a host, the From Location is a host ID and the To Location is an EGM ID (or a multicast ID if the message was sent as a multicast message).

*For S2S*, the From Location is the sending system’s ID and the To Location is the receiving system’s ID.

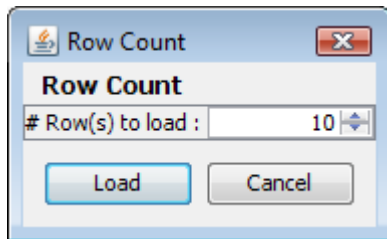
## Using the Multicast Transcript

The Multicast Transcript displays multicast messages that are sent between the host and EGM. Note that G2S commands contained in multicast messages can also be viewed through the normal Transcript object.

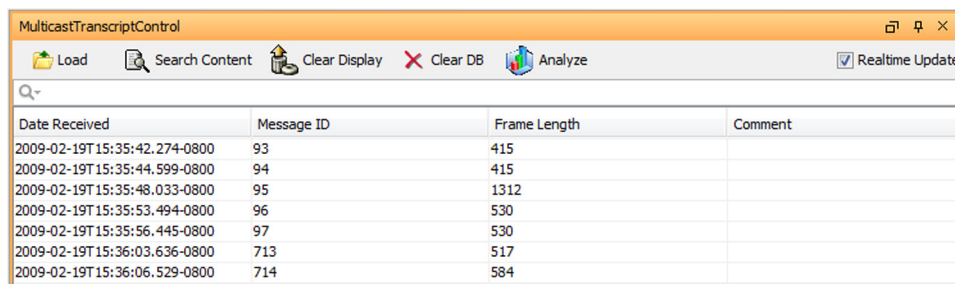
1. Select the Multicast Transcript object tab or drag-and-drop the Multicast Transcript object into the layout of your choice.



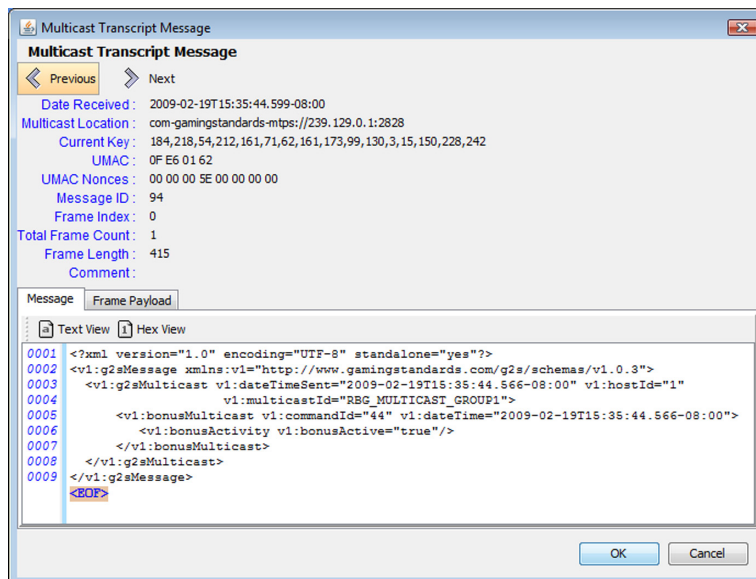
2. Click **Load** to load multicast message data.



3. Enter the number of messages (rows) you want to view by using the arrows or by typing the number into the combo box.
4. Click **Load**.
5. Select **Realtime Update** to automatically update the Multicast Transcript as messages are sent or received.

