Bulletin 03

Questions? Comments?
Visit us at www.radblue.com or check out our user forum at:
http://radblue.mywowbb.com/

Installable Packages

About Installable Packages

To use installable packages, two commands must be sent from either the RGS or host system:

1. `addPackage` - used to instruct an EGM to download a package
2. `setScript` - used to define the action that should be taken on the downloaded package (for example, install)

Once a package is installed, changes to the EGM's data model can be seen in the RST SmartEGM Data Model Viewer.

A `commsClosing` command is sent when the communications channel restarts. It is followed automatically with a `commsOnline` command.

Installable Package Format

At the highest level, a RadBlue installable package is a .zip file (WinZIP or Microsoft Zip format), containing the following:

1. An optional **readme.txt** file that describes the package. This file is not used in our tools.
2. A mandatory **package.xml** file that is used by RadBlue tools to provide information about the package (satisfying the G2S requests for information about the package). For more information, see *PACKAGE.XML File Format* on *page 2.*
3. One or more file folders, each of which can be considered a module within the package.

   Each file folder contains one or more XML files, with a valid XML document of one or more gamePlay devices in valid **smartegm-config.xml** file format (validated against the **smartegm-config.xsd**). The **package.xml** file points at the XML files through its *filename* attribute so they can be installed.

   Non-gamePlay devices can be present in the XML files, but are ignored (you can only install gamePlay devices).

   The gamePlay device IDs within the XML files must be non-zero and unique.

RST Tests on Installable Packages

The following tests are performed by RST to determine if the package is an installable package:

1. Is it a valid .zip file?
2. Is there a **package.xml** file in the root? (If so, it's a RadBlue package.)
3. Does the **package.xml** file list a package with **module-type= G2S_game**?
4. Modules are unpacked.
   a. Is the referenced file valid against our SmartEGM configuration schema?
   b. Are there gamePlay devices with unique, non-zero deviceId attributes.

If all of the above tests pass, the package can be installed.

PACKAGE.XML File Format

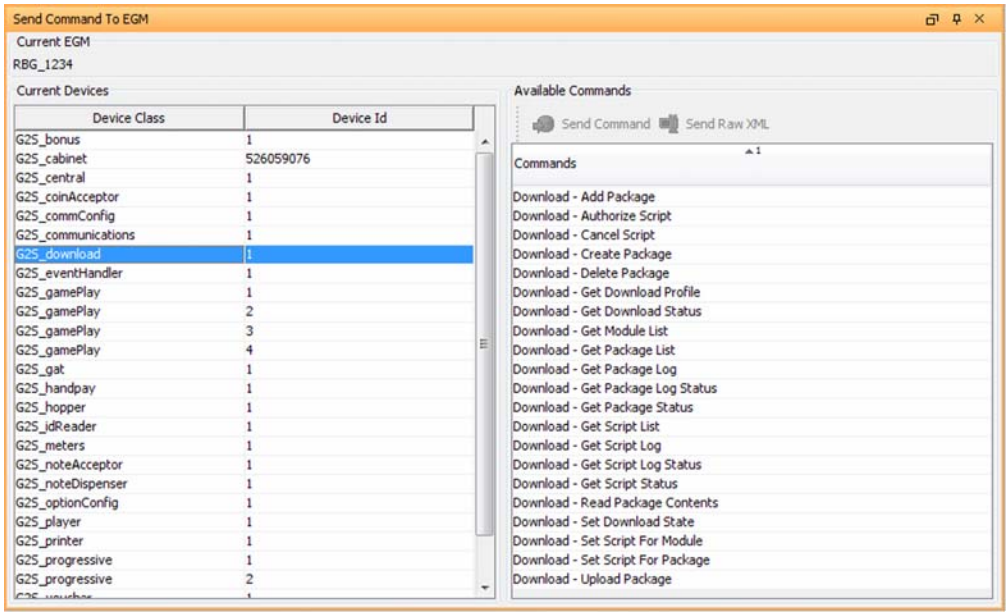
Name	Description
***Element*: package**	Contains data that is used to describe the package
Attribute: description	This attribute describes the package. It is used in the `G2S_packageContents` command.
Attribute: release-number	A release number for the package. This attribute is also used in the `packageContents` command.
Attribute: namespace	This attribute declares the namespace for the XML document.
***Element*: module**	This element contains data that describes one module within this package. It can be repeated for multiple modules in one package.
Attribute: module-id	This attribute is used in moduleInfo in the `packageContents` command.
Attribute: description	This attribute is used in moduleInfo in the `packageContents` command.
Attribute: module-exception	This attribute is used for the `moduleList.moduleStatus.modException` command.
Attribute: module-type	This attribute is used in moduleInfo in the `packageContents` command. The module type for an installable package must be **G2S_game**.
Attribute: release-number	This attribute is used in moduleInfo in the `packageContents` command.
Attribute: status	This attribute is used for the `moduleList.moduleStatus.modStatus` command.

