4.2.2.4.3.3 x4ftl01 Host

See section 4.2.2.4.3.2, x4eil01 Host.

4.2.2.4.3.4 DataPool WebAccess GUI

The URL to the Data Pool WebAccess is http://<hostname>:<port>/WebAccess, an example is http://f4eil01.hitc.com:22500/WebAccess. To monitor the website, Hyperic provides a HTTP service that can be configured to periodically submit an http request to a specified URL. Hyperic will read the responses and determine if the website is available or not. Below are the steps to configure Hyperic to monitor the DataPool WebAccess GUI.

1. Select a platform where you want to define your HTTP service. The agent on this platform will be in charge of submitting a http request and reading the responses. Mouse over the 'Resources' tab and select the 'Browse' option. Click on the 'Platforms' link if it isn't highlighted and select a platform.

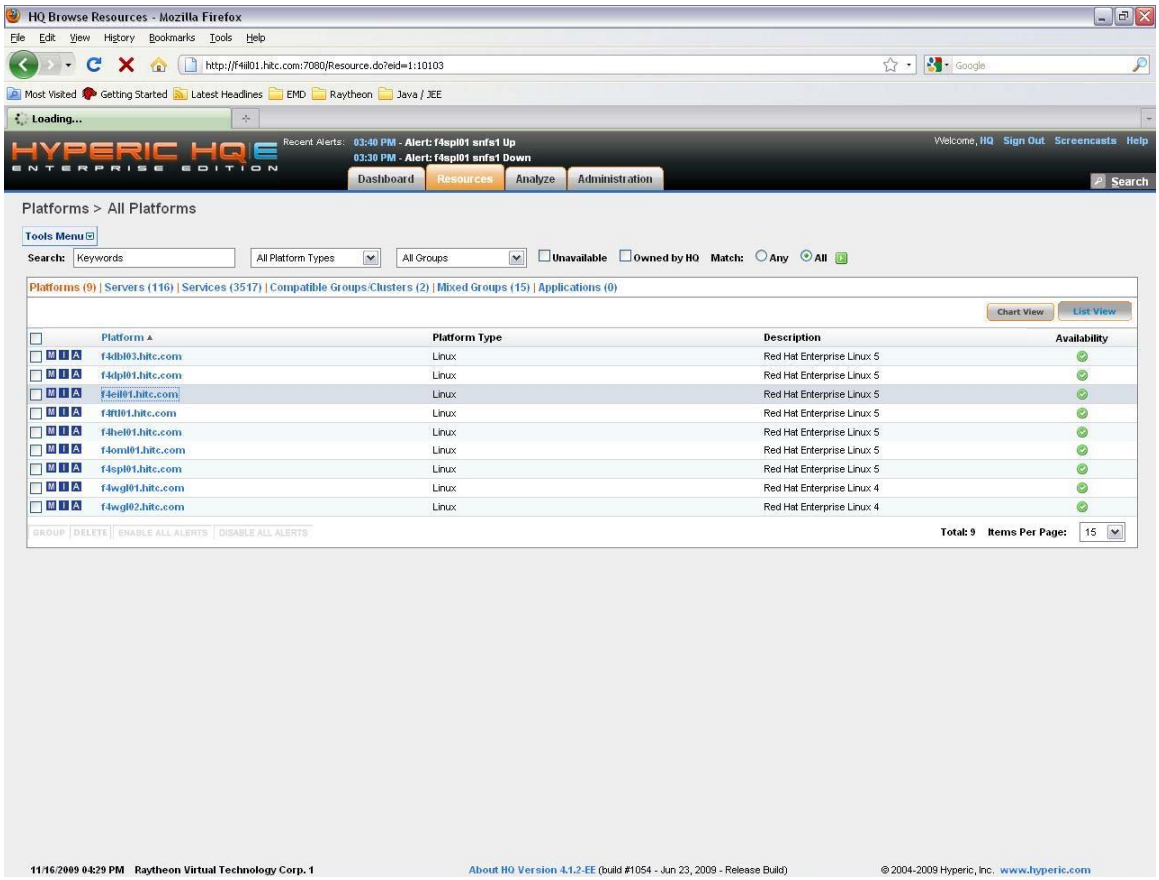

Figure 4.2.2.4.3-3. Select DPL WebAccess HTTP Service Platform

2. In the platform's detailed page, click on the 'Tools Menu' button and select the 'Add Platform Service' option.

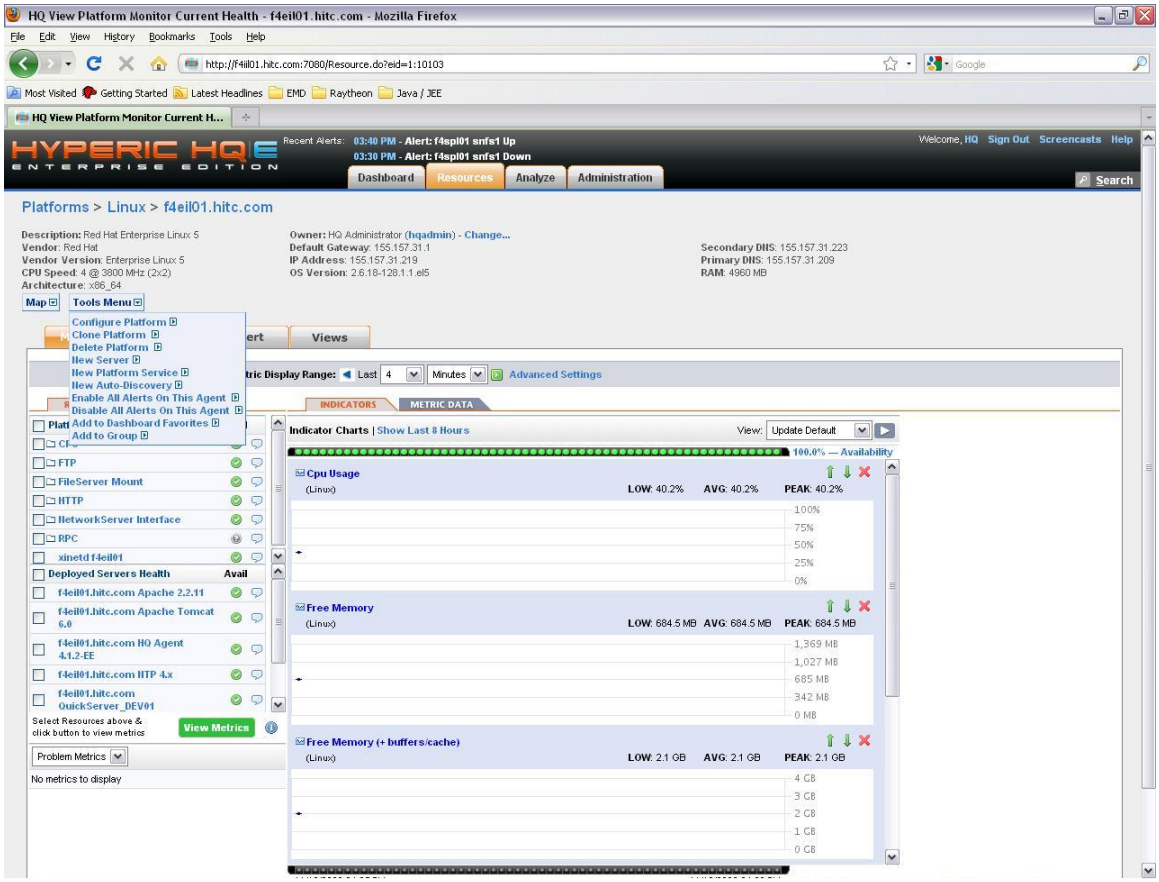

Figure 34.2.2.4.3-4. DPL WebAccess Add New Service

3. A new service general properties page will be displayed. Fill in the name, description, and service type. Afterward, click the 'OK' button.

   Name: WebAccess GUI OPS

   Description: This service monitors the WebAccess GUI in the OPS mode.

