JBoss Operations Network

Getting Started Guide

# Table of Contents

1 Introduction

This document is meant to be a brief guide for getting started with the JBoss Operations Network (JON) Management Server. JON is used to monitor and manage JBoss instances across a distributed environment.

This guide is not going to dive deep into the architecture of JON, but it's important to understand a few high level concepts. There are two fundamental pieces of JON. First, the JON Management Server is the web based application in the center of the graphic below. This is the web application that users log into to monitor and manage their systems. Second, the JON Agents are light weight daemon processes that run on each operating system being managed. These agents mediate between the JON Management Server and the various resources that JON manages (such as application servers, databases, web services, ESB services, EJBs, file systems, etc.).

The JON Management Server can be used to "discover" resources running on operating systems that have a JON Agent running, "configure" those resources, "deploy" to those resources, and "manage" those resources.

The JON Management Server can connect to the JBoss Customer Support Portal (CSP) to download patches. It can also generate alerts that send emails or send SNMP traps to third party management tools.

The JON Management Server also connects to a database so that it can store off metrics allowing users to see historical metrics for days, weeks, months, and even a year or more. In addition, the JON Management can interface with an LDAP server to perform user authentication.
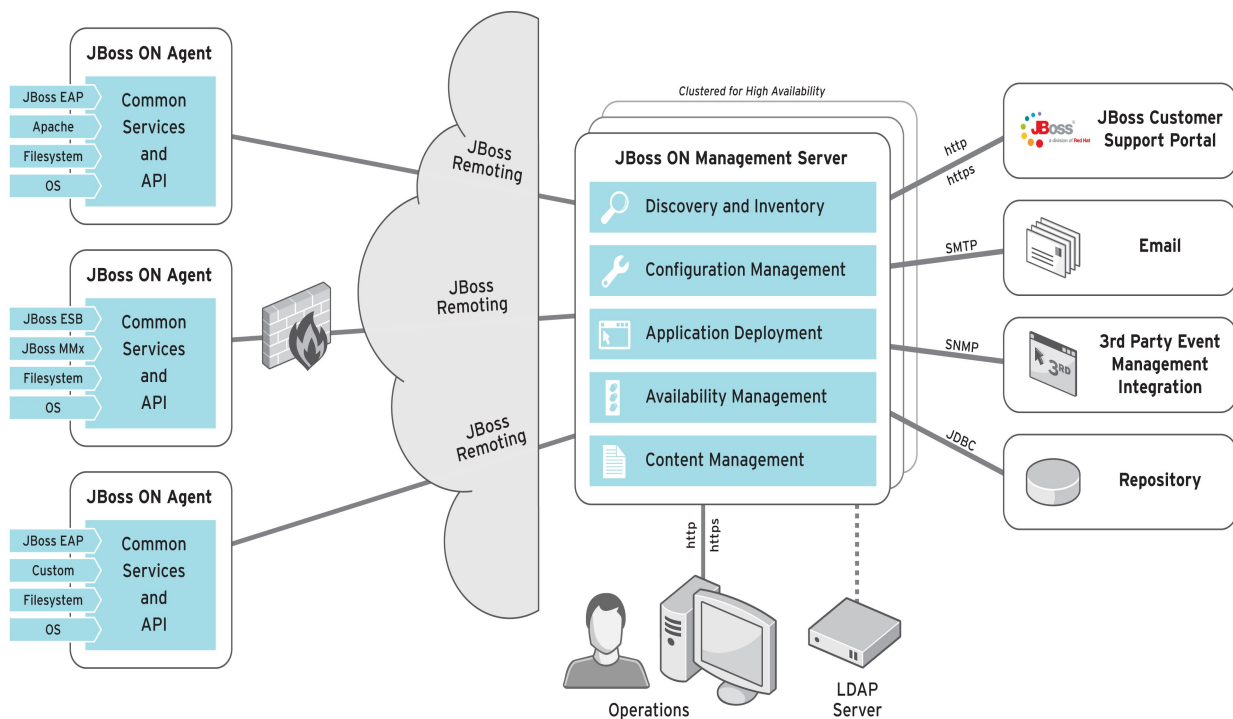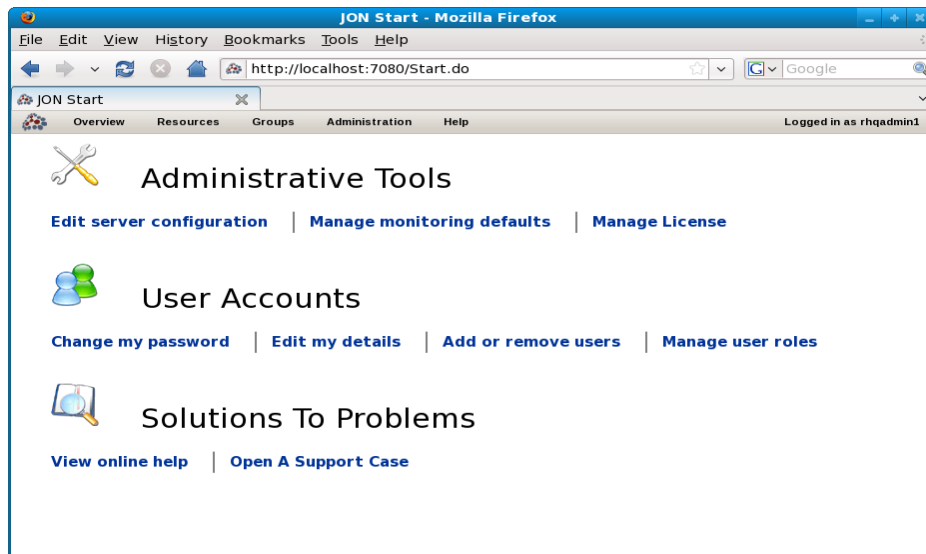
2 Login

The first screen users see when accessing JON is the login screen. To access JON, you will need to enter appropriate credentials (username and password).
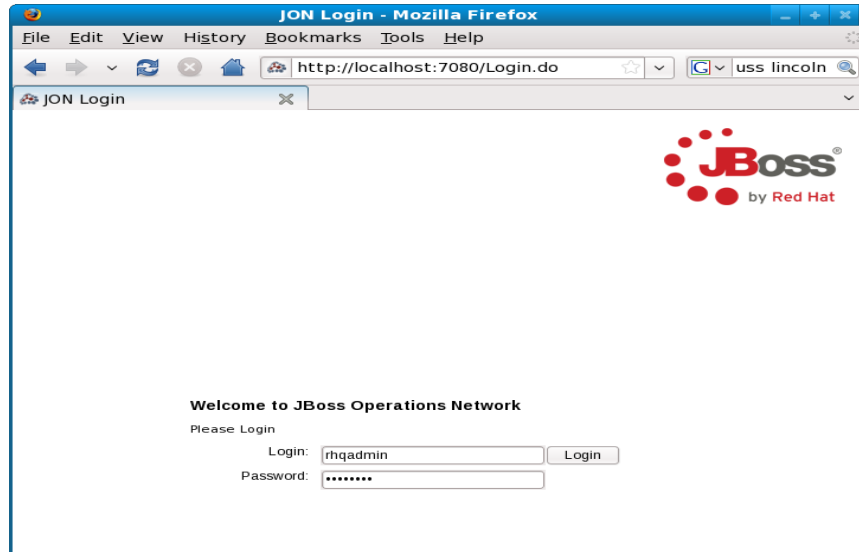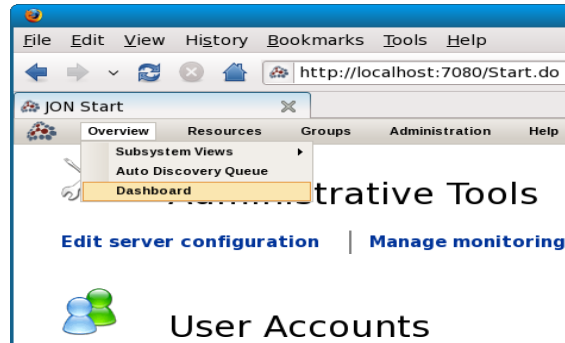

3 Dashboard

Once you have logged in, you will see the initial start page as shown below.

Generally, users will want to go to their Dashboard to get a high level overview of the system. This can be done by selecting the "Overview | Dashboard" link from the JON menu bar at the top of the screen as shown below.