Name	Description
Attribute: filename	Information that points to the installable file within this package, for this module. This attribute is used when directed to install the package through a `setScript` command.
***Element*: storage**	
Attribute: device-class	These attributes are used to load the attributes in the **storageUsed** element of the `moduleList.moduleStatus` command when (and if) the modules in this package are installed.
Attribute: device-id	
Attribute: storage-id	
Attribute: storage-application	
Attribute: storage-used	
***Element*: algorithm**	
Attribute: algorithm-type	This attribute provides a list of the several authentication algorithms supported for this module.
End of Module Element	
***Element*: algorithm**	
algorithm-type	This attribute provides a list of the several authentication algorithms supported for this package.
End of Package Element	

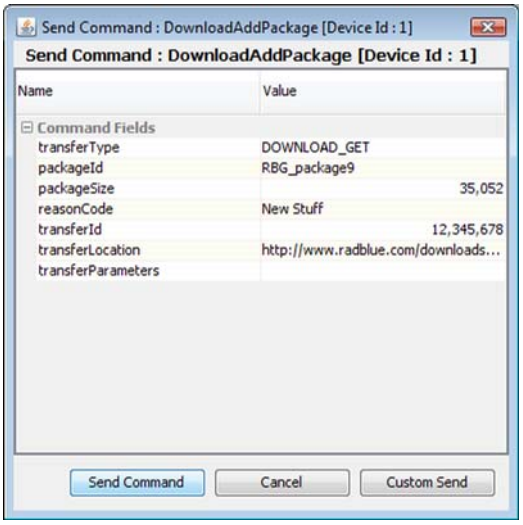
Install a Package Using RGS

Direct the EGM to Download a Package

1. From the RGS, select **SendCommand**.
2. Select **G2S_download** from Current Device.



3. Under the **Available Commands** list, double-click **Download - Add Package**.



4. Set the **Add Package** options as needed:

   - **transferType**: Click the drop-down arrow to indicate whether the host or EGM intiates the transfer as well as the direction of the transfer.
   - **packageId**: Type a unique identifier for the package. The first three characters must be a GSA-assigned manufacturer identifier. If you are using the RadBlue installable package, enter **RBG_package4**.

   Note If you install a package more than once, the package ID **must** be unique each time you add the package. Otherwise, a G2S_DLX002 error will be generated when you send the Set Script for Package command.

   - **packageSize**: Size of package according to the System Management Point (SMP). A value of zero (0) indicates that the file size is unknown.
   - **reasonCode**: Enter the reason for the transfer. *This field is optional.*
   - **transferId**: A host-provided unique identifier. To change the *transferId*, click in the dialog box and type a new value. *This field is optional.*
   - **transferLocation**: Enter the network address of the location on the Software Download Distribution Point (SDDP) where the package resides.

     If you are using the RadBlue installable package, type: http://www.radblue.com/downloads/smartegm/packages/package-4.zip

   Note The `download.addPackage` command supports **file:///** URLs. Use **file:///** to reference a file on the same computer that the RST is running on. The **file:///** convention applies to Java applications, which is used for all RadBlue tools.

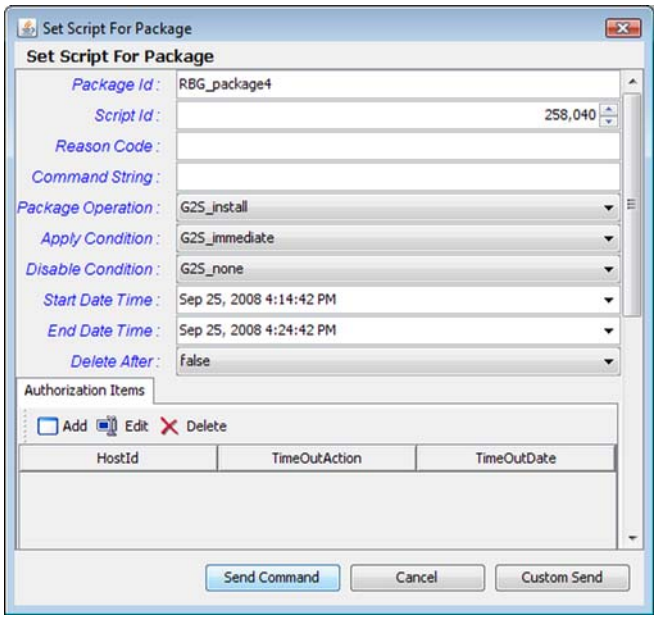
   Example: If you want to load the file **C:/downloads/test.zip**, the file URL would be **file:///c:/downloads/test.zip**. The SmartEGM reads the content directly from the disk, allowing you to add packages if the computer does not have access to an FTP or HTTP server.