   Service Type: HTTP

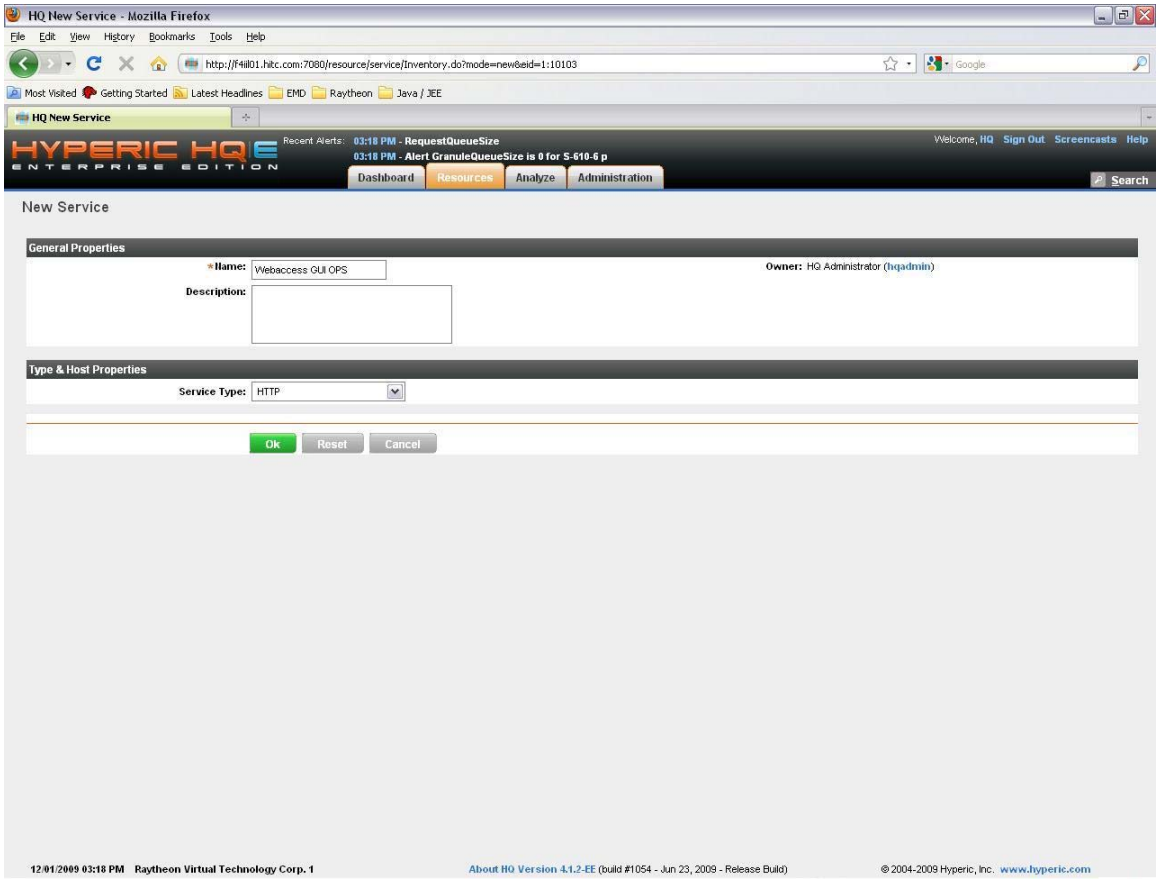

Figure 4.2.2.4.3-5. DPL WebAccess General Properties

4. You will be taken to the service detail page, but the service has not been configured to monitor the DPL WebAccess yet. Click on the 'Configuration Properties' link.