The Dashboard will look similar to what is shown below. We'll briefly discuss each of the components of the Dashboard.
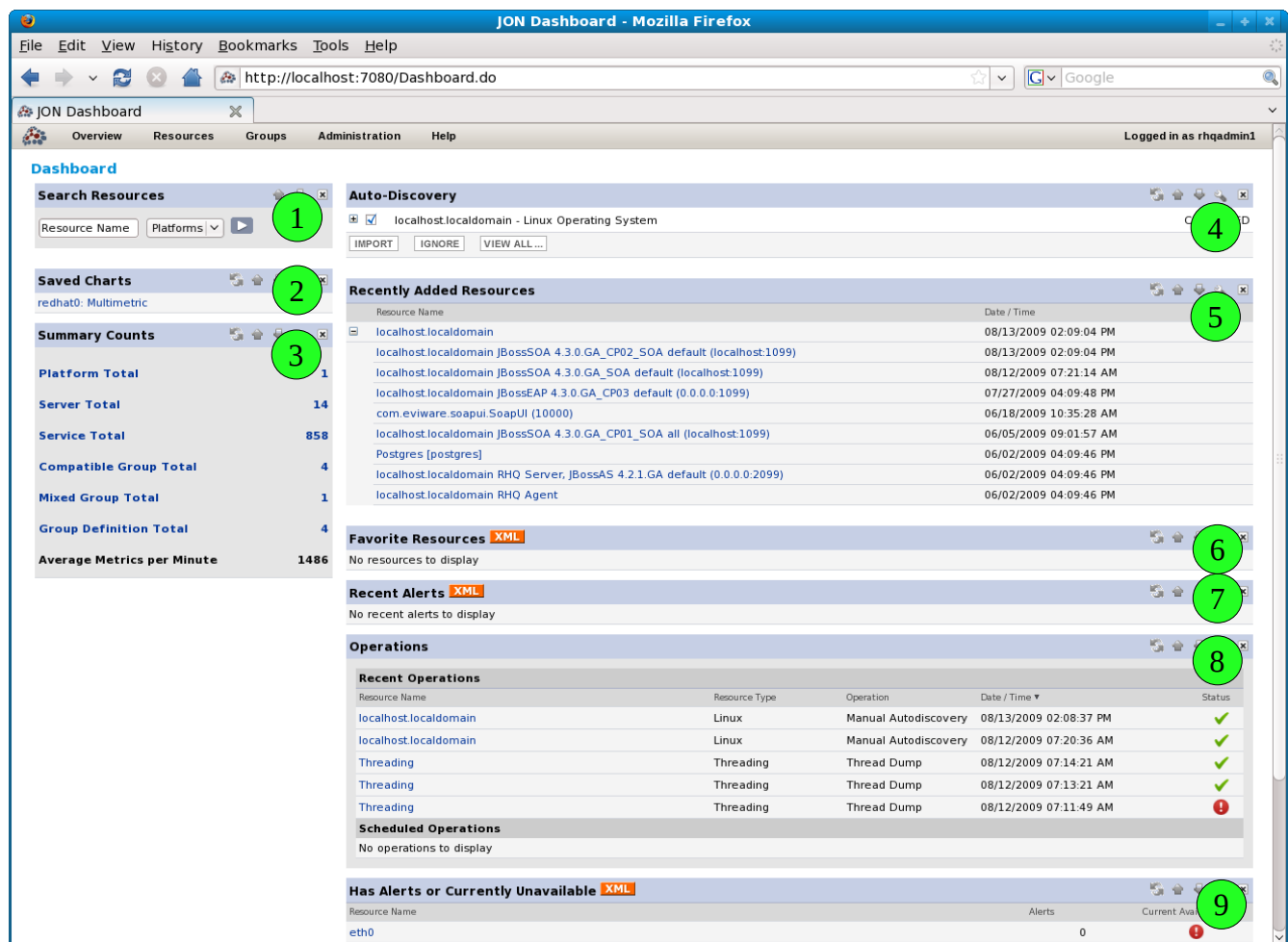
1.) Search Resources – This portlet allows you to search for any type of resource. A resource may be a server machine, a JBoss instance, a container on the JBoss instance, or even something to the level of a specific ESB service or deployed WAR file.

2.) Saved Charts – This portlet provides a link to any charts that the current user has elected to

save. We will see how to save links to charts later.

3.) Summary Counts – This provides a summary and categorized view of all resources accessible to the current user.

"Platforms" are analogous to an instance of an operating system. Generally this equates to a physical machine as well. However, if there are multiple virtual machines running on a single physical machine, this will show up as multiple platforms in JON.

"Servers" are generally analogous to processes running on a Platform such as a JBoss server instance, Tomcat instance, Postgresql database, etc. However, in some cases there are multiple "Servers" for a single process. The most common case of this is the JBoss application server process which has the JBoss "Server" as well as the embedded Tomcat "Server".

"Services" is the most broad category. This includes things such as EJBs, WARs, ESB services, JMS queues, etc.

"Groups" are a way of combining multiple resources into an aggregated view. For example, one might create a group of servers that are in a cluster so that aggregated metrics for the cluster can be viewed.

4.) Auto Discovery – This is a portlet that shows resources discovered by an agent that have not been "managed" yet. Discovered resources are not actually managed/monitored until they have been discovered by an agent and selected from this portlet for "import".

5.) Recently Added Resources – One a users has "Imported" resources that were auto discovered, they will show up in this portlet confirming that they are now being managed.

6.) Favorite Resources – These are resources that the user has selected as a "Favorite". It does nothing other than show a list of resources that the current user looks at often and wants a quick way to view.

7.) Recent Alerts – Users can configure "Alerts" on any metrics that they view. An example of an alert a user could configure might be when a JVM is using over 80% of its allocated memory or when an ESB service action takes longer than10 seconds. Alerts can send emails or generate SNMP traps, but can also show up in this portlet.

8.) Operations – These are actions that have recently been executed on managed resources. Some examples of operations are starting/stopping servers, viewing hibernate statistics, clearing hibernate statistics, etc.

9.) Has alerts or currently unavailable – As the title indicates, this shows a list of resource that generated alerts or are not available. For example, if a server that was being managed crashes or does not get started, it will show up in this list.

The dashboard is configurable. Using the control buttons at the top right of each portlet, a user can refresh, move up, move down, maximize, or remove these portlets from their dashboard. When portlets

are removed from the dashboard, a new portlet will be added to the bottom of the column to allow re-adding the portlet in the future if desired.

After a quick glance at their dashboard, a user will generally want to view more details about the resources they are managing. Typically, this will be done by switching to the "Resource | Platforms" view as shown below.
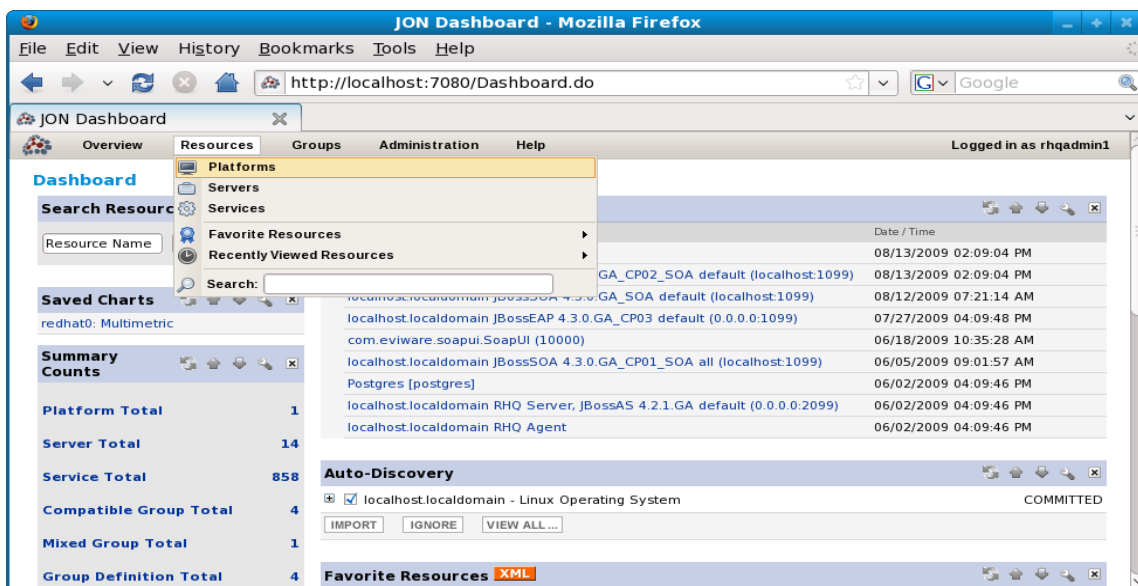

This will show a view like this.

This provide a more detailed view of the resources counted in the "Summary Counts" portlet of the user's dashboard that was discussed above. From this view, the user can see each Platform (OS), Server (managed process), Service (lower level resource), as well as the groups that have been defined.

Clicking on any resource in this view will display the resource in a tree view.

4 Tree View
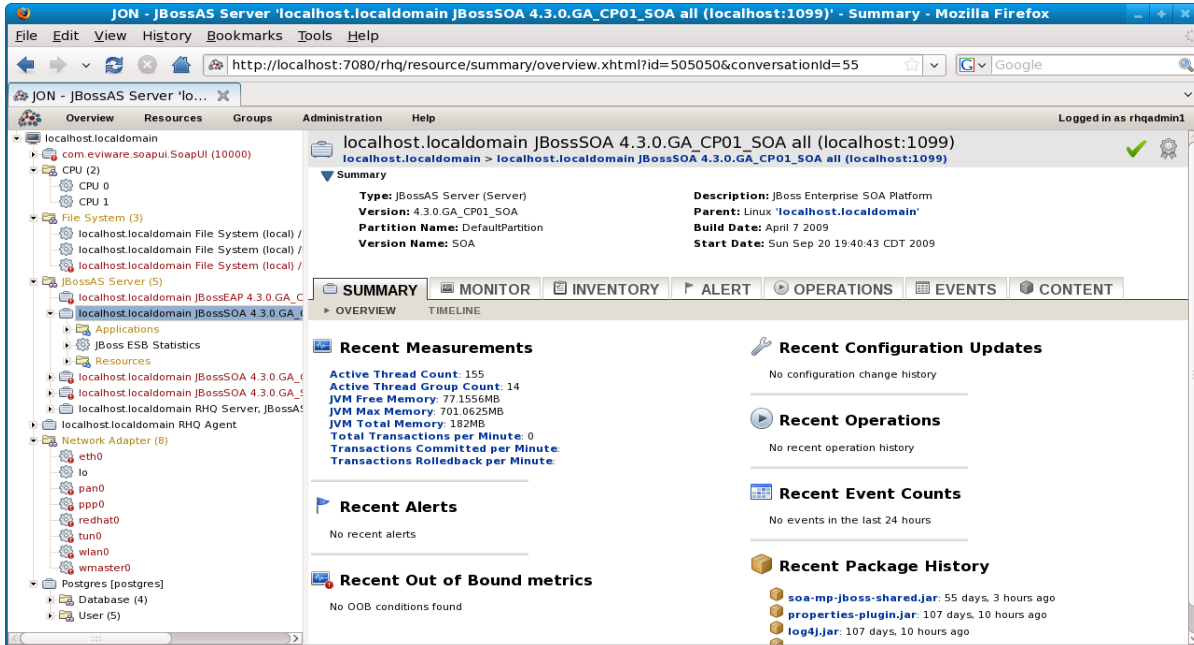
This view shows a hierarchical view of all the resources under a given platform. In this tree, one can see CPU resources, file system resources, JBoss server resources, network resources, database resource, and others. Most of these resources are folders as well and can be expanded to show more detailed child resources.

When a particular resource is selected in the tree on the left, the panel on the right will be updated with information for the selected resource.

Depending on the resource you select, there may be more or less tabs displayed on the right because some resources allow more monitoring/management than other resources. Following is an overview of each of the tabs that may be available for a resource selected in the tree view.