   - **transferParameters**: Enter any parameters required for the package transfer. *This field is optional.*

5. Click **Send Command**.

Direct the EGM to Perform an Action on a Package

1. From the RGS, select **SendCommand**.
2. Under the **Available Commands** list, double-click **Download - Set Script for Package**.

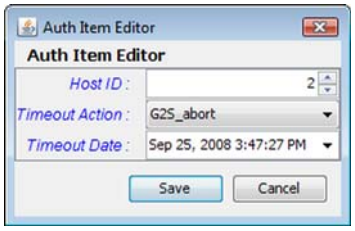


3. Set the **Set Script for Package** options as needed:
   - **Package Id**: Enter the *unique* identifier for the installable package. If you are using the RadBlue installable package, enter **RBG_package4**.
   - **Script Id**: Enter the script's unique identifier, which is provided by the host.
   - **Reason Code**: Enter a description or code that represents the action associated with the package. *This field is optional.*
   - **Command String**: Enter script command elements that you want to execute with the package. The EGM validates command strings prior to starting the script. *This field is optional.*
   - **Package Operation**: Click the drop-down arrow, and select **G2S_install** to install the specified package.
   - **Apply Condition**: Click the drop-down arrow to select the condition under which script command processing starts.
     - Select **G2S_immediate** to apply package changes immediately after required authorizations are complete.
     - Select **G2S_disable** to apply package changes after the EGM has disabled itself (startDateTime, endDateTime, and disableConditions apply).
     - Select **G2S_egmAction** to apply changes after an operator action at the EGM takes place, such as an audit key or menu option.

- Select **G2S_cancel** to cancel changes after validation.

- **Disable Condition**: Click the drop-down arrow to select the circumstances under which the EGM disables prior to the initiation of the script operation.
- **Start Date Time**: Enter the start of the time period that the script operation is allowed to occur. Click the drop-down arrow to select the date and time using a calendar.
- **End Date Time**: Enter the end of the time period that the script operation is allowed to occur. Click the drop-down arrow to select the date and time using a calendar.
- **Delete After**: Click the drop-down to select whether the package should be deleted from the package list and internal storage once the command has completed.

4. If you want to specify that certain hosts should take certain actions in case of a timeout, click **Add** under the Authorization Items tab.



5. Set the **Authorization Items** options as needed:

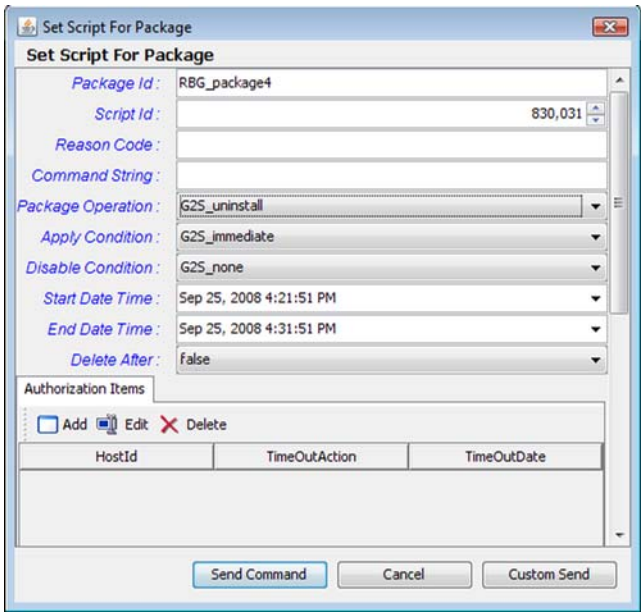
   - **Host ID**: Click in the dialog box, and enter the unique identifier of the host server you want to
   - **Timeout Action**: Click the drop-down arrow to select the type of timeout response you want to send from the specified host. Select **G2S_abort** if you want the host to cancel the configuration change, or select **G2S_ignore** if you want the host to continue with the package operation after a timeout occurs.
   - **Timeout Date**: Enter the date and time when the EGM should timeout if no authorization is received by the specified host. Click the drop-down arrow to select the date and time using a calendar.