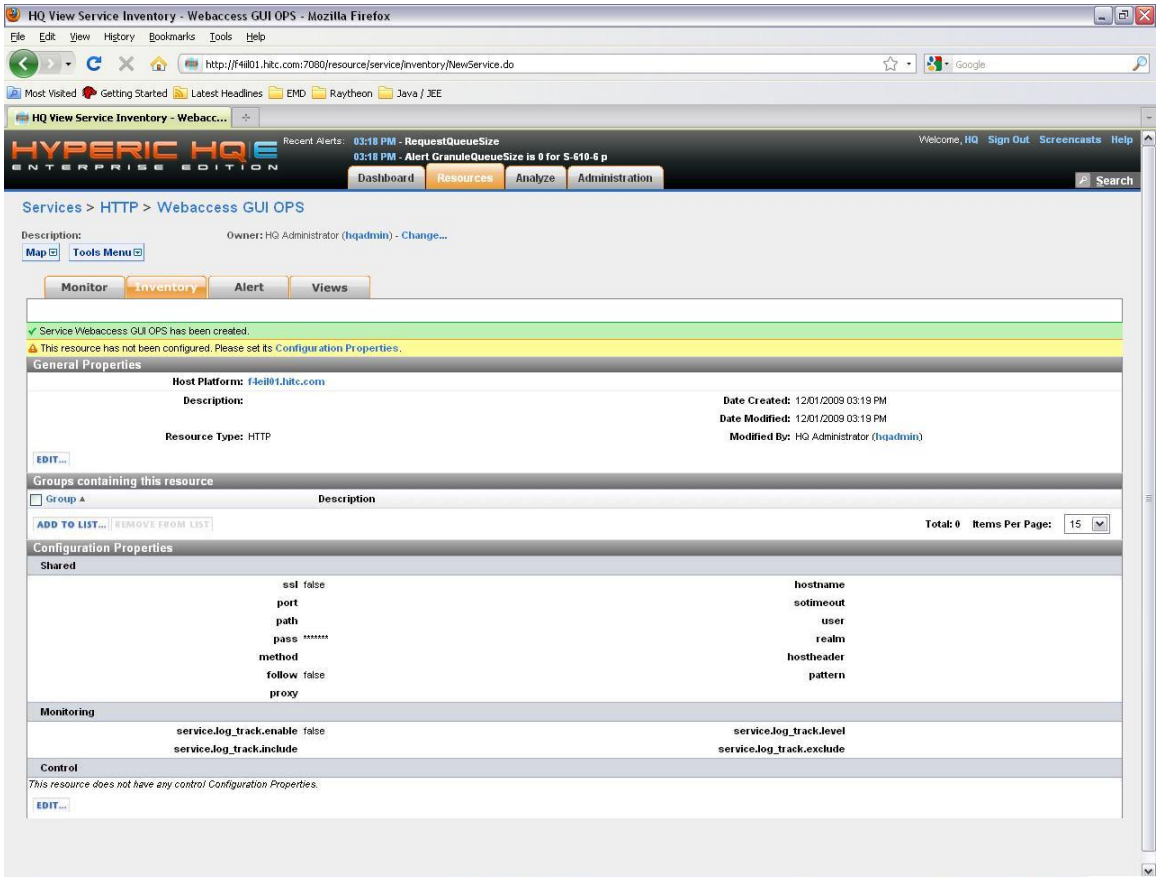

Figure 4.2.2.4.3-6. DPL WebAccess Configuration Properties Link

5. In the configuration properties page fill in the following fields (below is an example from EDF) and click the 'OK' button.

   hostname: f4eil01.hitc.com

   port: 22500

   ***sotimeout**: 60

   path: /WebAccess

   method: GET

   follow: check the box