4.1 Summary Tab
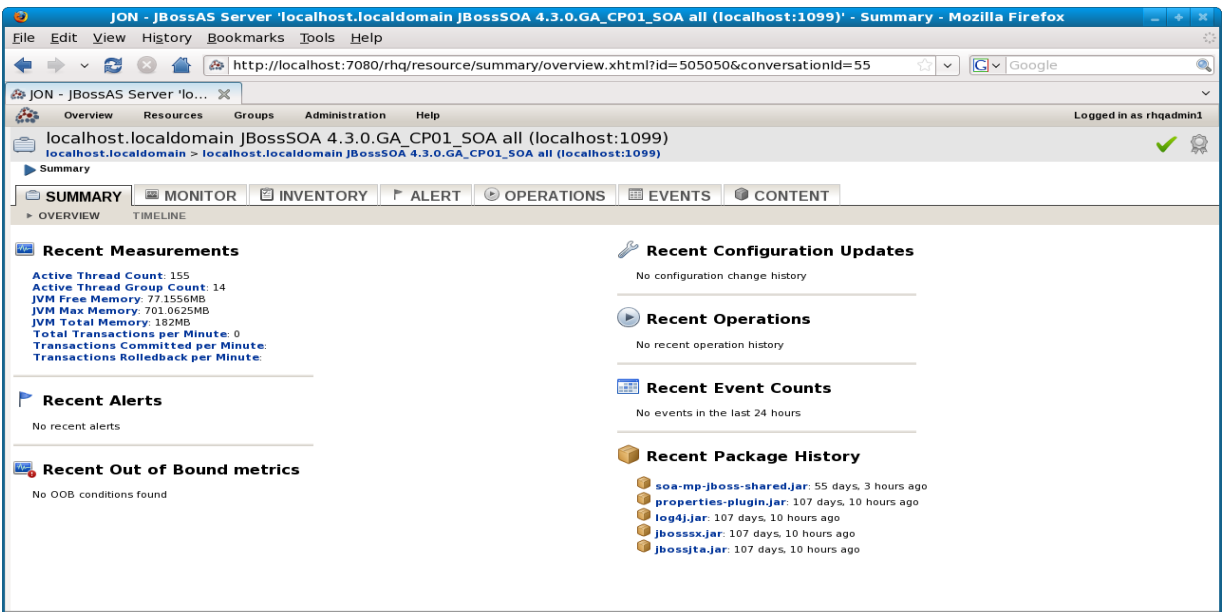
As shown above, the summary tab shows summary information for the selected resource. The attributes considered "Summary" are defined in the plugin for the given resource. Within the Summary tab, there are Overview and Timeline sub-tabs. The Timeline subtab shows summary events on a graphical timescale.

4.2 Monitor Tab
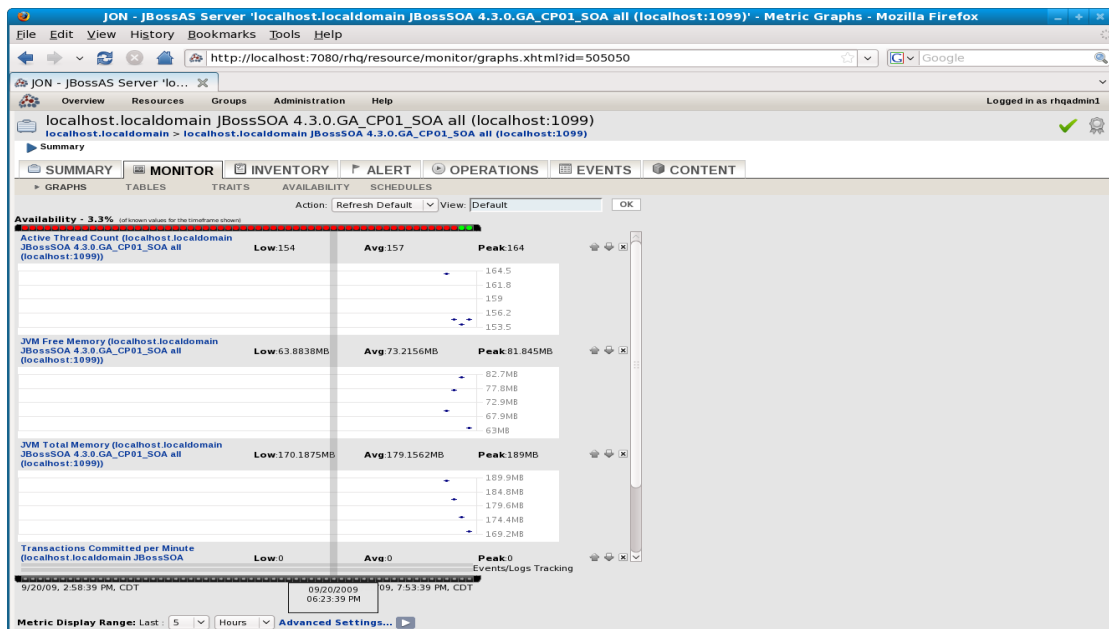
The Monitor tab is the primary area where summary metrics for this resource can be displayed. Within the Monitor tab, there are five sub-tabs: Graphs, Tables, Traits, Availability, and Schedules. The Graphs sub tab shown above shows graphs for the summary metrics.

