I think two areas apply to the concept in this bug (setting alert operations on remote resources). It's mentioned in a section in the overview and then alluded to in setting alert operations. If this isn't sufficient, let me know.

http://elladeon.fedorapeople.org/jon-drafts/3.0/admin/alerts.html#about-alert-operations

8.1.3. Alert Operations

When a certain alert condition occurs, the JBoss ON agent can launch a response on ~~that~~a resource through an operation. This is part of the alert definition configuration, but it's worth calling out because it is such a useful tool for managing responses to alerts. Whenever an alert is fired, the agent can perform some kind of action, like restarting a server. This can be done either on the resource which issued the alert or on another resource.

Remote operations can be exceedingly useful (and versatile). For example, a JBoss server may begin performing badly because its JVM is out of memory. The ~~JBoss server~~JVM is the resource which issues its alert, but the response by the agent is to restart the ~~JVM~~JBoss server.

Regular operations are either initiated immediately or run on defined schedules. Alert operations are executed when the alert is fired and are even more flexible because *any* resource in the JBoss ON inventory can be the target of the alert operation, not only the resource which sent the alert. That means that an operation can be run for a different application on the same host server or even on an entirely different server.

NOTE

The operations performed in response to an alert are the same as the operations which can be scheduled to run on a resource. The operations available for an alert depend on the target resource on which the operation will run — not the resource where the alert is set.

The type of operation which is available to be run for an alert depends on the type of resource that is the target of the operation. (This may not be the same as the resource which has the alert configured.) There are two types of alert operations:

- Operations that are the same as regular operations.

- Scripts, such as JavaScript or Ruby, that can be run on any resource . Those are also regular operations of the Script(2) resource

- The specialiced script alert sender is not part of JbossON -

TIP

Alert operations senders can be used to run scripts on remote resources. For example, if a resource goes down, a diagnostic script can be run on its parent platform or another resource can be brought online and properly configured to take its place.

Tip: it is possible to run multpile operations sender for one alert definition, which can trigger different operations. All notifications are executed in the order they are listed in the definition.

http://elladeon.fedorapeople.org/jon-drafts/3.0/admin/assigning-alert-operations.html

8.3. Assigning an Operation to an Alert

To set an alert operation, select the **Operations** alert method when configuring notifications. Operations can perform tasks or run scripts on a target resource; this is detailed in Section 8.1.3, "Alert Operations" and correlates to using resource operations, as described in Chapter 6, *Managing Operations*.

1. Configure the basic alert definition, as in Section 8.2, "Setting Alerts for a Resource".

2. Click **OK** to get to the next screen.

3. The bottom of the page has the **Notifications** section. Click the **Edit** button to add a response.

4. Give the notification method a name, and select the **Resource Operations** method from the **Alert Senders** drop-down menu.

5. Select the new alert sender in the table. This opens the configuration for the operation.

6. First, set the resource that the operation will run on. The default is the resource that the alert is set for; it is also possible to set it on another specific resource or on the results of a search.

7. Select the operation type. The available operations and their configuration parameters depend on the type of resource selected as the target of the operation.

   The *Resource Monitoring Reference* lists the available operations per resource type. Chapter 6, *Managing Operations* has more information on setting operations in general.

8. Configure the parameters of the operation. The available settings depend on the type of operation selected. Note that the parameters allow fro substitution of tokens in the form of <% space.name %> like e.g. <% alert.url %> or <%resource.id %>

http://elladeon.fedorapeople.org/jon-drafts/3.0/admin/assigning-alert-scripts.html

8.4. Initiating Scripts from an Alert

To set an alert operation, select the **Operations** alert method when configuring notifications, and then . This is the same as using an operation to execute a script, as described in Section 6.6, "Running Scripts as Operations".
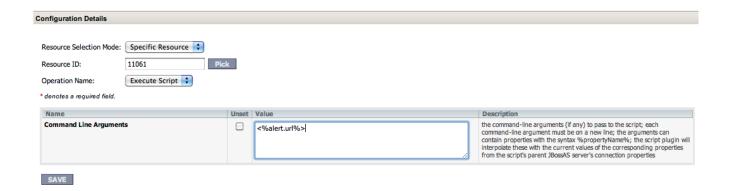
NOTE

The script must be uploaded to the resource and added into the JBoss ON inventory before it can be used in an alert operation.

1. Upload the script to the resource on which is should run in response to the alert. If necessary,

run manual discovery to detect and add the script. See <u>Section 3.1.2, "Manual Discovery"</u>.

2. Configure the basic alert definition, as in <u>Section 8.2, "Setting Alerts for a Resource"</u>.
3. Click **OK** to get to the next screen.
4. The bottom of the page has the **Notifications** section. Click the **Edit** button to add a response.
5. Give the notification method a name, and select the **Resource Operations** method from the **Alert Senders** drop-down menu.
6. Select the new alert sender in the table. This opens the configuration for the operation.
7. Select the script resource that will be run in response to the alert.

The script resource allows to pass tokens to the script in the form %foo% where foo is a java property of the sripts parent jboss on server (see Description box on next screen shot)

Configuration Details

Resource Selection Mode:	Specific Resource	
Resource ID:	11061	Pick
Operation Name:	Execute Script	

* denotes a required field.

Name	Unset	Value	Description
Command Line Arguments	☐	<%alert.url%>	the command-line arguments (if any) to pass to the script; each command-line argument must be on a new line; the arguments can contain properties with the syntax %propertyName%; the script plugin will interpolate these with the current values of the corresponding properties from the script's parent JBossAS server's connection properties

SAVE

This is a two step process

1: <% space.name%> is resolved by the alert operations sender engine and the result of this is passed into the script plugin, which then itself runs token substitution on the %foo% tokens.

<% space.name %> is available for all operations, while %foo% is specific to the script resource.