   ****proxy**: <hostname>:<port>

   *This is the amount of time in seconds Hyperic will timeout from the http request.

   **If your internal network uses a proxy to get out supply it here.

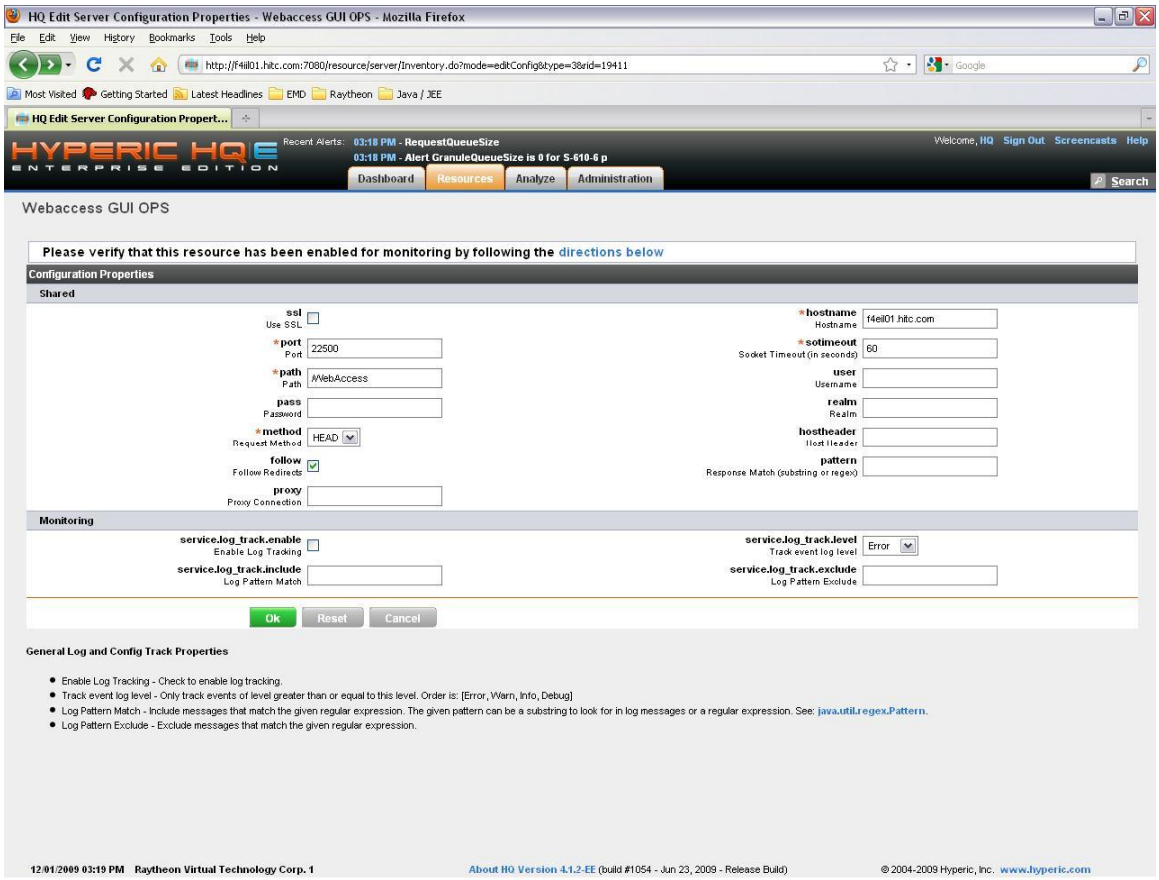

Figure 4.2.2.4.3-7. DPL WebAccess Configuration Properties Page

6. The DPL WebAccess HTTP service is now configured and you can go ahead and monitor it, setup alerts, and change the monitoring interval of the metrics it gathers.