The Tables subtab shown below shows all the enabled metrics.
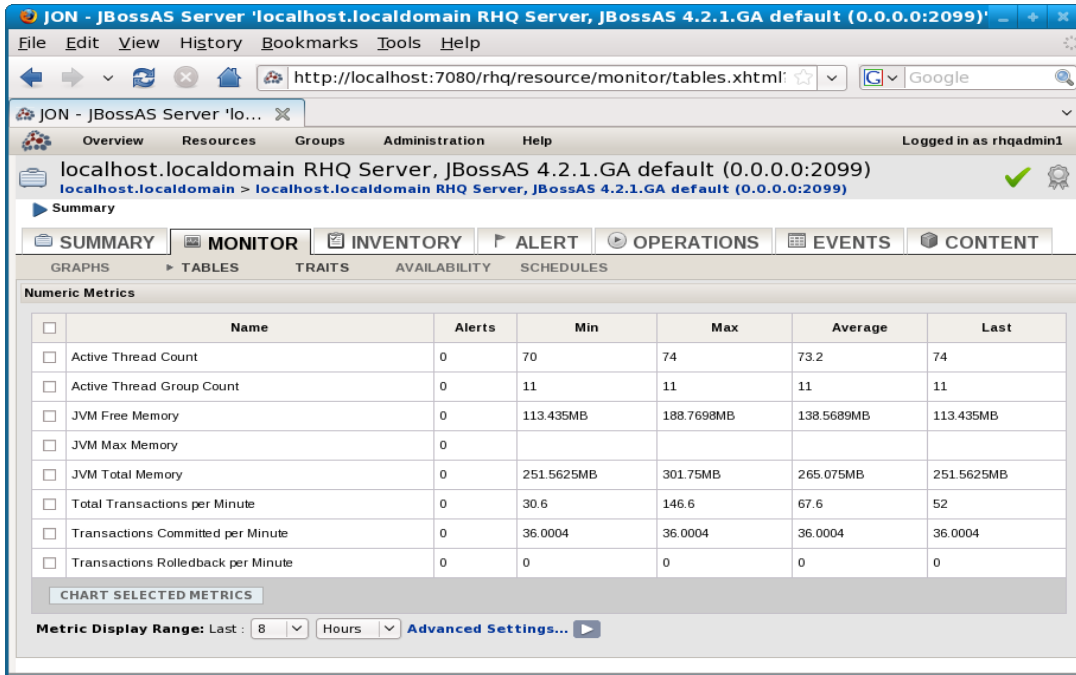
By selecting a checkbox next to a metric and selecting the "Chart Selected Metrics" button, the user can see a chart for the given metric as shown below. By selecting the "Save Chart to Dashboard" link, a link to this chart can be added to the user's dashboard.
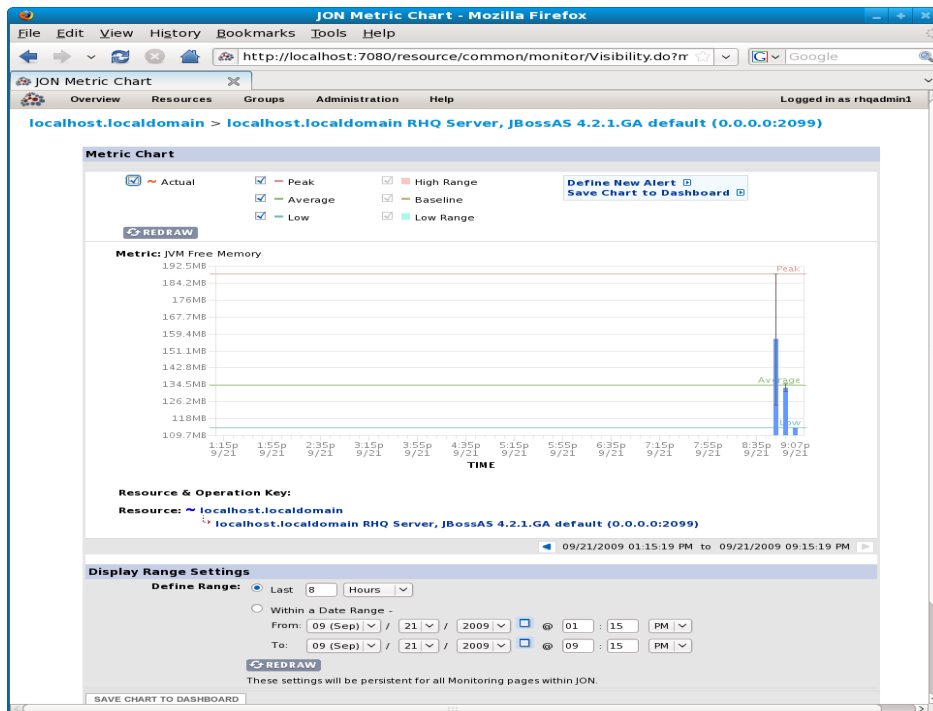
The Traits sub tab shows all non-metric traits for the give resource as shown below. Traits are static attributes for a resource.
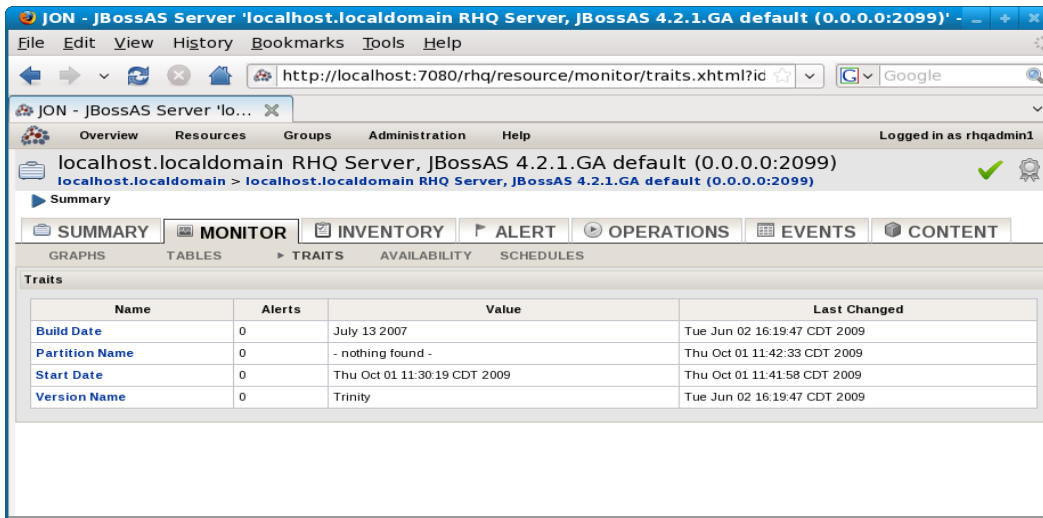
The Availability sub tab shows historically when this resource has come up or down (become available or unavailable).
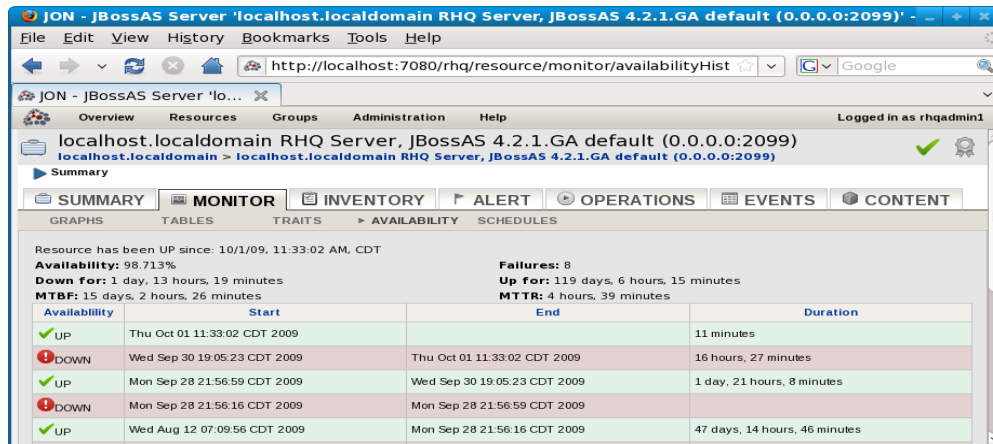
The Schedules sub tab shows all the available metrics for this resource, whether or not they are enabled for collection, and what the collection interval is. This tab can also be used to change this information.
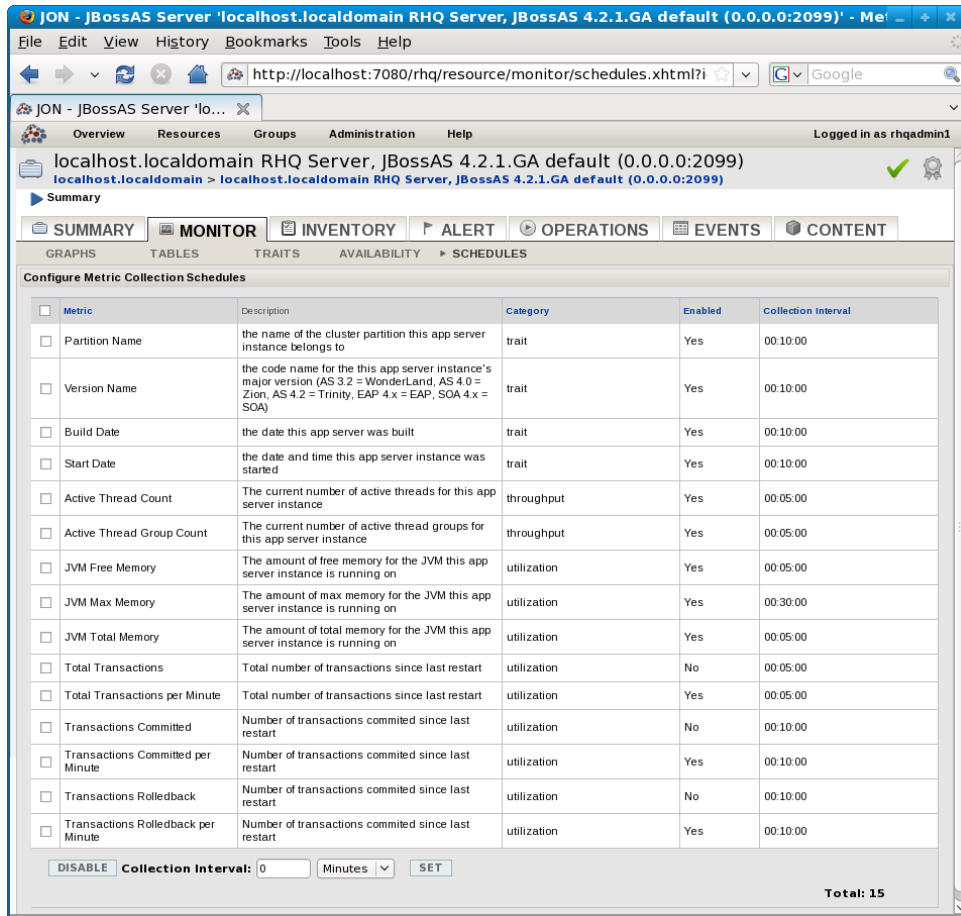
4.3 Inventory Tab

The Inventory tab has three sub tabs: Overview, Connection, and Agent.

The Overview sub tab as shown below displays a list of all child resources for the resource we are looking at. The child resources can be filtered to display a smaller list of resources. Also, if the resource supports adding new children (like EARs, WARs, etc.), then those can be added (deployed) here.

The Connection sub tab shows any settings that are required for this JON resource to connect to the actual entity it is monitoring. In this case below, for example, one may need to specify the JMX username/password so that this resource can actually connect to the application server it is representing.
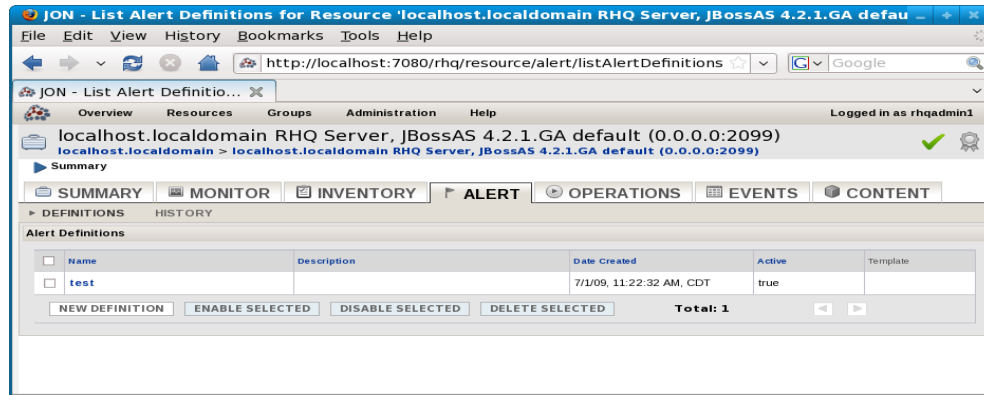
Lastly, the Agent sub tab provides information about the JON agent that is managing (reporting metrics for) this resource.
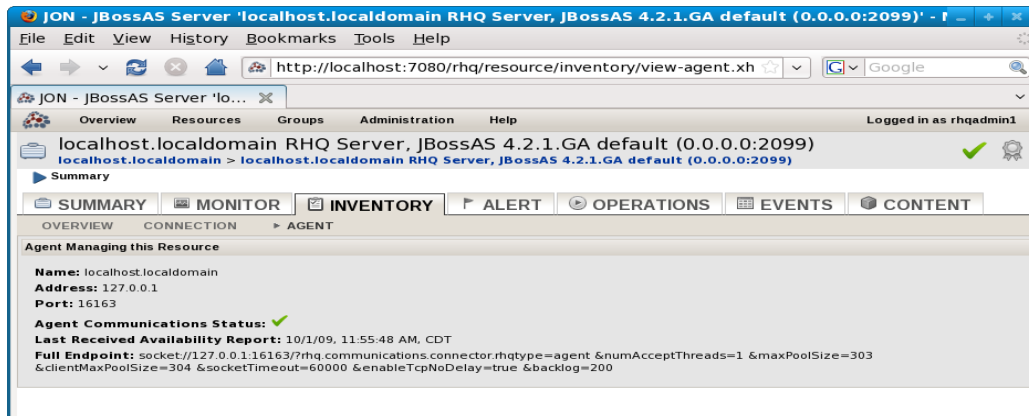
4.4 Alert Tab

The Alert tab has two sub tabs: Definitions and History

The Definitions sub tab as shown below is used to create alert definitions. An alert is how administrations can be proactively notified of certain conditions of interest (high memory, high cpu utilization, long response time, or a combination of conditions). Alerts can send emails or SNMP traps.

By clicking the "New Definition" button, a user can create a new Alert as shown below.

