


Configuring Actions for the Database Engine Processor

You can configure actions that the database engine processor performs on events. For example, you can set up an action to calculate usage in a service-tracking event by using the usage metric that you configured for a service.

To configure actions for the database engine processors:

1. In the VTA Configuration Manager navigation pane, select **Edit**.
2. Under Current Configuration, select **Actions**.

A list of actions appears. For example:



VTA Configuration Manager
Edit Configuration

- Home
- Dir Connection
- Configuration
 - Create
 - Edit
 - Load
 - Commit
 - Import
 - Export

Current Configuration


Event Handlers
Actions
Processors

Quota VTA Actions

GetAccountBalances	Delete
CalcUsage	Delete
DebitAccounts	Delete
CalculateInterim	Delete
SetInterim	Delete
StopQuotaService	Delete
StartQuotaLocalService	Delete
StartQuotaInternetService	Delete
TerminateSession	Delete

New Quota VTA Actions

Create



- To add an action, enter a name for the action in the New Quota VTA Actions field, and click **Create**.

The action configuration screen appears.

CalculateInterim					
Name	Value				
Processor	DBEngine ▼				
Function	CalculateInterim				
Parameter ⓘ	<table border="1"> <thead> <tr> <th>Parameter Name</th> <th>Parameter Content</th> </tr> </thead> <tbody> <tr> <td colspan="2">Add Parameter</td> </tr> </tbody> </table>	Parameter Name	Parameter Content	Add Parameter	
Parameter Name	Parameter Content				
Add Parameter					
Abort On Error	<div> <input type="text"/> <input type="button" value="Enable"/> </div>				
<input type="button" value="Save"/>					

4. Select DBEngine in the Processor field, and click **Save**. (If DBEngine does not appear in the drop-down list, enable the database engine processor.)

An expanded configuration screen for the action appears.

RefillBucketWithBehavingRate											
Name	Value										
Processor	DBEngine ▼										
Function	UpdateAccounts										
Parameter ⓘ	<table border="1"> <thead> <tr> <th>Parameter Name</th> <th>Parameter Content</th> <th></th> </tr> </thead> <tbody> <tr> <td>ScriptName</td> <td>RefillBucketWithBehavingRate</td> <td> <input type="button" value="Delete"/> <input type="button" value="Disable"/> </td> </tr> <tr> <td colspan="3">Add Parameter</td> </tr> </tbody> </table>	Parameter Name	Parameter Content		ScriptName	RefillBucketWithBehavingRate	<input type="button" value="Delete"/> <input type="button" value="Disable"/>	Add Parameter			
Parameter Name	Parameter Content										
ScriptName	RefillBucketWithBehavingRate	<input type="button" value="Delete"/> <input type="button" value="Disable"/>									
Add Parameter											
Abort On Error	<div> <input type="text"/> <input type="button" value="Enable"/> </div>										

5. Edit the action fields.

See “Action Fields for the Database Engine Processor” on page 4 .

6. If you are finished configuring the SRC-VTA, save the configuration to a directory or local file.

See Committing a VTA Configuration to a Directory .

- Related Topics**
- Database Engine Processor Fields
 - Configuring a Usage Metric for Service Accounts
 - Configuring VTA Actions to Run Scripts
 - Configuring Actions for the SAE Proxy Processor

Action Fields for the Database Engine Processor

In VTA Configuration Manager, you can edit the following fields in the Quota VTA Actions screen.

Processor

- Processor for which you are configuring the action.
- Value—DBengine
- Default—No value

Function

- Function calls that the database engine processor invokes. These functions are variables that you can use to update database accounts.
- Value
 - CalculateInterim—Calculates the interim interval in the service-tracking event by using the interim interval configured for the service. It has no parameters. It adds the following attribute to the event after the function is executed:
 - interimInterval—Interim interval of the service
 - CalculateUsage—Calculates usage in the service-tracking event by using the usage metric configured for the service. It has no parameters. It adds the following attributes to the event after the function is executed:
 - currentUsage—Usage since the previous usage report
 - sessionSinceLastReport—Session length since the previous usage report
 - GetAccounts—Gets account data for the corresponding subscriber for the event. Subsequent event handlers of the event can use the retrieved data. It has no parameters. It adds the following attributes to the event after the function is executed:
 - balance_ < accountName > —Balance for the account.
 - lastUpdateTime_ < accountName > —Last update time in milliseconds since January 1, 1970 UTC for the account.
 - status_ < accountName > —Status of the account.
 - TerminateSession—Closes active VTA sessions that have a status of Start or Interim. It does not stop the corresponding services in the SAE. You can use this function to stop a service at the end of a billing period. Usage data collected after the VTA session is stopped is stored in new VTA session records.
 - UpdateAccounts—Runs an account update script that changes the account balances of the corresponding subscriber for the event. It adds the following attributes to the event after the function is executed:
 - balance_ < accountName > —Balance for the account after it is updated.

- lastUpdateTime_ < accountName > —Last update time in milliseconds since January 1, 1970 UTC for the account after it is updated.
- status_ < accountName > —Status of the account after it is updated.

- Default—No value

Parameter Name

- Parameters to pass to the function.
- Value
 - Event attribute name; the event attribute name is replaced by the attribute's value wherever it appears.
 - For the UpdateAccounts function, you can configure the following parameter:
 - scriptName—Name of the account update script defined in the database engine processor
- Example—ScriptName

Parameter Content

- Values for the parameters.
- Value
 - For the scriptName parameter—Name of the account update script defined in the database engine processor
- Example—DebitQuotaUsage

Abort On Error

- Disables or enables and sets the processing in response to an error.
- Value
 - Break—Stop processing the current event.
 - Continue—Continue with the next action, if any, in the same event handler.
 - Next Event Handler—Continue with the next event handler (if any).
- Default—No value