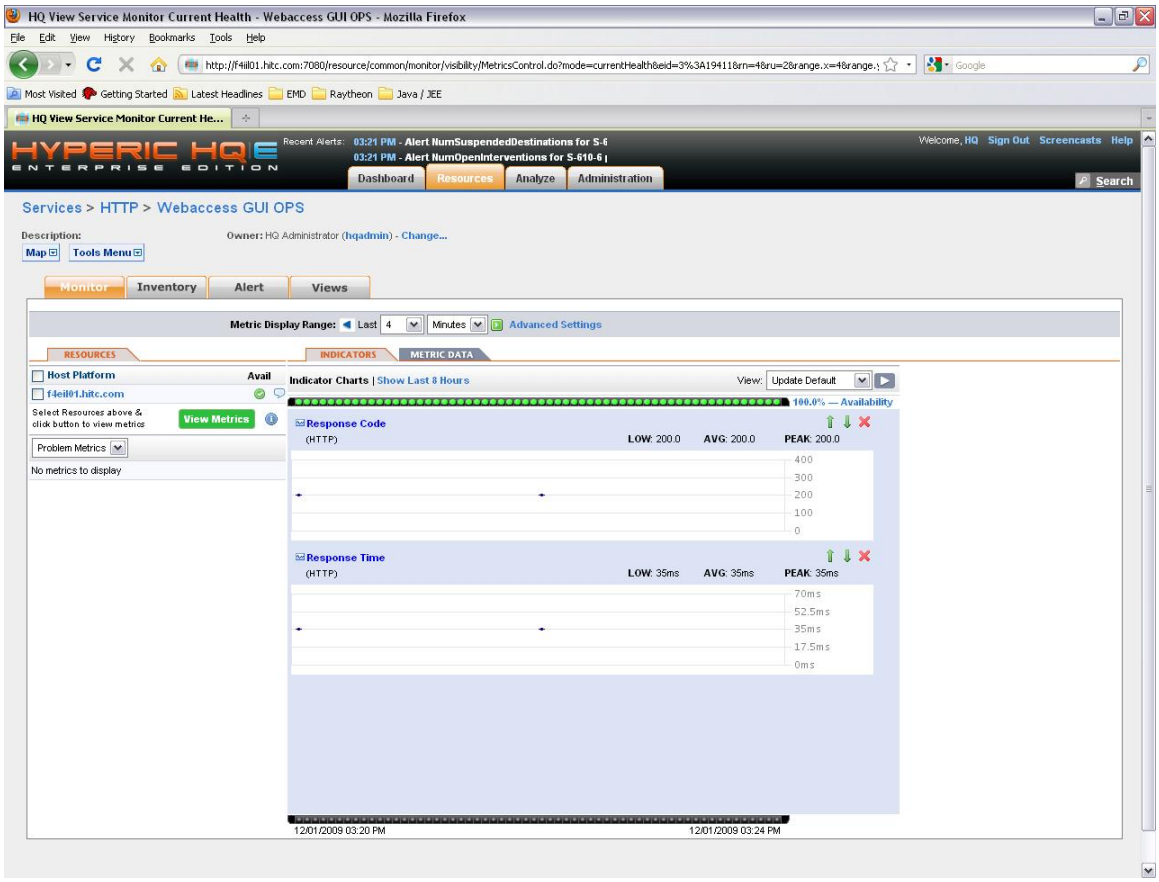

Figure 4.2.2.4.3-8. DPL WebAccess Monitoring

4.2.2.4.3.5 Web Order Status GUI

The URL to the Web Order Status GUI is http://<hostname>:<port>/OrderStatus <_MODE>, an example is http://f4eil01.hitc.com:22500/OrderStatus. To monitor the website, Hyperic provides a HTTP service that can be configured to periodically submit an http request to a specified URL. Hyperic will read the responses and determine if the website is available or not. Below are the steps to configure Hyperic to monitor the Web Order Status GUI.

1. Select a platform where you want to define your HTTP service. The agent on this platform will be in charge of submitting a http request and reading the responses. Mouse over the 'Resources' tab and select the 'Browse' option. Click on the 'Platforms' link if it isn't highlighted and select a platform.