After defining the alert and clicking "Ok", the next screen will allow specifying if this alert should notify roles of JON users or specific email addresses as shown below.
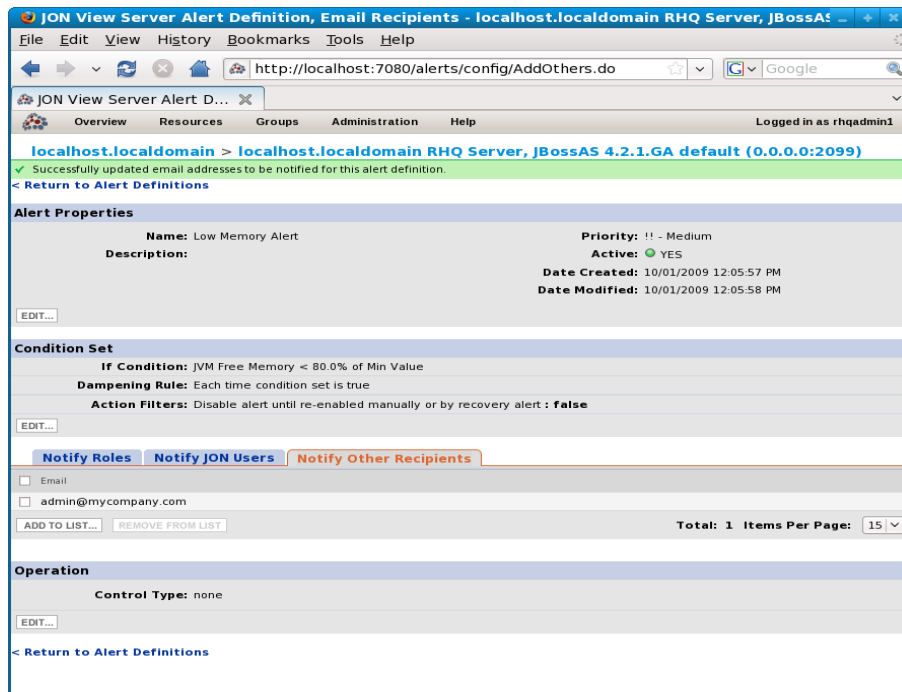
The History sub tab shows any recent alerts that have fired on this resource.
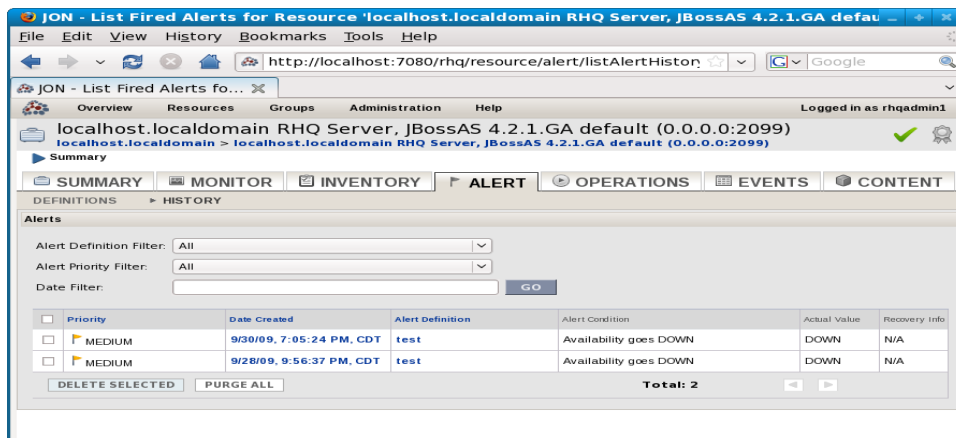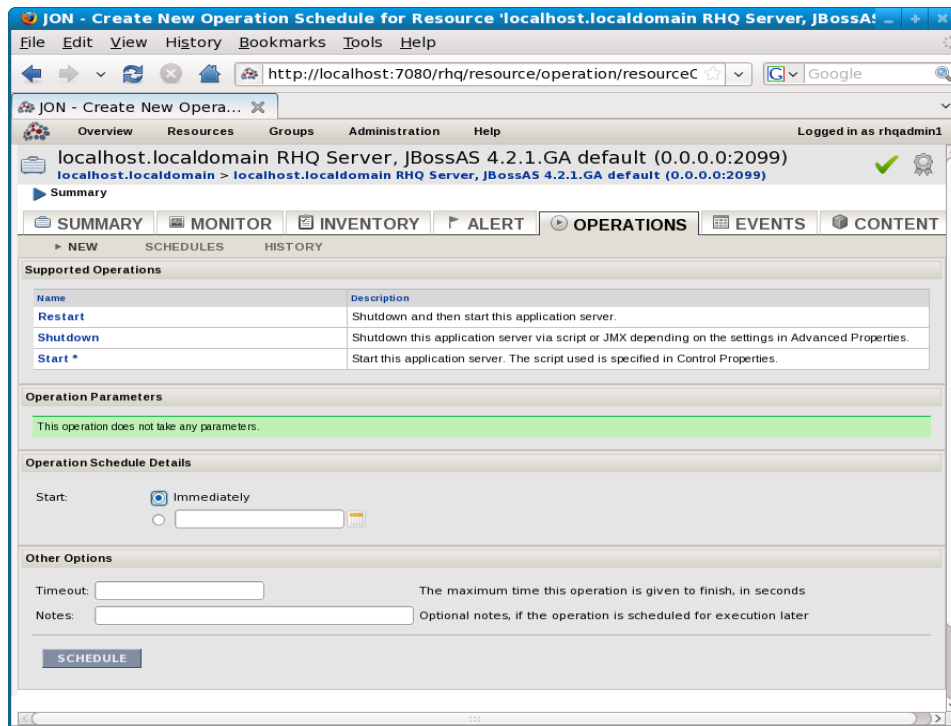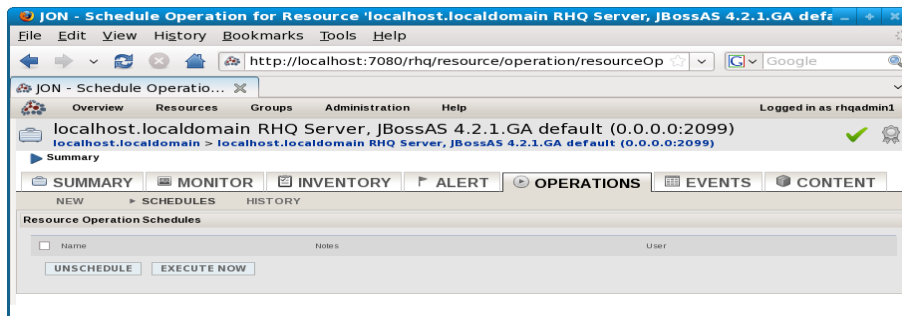

4.5 Operations Tab

The Operations tab is used to run operations on the selected resource, for example: start, stop, clear cache, etc. All operations are run asynchronously, even if they are very quick. Therefore there is the notion of "scheduling" operations. Operations can be scheduled to run immediately or according to a more complex schedule like a point in the future or a recurring schedule. The Operations tab has three sub tabs: New, Schedule, and History.

The New sub tab shows what operations can be schedule and when an operation is selected, it will show the scheduling options.
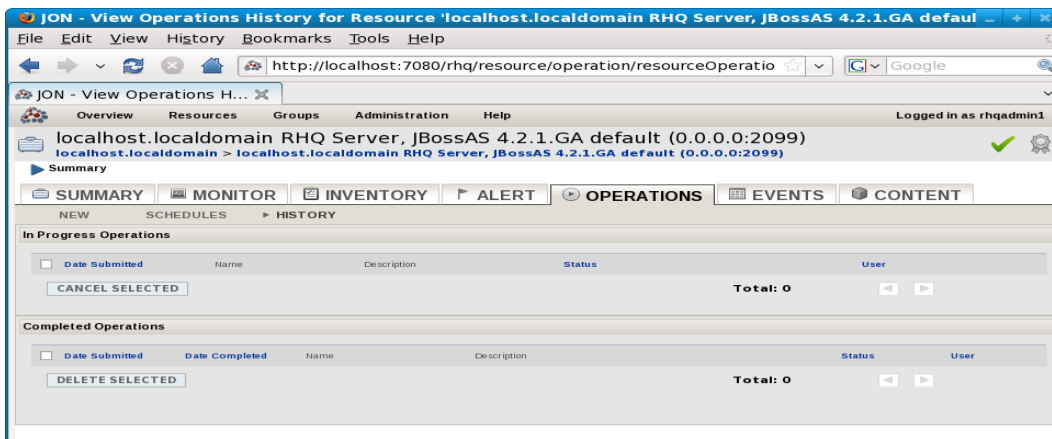
The "Schedules" sub tab shows any operations that are currently scheduled.
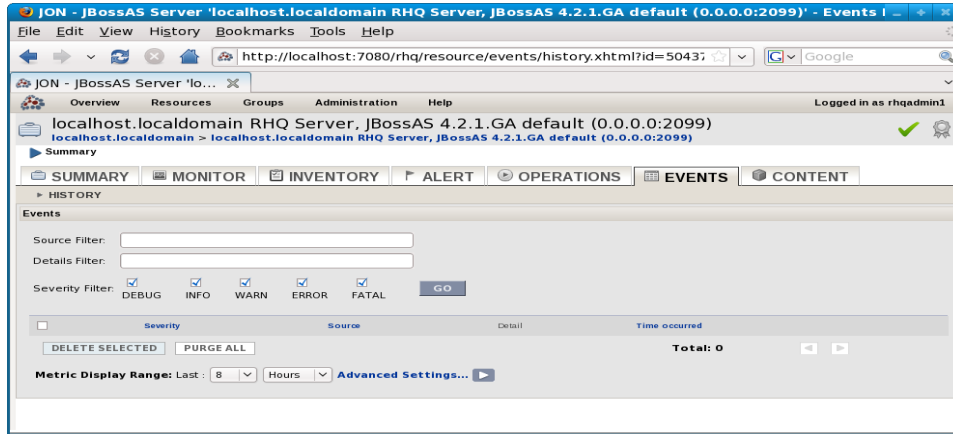