6. Click **Save** to add an authentication item to the list.
7. Click **Send Command** to send the `setScript` command.

Uninstall a Package Using RGS

Use this procedure to uninstall an entire package, including all modules within the package.

1. From the RGS, select **SendCommand**.
2. Under the **Available Commands** list, double-click **Download - Set Script for Package**.

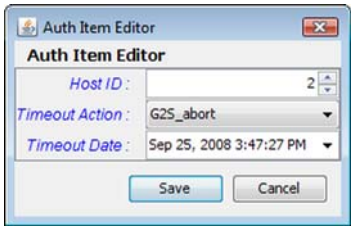


3. Set the following **Set Script for Package** options:
   - **Package Id**: Enter the identifier for the package you want to uninstall.
   - **Script Id**: Enter the script's unique identifier, which is provided by the host.
   - **Package Operation**: Click the drop-down arrow, and select **g2s_uninstall** to uninstall the package.
   - **Apply Condition**: Click the drop-down arrow to select the condition under which script command processing starts.
     - Select **G2S_immediate** to apply package changes immediately after required authorizations are complete.
     - Select **G2S_disable** to apply package changes after the EGM has disabled itself (startDateTime, endDateTime, and disableConditions apply).
     - Select **G2S_egmAction** to apply changes after an operator action at the EGM takes place, such as an audit key or menu option.

- Select **G2S_cancel** to cancel changes after validation.

- **Disable Condition**: Click the drop-down arrow to select the circumstances under which the EGM disables prior to the initiation of the script operation.
- **Start Date Time**: Enter the start of the time period that the script operation is allowed to occur. Click the drop-down arrow to select the date and time using a calendar.
- **End Date Time**: Enter the end of the time period that the script operation is allowed to occur. Click the drop-down arrow to select the date and time using a calendar.
- **Delete After**: Click the drop-down to select whether the package should be deleted from the package list and internal storage once the command has completed.

4. If you want to specify that certain hosts should take certain actions in case of a timeout, click **Add** under the Authorization Items tab.



5. Set the **Authorization Items** options as needed:

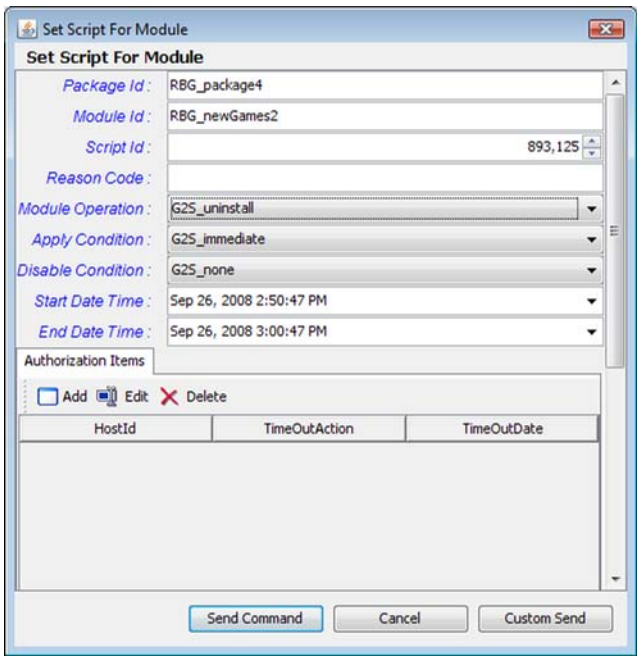
   - **Host ID**: Click in the dialog box, and enter the unique identifier of the host server you want to
   - **Timeout Action**: Click the drop-down arrow to select the type of timeout response you want to send from the specified host. Select **G2S_abort** if you want the host to cancel the configuration change, or select **G2S_ignore** if you want the host to continue with the package operation after a timeout occurs.
   - **Timeout Date**: Enter the date and time when the EGM should timeout if no authorization is received by the specified host. Click the drop-down arrow to select the date and time using a calendar.

6. Click **Save** to add an authentication item to the list.
7. Click **Send Command** to send the `setScript` command.

Uninstall a Module Using RGS

Use this procedure to uninstall a single module from an EGM.