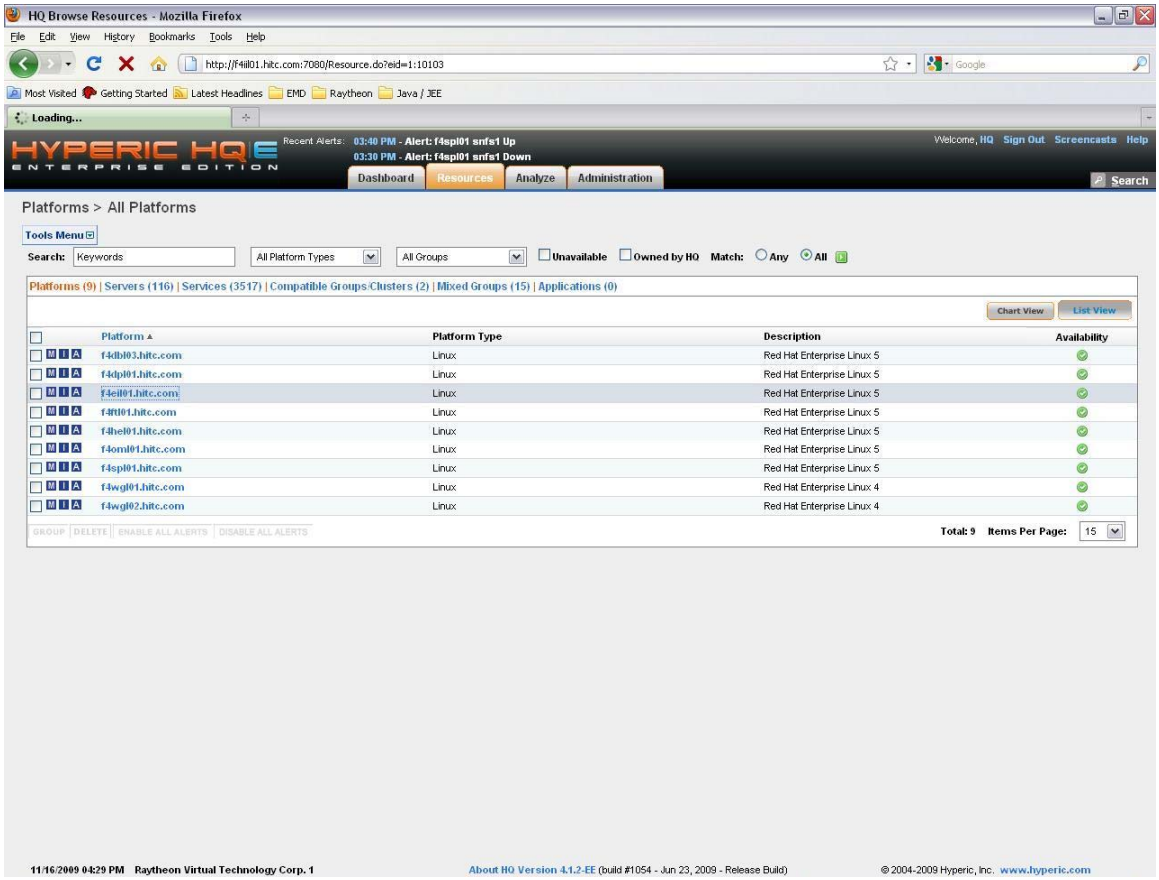

Figure 4.2.2.4.3-9. Select Web Order Status GUI HTTP Service Platform

2. In the platform's detailed page, click on the 'Tools Menu' button and select the 'Add Platform Service' option.

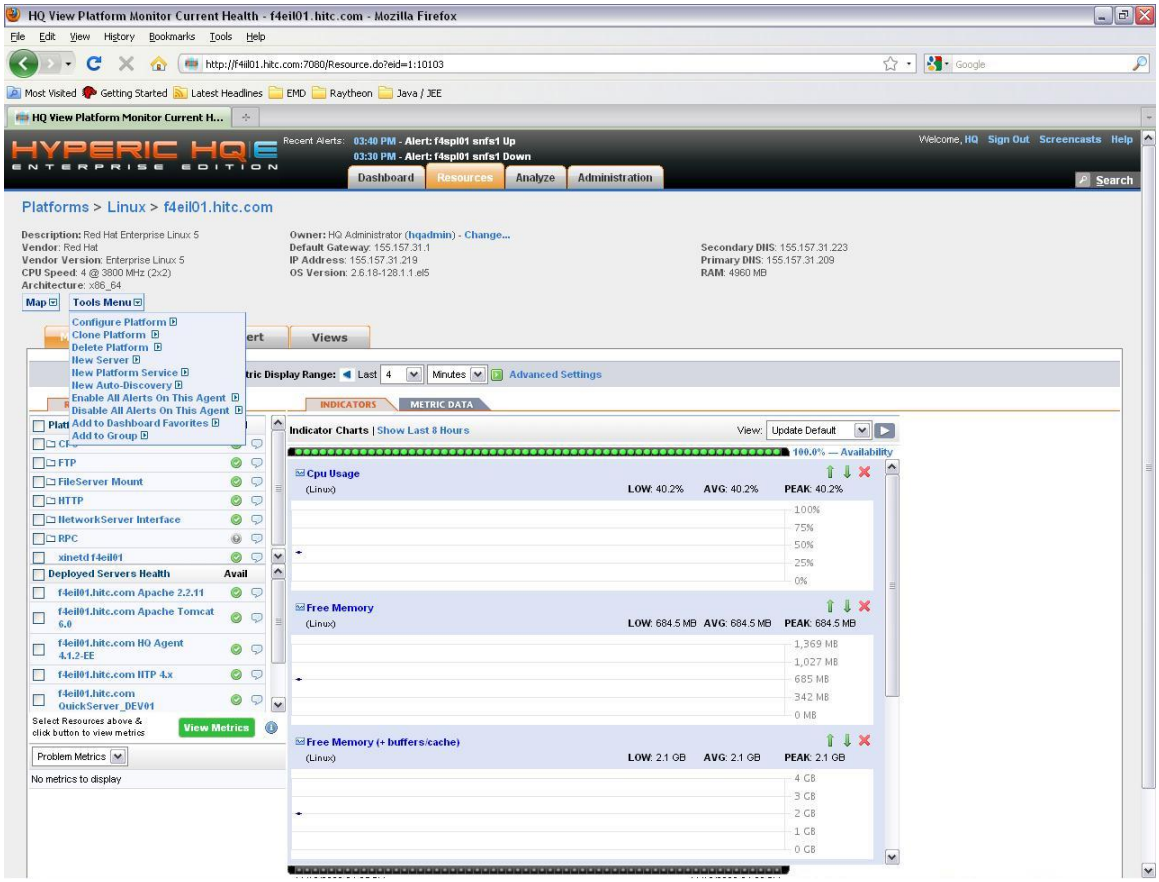

Figure 4.2.2.4.3-10. Web Order Status GUI Add New Service

3. A new service general properties page will be displayed. Fill in the name, description, and service type. Afterward, click the 'OK' button.

   Name: OrderStatus GUI OPS

   Description: This service monitors the OrderStatus GUI in the OPS mode.