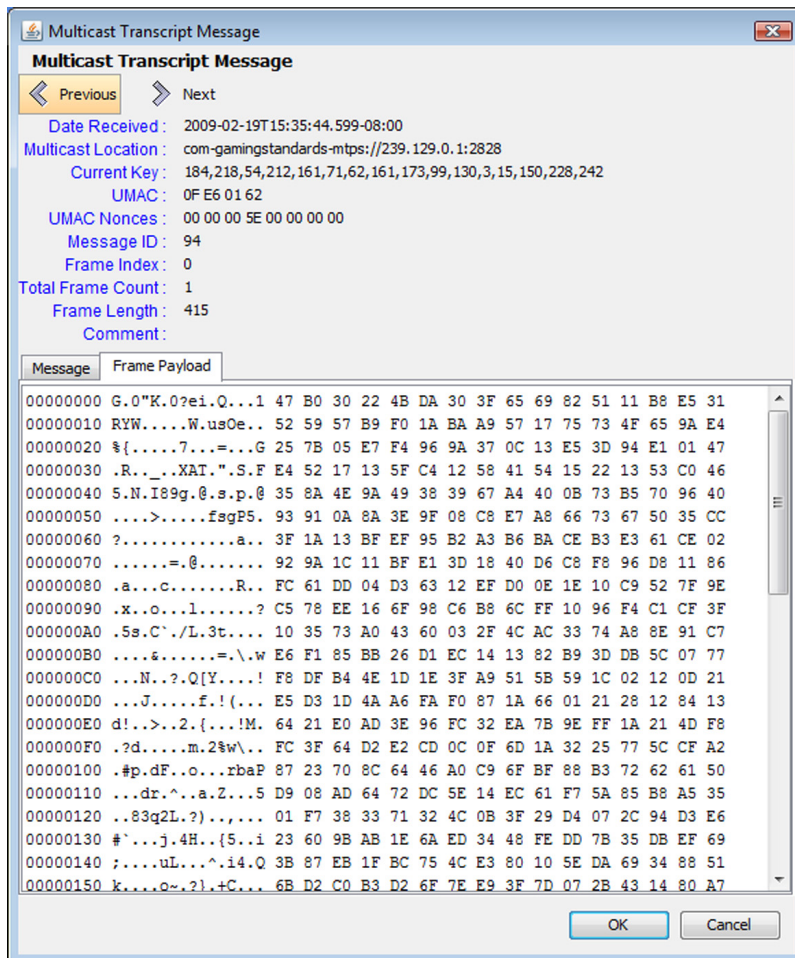


6. Click any column header to sort on that column.
  - **Date Received** - Date and time message is received by the tool.
  - **Message ID** - Identification number of the message.
  - **Frame Length** - Length of the frame contents.
  - **Comment** - Indicates an invalid UMAC (Message Authentication Code using Universal Hashing) or an invalid frame length if an error occurs. If the message is valid, this field is blank.
7. Click **Clear Display** to clear the Multicast Transcript display.
8. Click **Clear DB** to remove all saved data from the Multicast Transcript.
9. Double-click a message to view additional details.



- **Date Received** - Date and time message is received by the tool.
- **Multicast Location** - Multicast address or other transport-specific parameters.
- **Current Key** - Key used to authenticate message.
- **UMAC** - Message Authentication Code using Universal hashing.
- **UMAC Nonces** - The core security requirement of the UMAC is the notion of the *nonce*. MTP nonces are a combination of (Message ID + Frame Index + 0x0 + 0x0).
- **Message ID** - Identification number of the message.
- **Frame Index** - Index number of the frame.
- **Total Frame Count** - Total number of frames in message.
- **Frame Length** - Length of the frame contents.
- **Comments** - Indicates an invalid UMAC or an invalid frame length. If the message is valid, this field is blank.

10. On the Message tab, click **Text View** to view the message in either text format, or click **Hex View** to view the message in hexadecimal format.
11. Click **Frame Payload** to view the message payload. If the message is encrypted, the information displays in hexadecimal format.



If the message is not encrypted, the information displays in text format.

12. Click **OK** to close the Multicast Transcript Message screen.





# RadBlue System Tester Version 1.9

[Released: 02 FEB 2009]

Questions?

Log on to the RadBlue Forum: <http://radblue.mywowbb.com/>

## Release Notes

### Release Summary

For release 1.9, we've created a Student Edition of RST that contains all of the functionality of the full RST version, but can communicate *only* with RGS. The RST Student Edition lets you learn the G2S protocol without purchasing a full version of the tool.

**Note** You must have a 2009 license to use RST versions released in 2009. This includes RST 1.9.

### New Features

- A new RST Student Edition is now available. The RST Student Edition is a fully functional version of RST for use with RGS, the RadBlue G2S host simulator. This new edition of RST lets you get the experience you need to understand the G2S protocol. The RST Student Edition comes with the *RST and RGS Quick Start*, a self-paced training guide to get you up-and-running quickly.

### Improvements

- RST has been modified to support a `sourceRef` element when creating a new handpay. See [Generating a Handpay Source Reference](#) for more information.

**Note** The `progressive.setProgressiveWin` command with a `payMethod` of **payHandpay** and **payVoucher** is not supported.  
The `bonus.setBonusAward` command with a `payMethod` of **payHandpay** and **payVoucher** is not supported.