1. From the RGS, select **SendCommand**.
2. Under the **Available Commands** list, double-click **Download - Set Script for Module**.

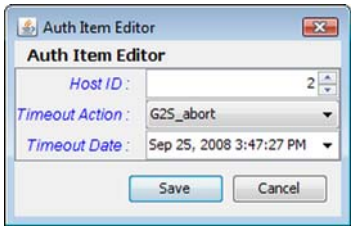


3. Set the following **Set Script for Module** options:
   - **Package Id**: Enter the identifier for the package you want to uninstall. *This field is not required to uninstall a module.*
   - **Module Id**: Enter the identifier for the module you want to uninstall.
   - **Script Id**: Enter the script's unique identifier, which is provided by the host.
   - **Module Operation**: Click the drop-down arrow, and select **g2s_uninstall** to uninstall the module.
   - **Apply Condition**: Click the drop-down arrow to select the condition under which script command processing starts.
     - Select **G2S_immediate** to apply package changes immediately after required authorizations are complete.
     - Select **G2S_disable** to apply package changes after the EGM has disabled itself (startDateTime, endDateTime, and disableConditions apply).
     - Select **G2S_egmAction** to apply changes after an operator action at the EGM takes place, such as an audit key or menu option.

- Select **G2S_cancel** to cancel changes after validation.

- **Disable Condition**: Click the drop-down arrow to select the circumstances under which the EGM disables prior to the initiation of the script operation.
- **Start Date Time**: Enter the start of the time period that the script operation is allowed to occur. Click the drop-down arrow to select the date and time using a calendar.
- **End Date Time**: Enter the end of the time period that the script operation is allowed to occur. Click the drop-down arrow to select the date and time using a calendar.

4. If you want to specify that certain hosts should take certain actions in case of a timeout, click **Add** under the Authorization Items tab.

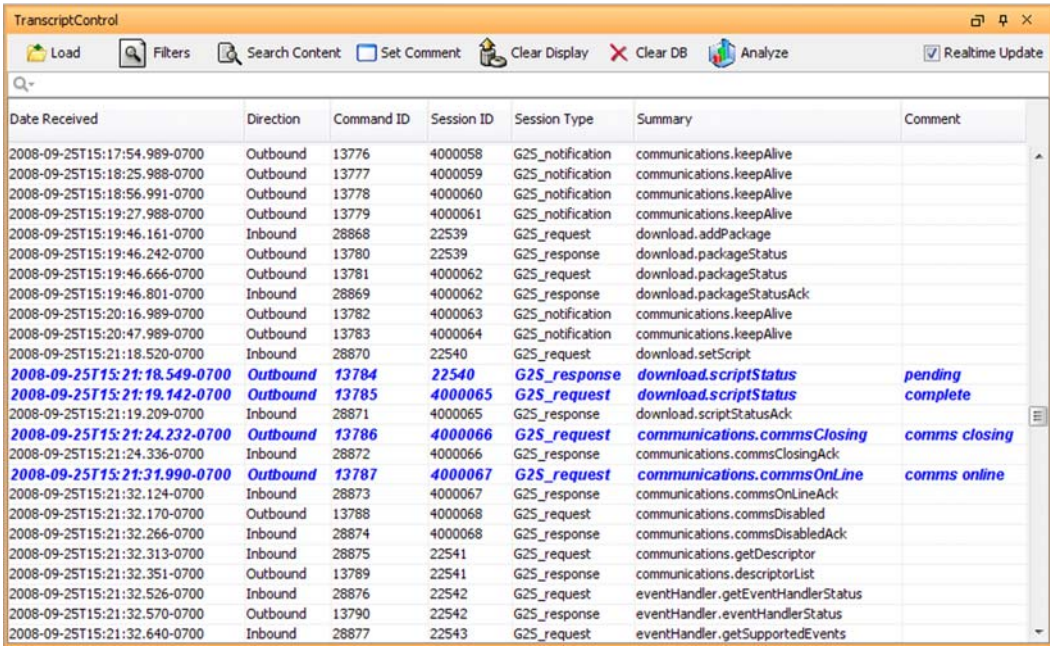


5. Set the **Authorization Items** options as needed:
   - **Host ID**: Click in the dialog box, and enter the unique identifier of the host server you want to
   - **Timeout Action**: Click the drop-down arrow to select the type of timeout response you want to send from the specified host. Select **G2S_abort** if you want the host to cancel the configuration change, or select **G2S_ignore** if you want the host to continue with the package operation after a timeout occurs.
   - **Timeout Date**: Enter the date and time when the EGM should timeout if no authorization is received by the specified host. Click the drop-down arrow to select the date and time using a calendar.
6. Click **Save** to add an authentication item to the list.
7. Click **Send Command** to send the `setScript` command.