The History sub tab shows a history of completed and in progress operations as shown below.
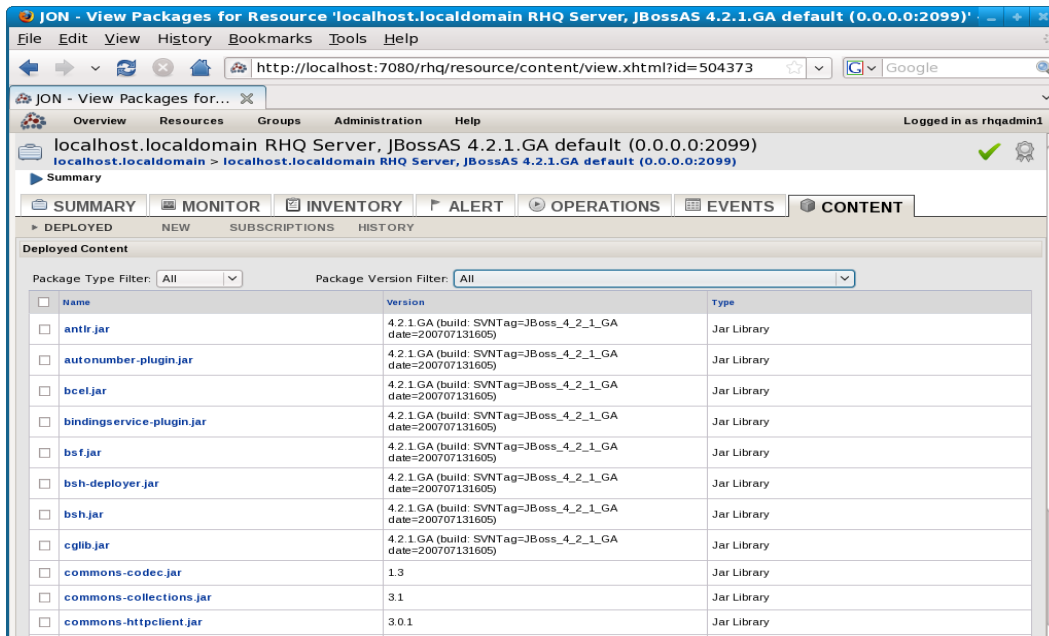
4.6 Events Tab

The Events tab shows events that have occurred for this resource, which can include application server log events if enabled.


4.7 Content Tab

The Content sub tab shows packages deployed to this resource. This is also the tab where a user could configure channels for downloading and applying patches to this resource.

4.8 Configure Tab

Some resources will also have a Configure tab, if they have attributes that can be configured. Below is an example of a configure tab for a JBoss datasource that can be configured.
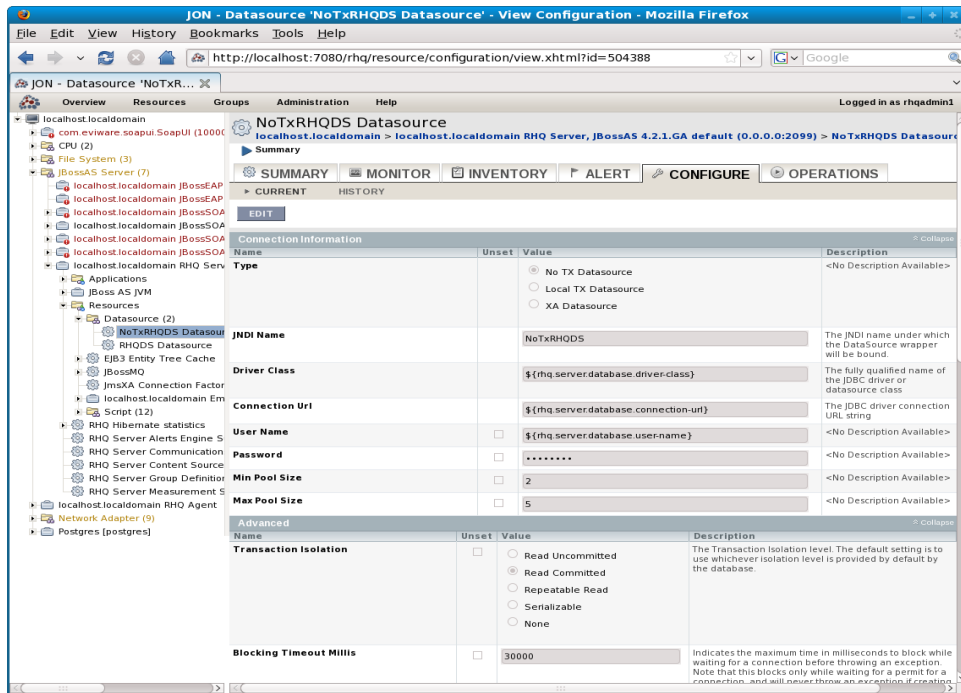
5 Groups

Groups are a mechanism by which multiple resources can be aggregated together.

Groups can be "Compatible" groups which mean they are groups of the same type of resource like a group of JBoss AS server resources. Compatible groups can provided aggregated metrics at the group level since each of the group members are of the same type.

Groups can also be "Mixed" groups which means groups of different types of resources. For example, one could create a group that has a JBoss AS Server, a Tomcat Server, and a Postgres database. Mixed groups can't display aggregated metrics because the resources in them are of different types that have different metrics.

Finally, groups can be dynamic – called Group Definitions. This can be either a compatible or mixed group, but it is slightly different than a normal group because the user does not assign resources to this group. Instead, the user defines a group definition which has a condition that is dynamically used to determine what resources belong in this group.