- A source reference is now included for the `commitTransfer` and `watLogList` WAT commands, for handpay-only initiated WAT transfers. See [Generating a Handpay Source Reference](#) for more information.
- The SmartEGM now creates a source reference for a handpay to a voucher. See [Generating a Voucher Source Reference](#) for more information.
- The following voucher timing attributes have been modified in the **smartegm-config.xml** file:
  - ♦ `G2S_valldListLife` - 10 minutes
  - ♦ `G2S_valldListRefresh` - 5 minutes
  - ♦ `G2S_voucherHoldTime` - 15 seconds
  - ♦ `G2S_minLevellds` - 15

- The License Manager has been improved with additional user information for invalid MAC addresses.
- The *disableText* and *lockText* attribute values now appear on the Player Display for bonus devices, voucher devices, WAT devices, and cabinet devices. For example:



## Corrections

- In accordance with the G2S protocol, the **smartegm-config.xml** has been modified to send `keepAlive` commands as requests.
- When you enable an `eventHandler` device, RST now correctly generates a `G2S_EHE003` event.
- When you disable an `eventHandler` device, RST now correctly generates a `G2S_EHE004` event.
- The *enabled* attribute in the `optionConfig.optionConfigModeStatus` command is now correctly set to **false** for the option config device in the **smartegm-config.xml** file.
- The *allowMulticast* attribute in the `commsProfile` command now defaults to **true**.
- When the host disables a progressive device, all `gamePlay` devices that contribute to that progressive are automatically disabled.
- Disabled devices are no longer available for selection in `Player Verb` objects (for example, `Play Simple Game` and `Play Progressive Game`).
- The `setPlayerMessage` command can now be sent through multicast.
- When the host locks the cabinet device, the *egmState* is now set correctly, and the *enableMoneyIn* and *enableMoneyOut* attributes are disabled.
- In a multi-command message, the Transcript summary now displays the command ID and session ID of the first command.
- Issues with running SmartEGM on Linux have been corrected.

## Generating a Handpay Source Reference

To see the added handpay source reference in RST:

1. From the RST SmartEGM Player Verbs tab, click **Insert Note** and insert a 1-dollar note into EGM.
2. Click **Play Simple Game**.
3. Type **1** in the **Primary Win (in Credits)** field.
4. Select **Win to handpay**.
5. Click the drop-down arrow for **How to pay handpay**, and select **Pay to Handpay**.
6. Click **Play Game**.
7. Select the Transcript tab, and review the `handpay.handpayRequest` command to see the **handpaySourceRef**. For example:

```
<g2s:handpaySourceRef g2s:cashableAmt="1000" g2s:deviceClass="G2S_gamePlay" g2s:deviceId="1"
                        g2s:logSequence="8"
                        g2s:nonCashAmt="1000"
                        g2s:promoAmt="1000"
                        g2s:transactionId="27"/>
```

## Generating a WAT Source Reference

To see the added WAT source reference in RST:

1. From the RST SmartEGM, select the **WAT Transfer** tab.
2. Click **Get Key Pair**, and then click **Get Key Pair**.
3. Click **Insert ID**, enter a valid player ID (**12345678**), and click **Insert ID**.
4. From the RGS (or your host system), send the `WAT.setWatCashOut` command with a `cashOutToWat` value of **true**. (In RGS, go to: **Send Command > G2S\_wat > WAT - Set WAT Cash Out**)
5. From the RST SmartEGM Player Verbs tab, click **Insert Note** and insert a 1-dollar note into EGM.
6. Click **Play Simple Game**.
7. Type **1** in the **Primary Win (in Credits)** field.
8. Select **Win to handpay**.
9. Click the drop-down arrow for **How to pay handpay**, and select **Pay to WAT**.
10. Click **Play Game**.

11. Select the Transcript tab, and review the `wat.commitTransfer` command to see the **watSourceRef**. For example:

```
<g2s:watSourceRef g2s:cashableAmt="1000" g2s:deviceClass="G2S_handpay" g2s:deviceId="1"
    g2s:logSequence="5"
    g2s:nonCashAmt="0"
    g2s:promoAmt="0"
    g2s:transactionId="21"/>
```

## Generating a Voucher Source Reference

To see this in RST, from the SmartEGM Player Verbs tab:

1. Click **Insert Note**, and insert a 1-dollar note into EGM.
2. Click **Cash Out to Handpay**.
3. Click **Key Off Handpay**.
4. In the **Key Off Type** field, click the drop-down arrow, and select **Pay to Voucher**.
5. Click **Key Off**.
6. Select the Transcript tab, and review the `voucher.issueVoucher` and `voucher.voucherLogList` commands to see the **voucherSourceRef**. For example:

```
<g2s:voucherSourceRef g2s:cashableAmt="100000" g2s:deviceClass="G2S_handpay"
g2s:deviceId="1"
    g2s:logSequence="2"
    g2s:nonCashAmt="0"
    g2s:promoAmt="0"
    g2s:transactionId="3"/>
```