View the Message Flow for Installable Packages on RST

The RST SmartEGM lets you view incoming download commands as well as outgoing responses.

To view download messages, select the **Transcript Control** object.



Date Received	Direction	Command ID	Session ID	Session Type	Summary	Comment
2008-09-25T15:17:54.989-0700	Outbound	13776	4000058	G2S_notification	communications.keepAlive	
2008-09-25T15:18:25.988-0700	Outbound	13777	4000059	G2S_notification	communications.keepAlive	
2008-09-25T15:18:56.991-0700	Outbound	13778	4000060	G2S_notification	communications.keepAlive	
2008-09-25T15:19:27.988-0700	Outbound	13779	4000061	G2S_notification	communications.keepAlive	
2008-09-25T15:19:46.161-0700	Inbound	28868	22539	G2S_request	download.addPackage	
2008-09-25T15:19:46.242-0700	Outbound	13780	22539	G2S_response	download.packageStatus	
2008-09-25T15:19:46.666-0700	Outbound	13781	4000062	G2S_request	download.packageStatus	
2008-09-25T15:19:46.801-0700	Inbound	28869	4000062	G2S_response	download.packageStatusAck	
2008-09-25T15:20:16.989-0700	Outbound	13782	4000063	G2S_notification	communications.keepAlive	
2008-09-25T15:20:47.989-0700	Outbound	13783	4000064	G2S_notification	communications.keepAlive	
2008-09-25T15:21:18.520-0700	Inbound	28870	22540	G2S_request	download.setScript	
2008-09-25T15:21:18.549-0700	***Outbound***	***13784***	***22540***	***G2S_response***	***download.scriptStatus***	***pending***
2008-09-25T15:21:19.142-0700	***Outbound***	***13785***	***4000065***	***G2S_request***	***download.scriptStatus***	***complete***
2008-09-25T15:21:19.209-0700	Inbound	28871	4000065	G2S_response	download.scriptStatusAck	
2008-09-25T15:21:24.232-0700	***Outbound***	***13786***	***4000066***	***G2S_request***	***communications.commsClosing***	***comms closing***
2008-09-25T15:21:24.336-0700	Inbound	28872	4000066	G2S_response	communications.commsClosingAck	
2008-09-25T15:21:31.990-0700	***Outbound***	***13787***	***4000067***	***G2S_request***	***communications.commsOnLine***	***comms online***
2008-09-25T15:21:32.124-0700	Inbound	28873	4000067	G2S_response	communications.commsOnLineAck	
2008-09-25T15:21:32.170-0700	Outbound	13788	4000068	G2S_request	communications.commsDisabled	
2008-09-25T15:21:32.266-0700	Inbound	28874	4000068	G2S_response	communications.commsDisabledAck	
2008-09-25T15:21:32.313-0700	Inbound	28875	22541	G2S_request	communications.getDescriptor	
2008-09-25T15:21:32.351-0700	Outbound	13789	22541	G2S_response	communications.descriptorList	
2008-09-25T15:21:32.526-0700	Inbound	28876	22542	G2S_request	eventHandler.getEventHandlerStatus	
2008-09-25T15:21:32.570-0700	Outbound	13790	22542	G2S_response	eventHandler.eventHandlerStatus	
2008-09-25T15:21:32.640-0700	Inbound	28877	22543	G2S_request	eventHandler.getSupportedEvents	

In the example above, installable package messages that are generated by RST after a `setScript` command are received.

Once authorization from all hosts have been received, RST automatically restarts in order to update its data model. This process is the same whether the `setScript` operation is to install or uninstall a package.

The outbound message flow from the RST is:

1. `download.scriptStatus` (status: *in progress*)
2. `download.scriptStatus` (status: *complete*)
3. `communications.commsClosing`
4. `communications.commsOnline`

By double-clicking any message, you can see the XML content.

View SmartEGM Data Model Updates on RST

The SmartEGM Data Model Viewer allows you to see the current EGM data model. Any data model changes that are contained in a package can be viewed in the Data Model Viewer. Simply open the SmartEGM, and select the **Data Model Viewer** tab. Changes take effect after RST restarts on a `download.setScript` command.