5.1 Creating Standard Group

To create a group, select "Groups | New Group" from the menu bar as shown below.
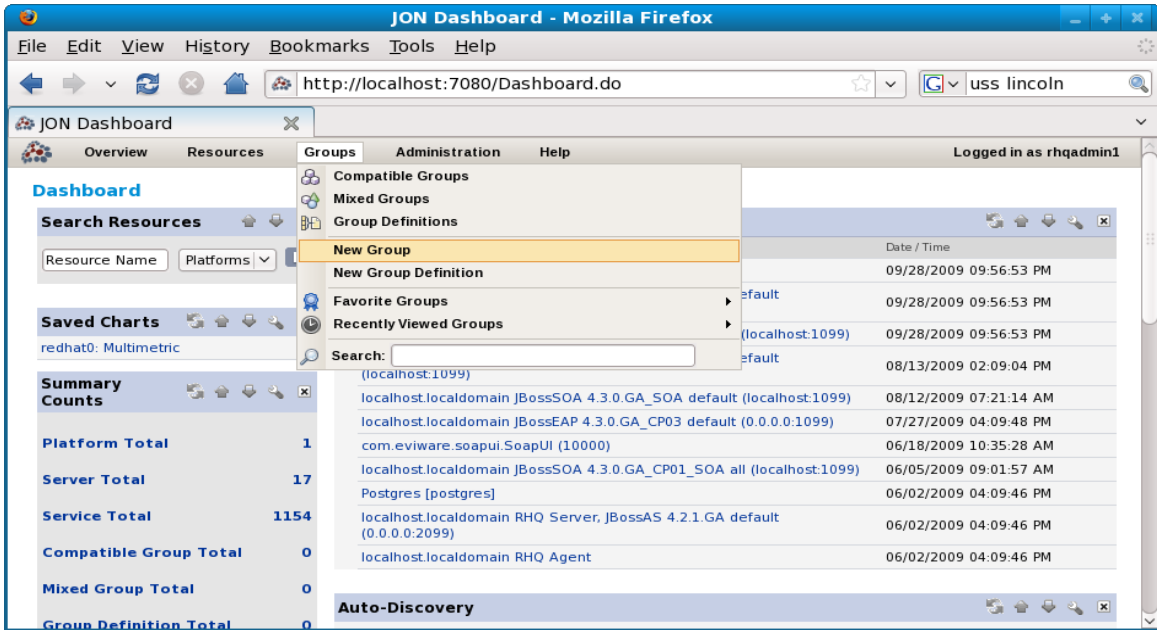
That will open the new group screen shown below. Note that the group can be compatible or mixed. If compatible, then a resource type must be selected. Also, the group can optionally be recursive, meaning it will include all child resources of the group resource members.
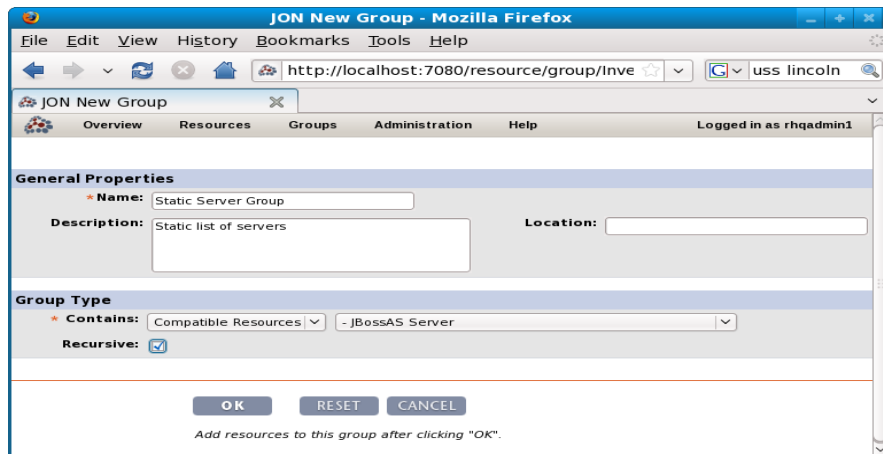
After clicking "Ok" a screen is presented that allows adding resources to this group, as shown below.
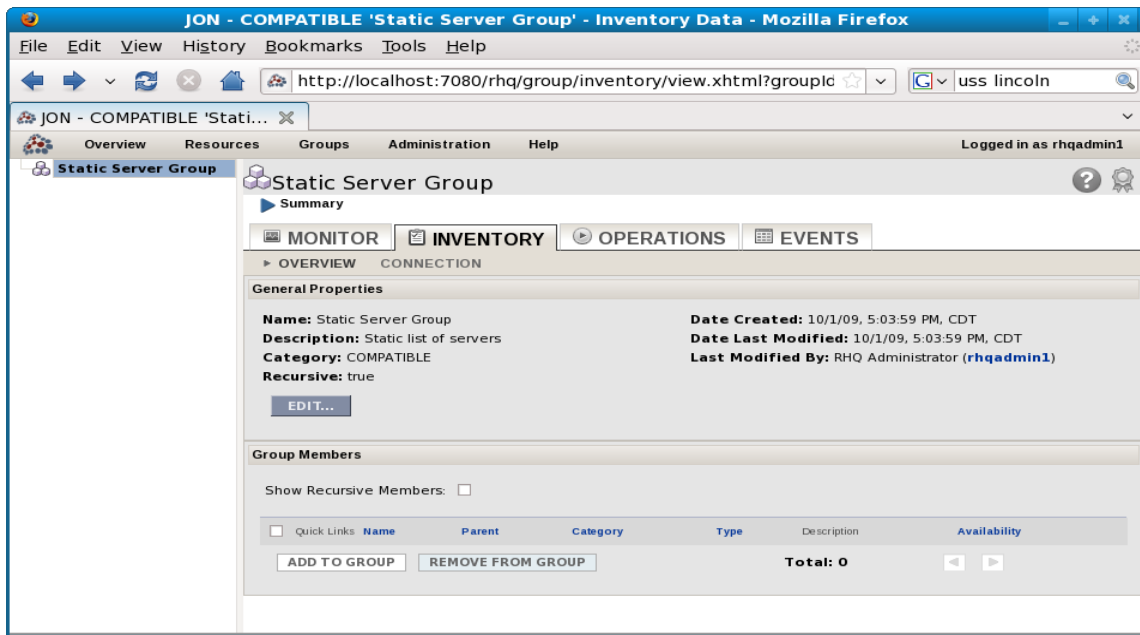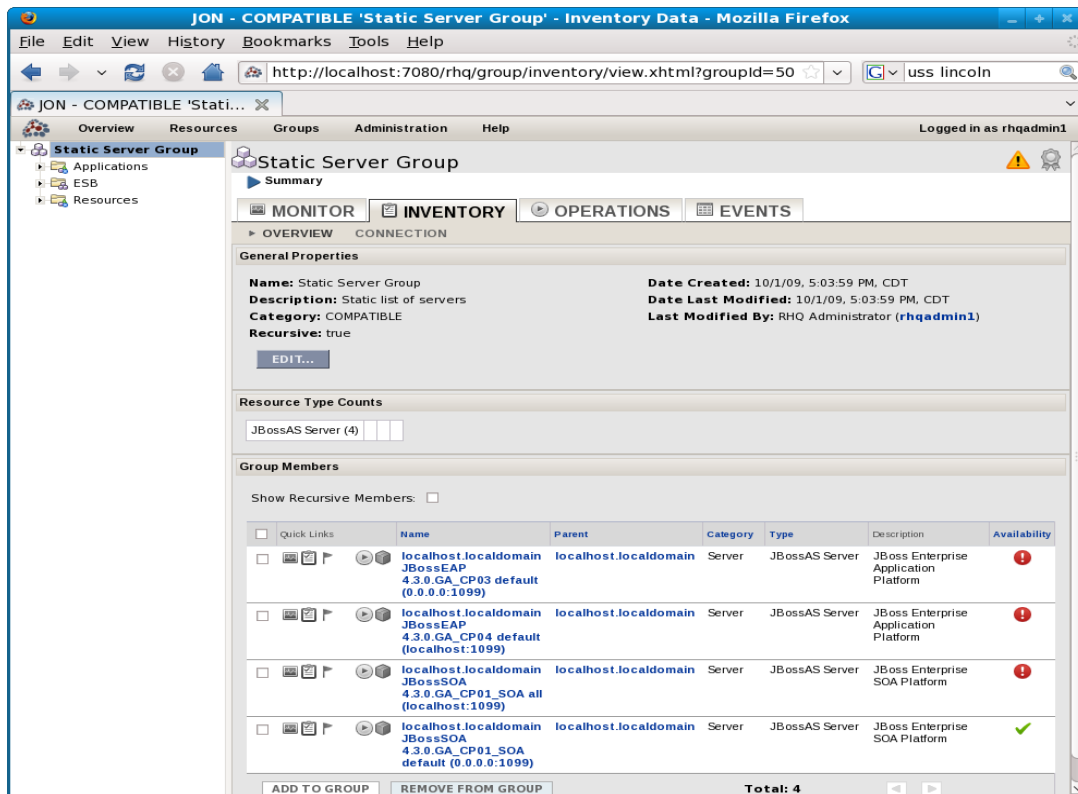
Clicking "Add to Group" opens the following screen where group members can be selected from.
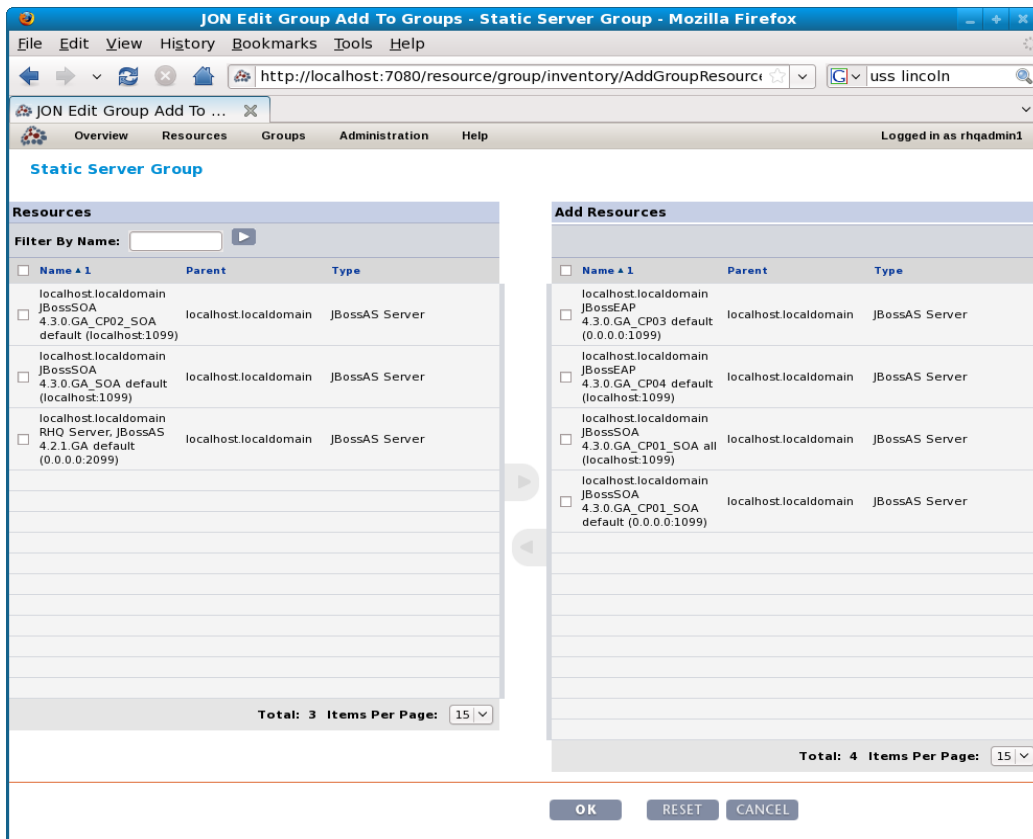
By selecting the desired resources from the list on the left and clicking the arrow to the right, the resources will be moved to the "Add Resources" list on the right.

Now, clicking the "Ok" button will display the group in the tree view.

This is the same tree view as was discussed earlier for general resources. The difference is that when looking at resources in a compatible group, the metrics are aggregated. So for example, the monitor metrics graphed below are aggregated from all members in the group.
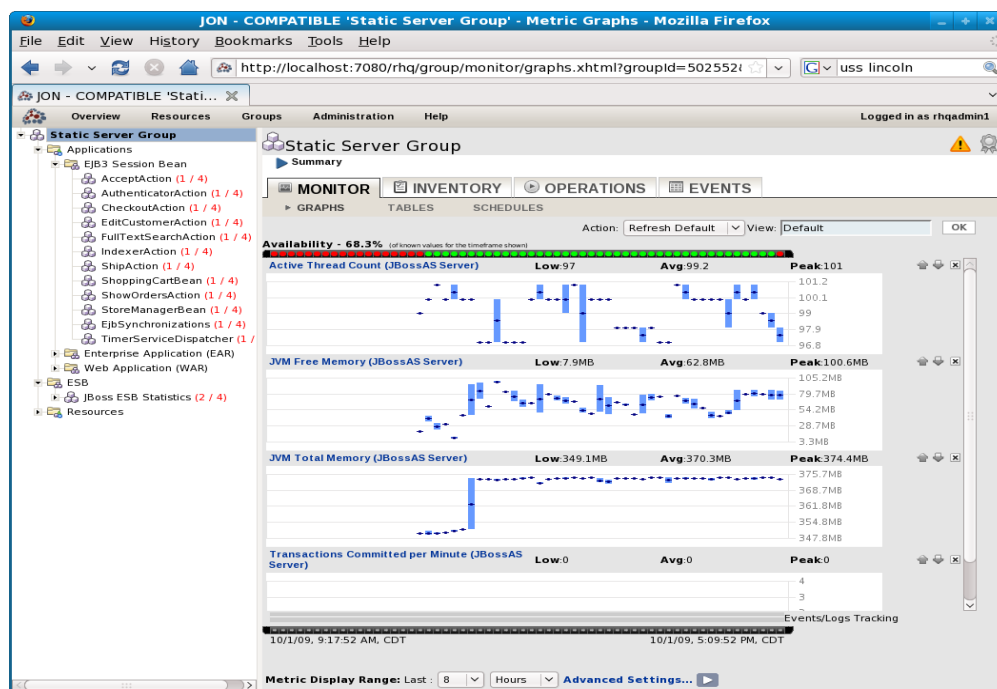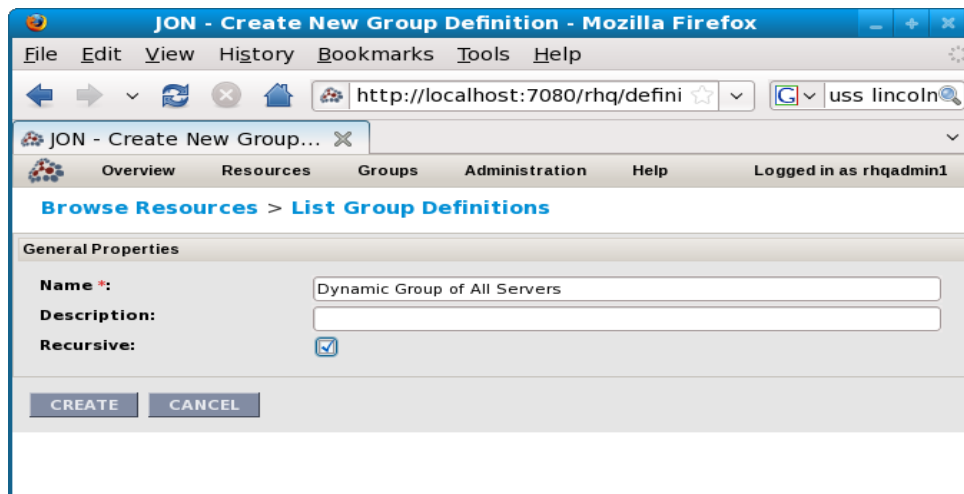
5.2 Creating Dynamic Groups (Group Definitions)

To create a dynamic group (Group Definition), select "Groups | New Group Definition" as shown below.
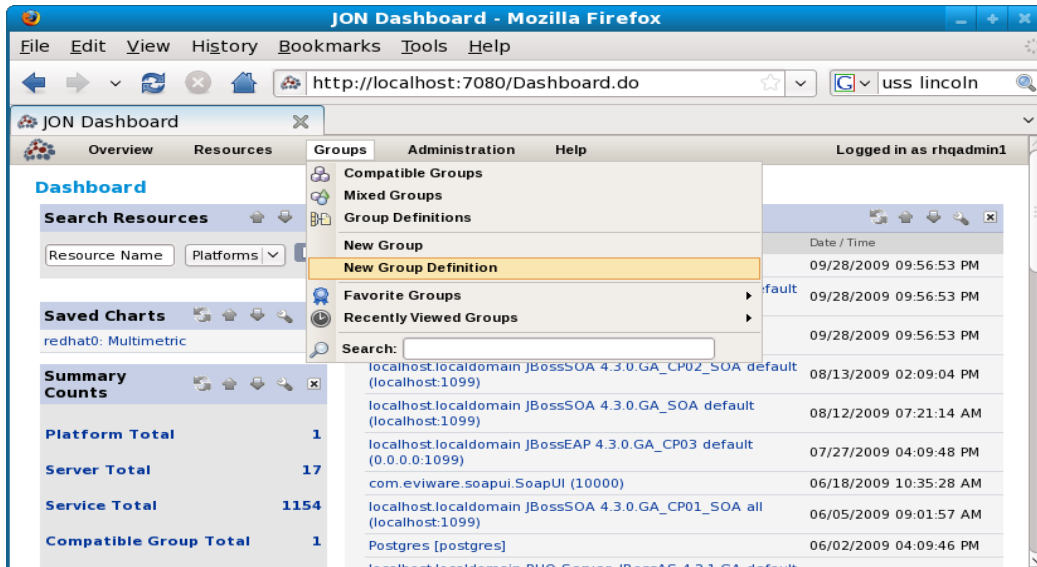

Similar to the regular groups, dynamic groups need a name and provide the option to include child resources recursively.

After selecting "Create", a screen is displayed allowing specification of the group definition. The mechanism for specifying which resources are in a group is very flexible. There are also some standard templates provided that create default conditions like below which will create a dynamic group for each server cluster (partition).
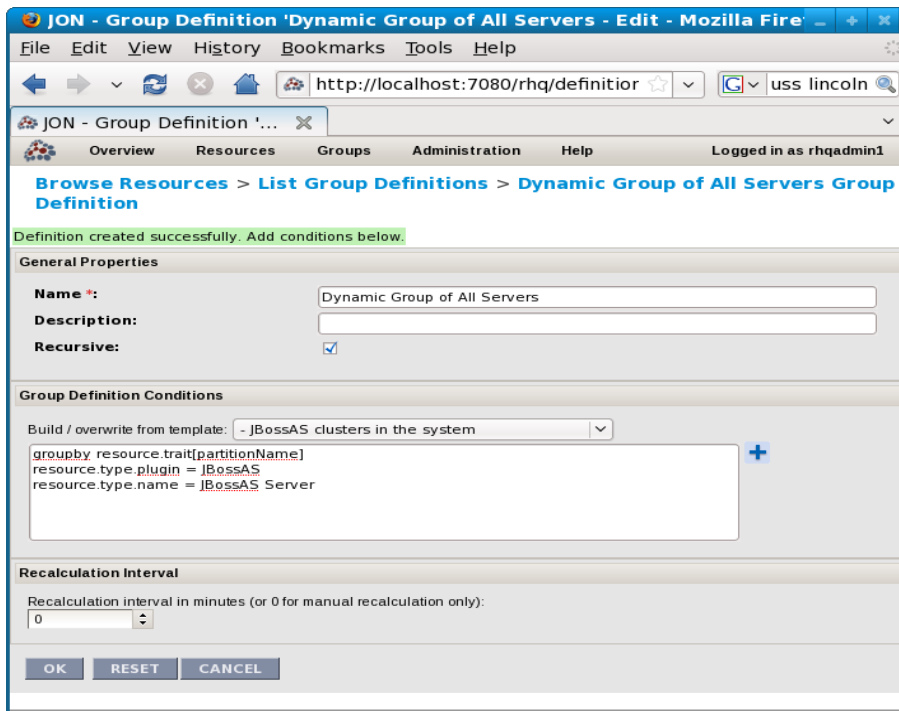
Clicking "OK" will display a page to allow calculating the members in the group. Click "Calculate Groups" to calculate the groups defined by this group definition.
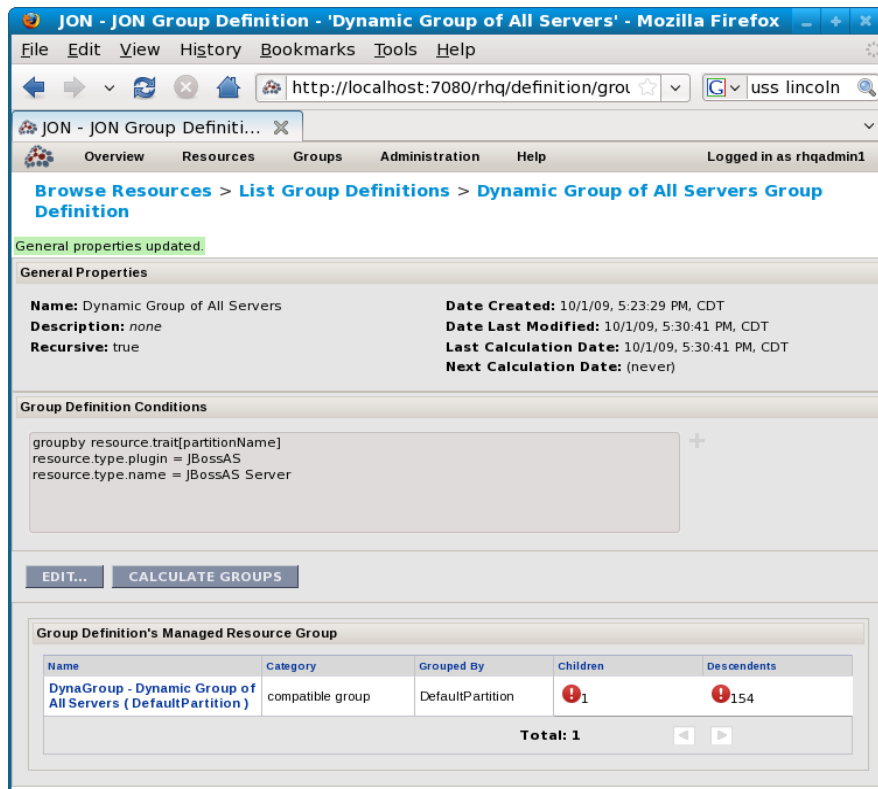
Now, by selecting "Groups | Compatible Groups", a screen like below will be displayed showing the

static and dynamic groups of JBoss servers that were just created. Clicking either of these groups will go back to the standard resource page where one can monitor, inventory, run operations, etc. as with any resource managed by JON.

6 Logout

To log out of JON, simply scroll the mouse over the "Logged in as <user>" message and select "Logout" as shown in the top right of this screenshot.
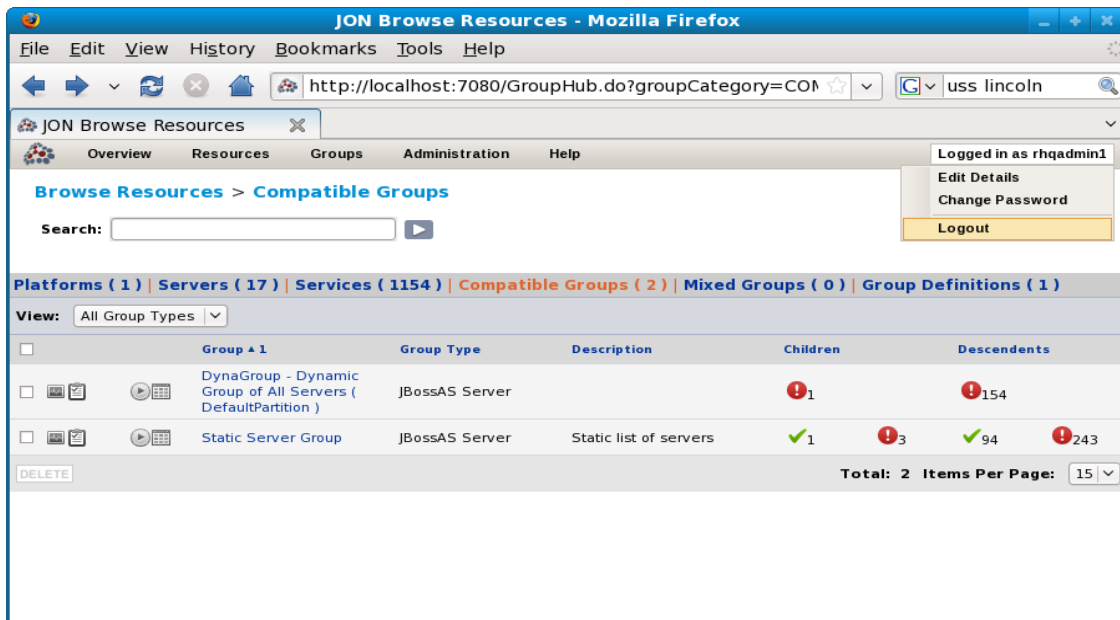
7 Summary

This document provides just a brief introduction to some of the features provided by JON. The main goal is to get the user somewhat familiar with the main pages and concepts of JON. For more detailed information, please consult the official JBoss Operations Network documentation.

http://www.redhat.com/docs/en-US/JBoss_ON/