   Service Type: HTTP

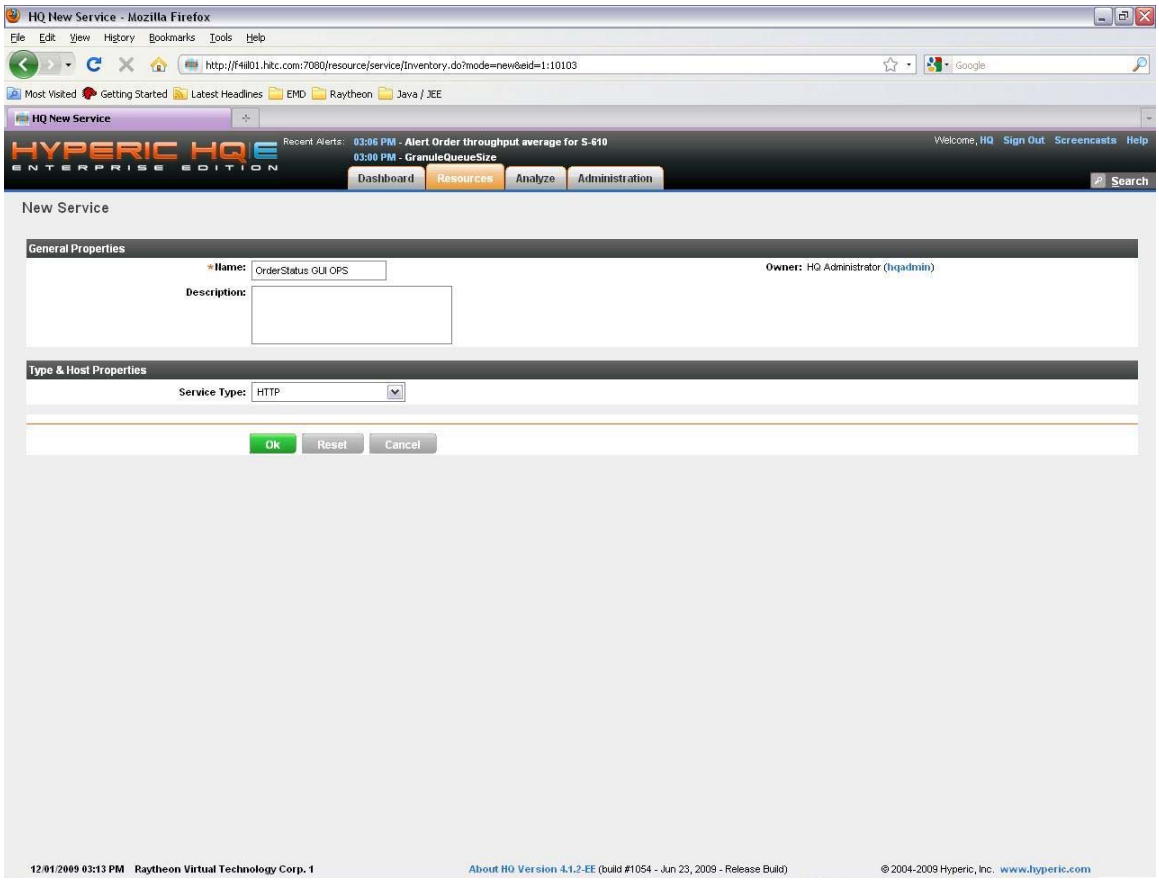

Figure 4.2.2.4.3-11. Web Order Status GUI General Properties

4. You will be taken to the service detail page, but the service has not been configured to monitor the Web Order Status GUI yet. Click on the 'Configuration Properties' link.

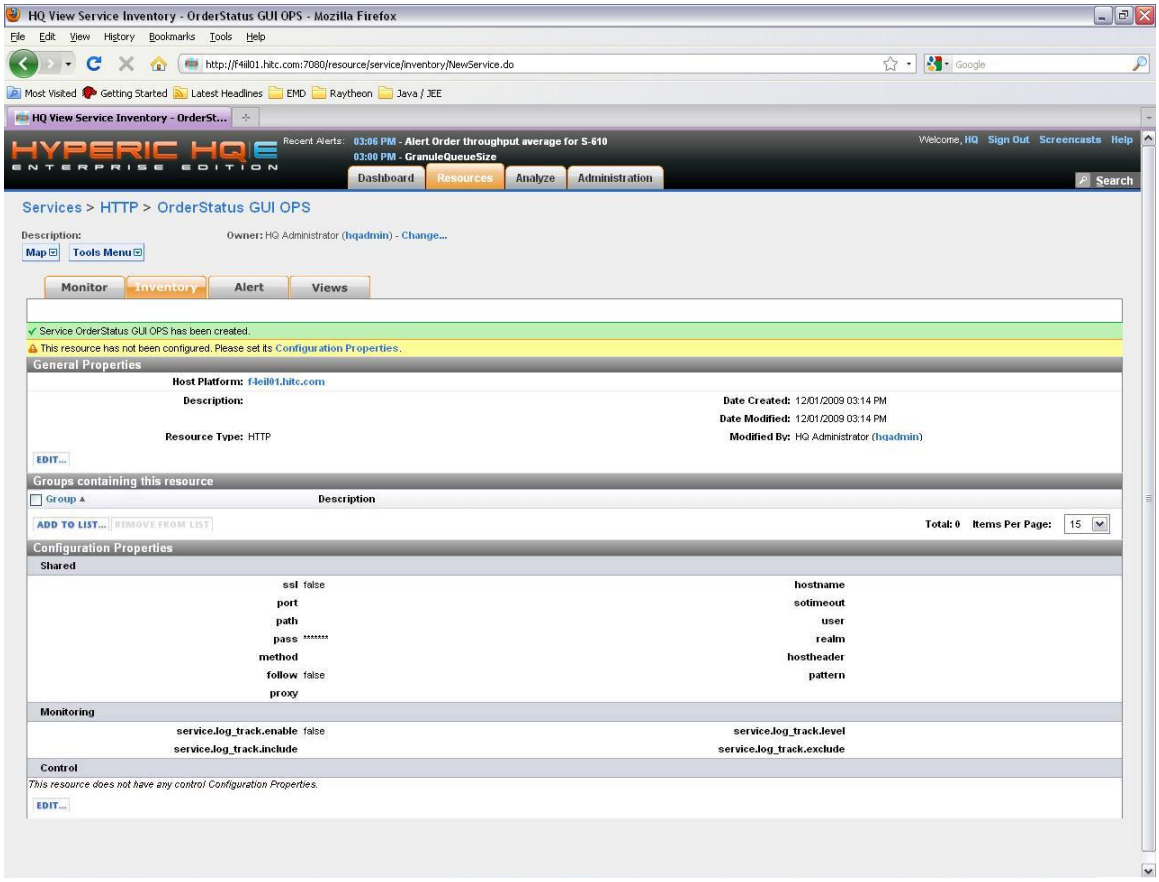

Figure 4.2.2.4.3-12. Web Order Status GUI Configuration Properties Link

5. In the configuration properties page fill in the following fields (below is an example from EDF) and click the 'OK' button.

   hostname: f4eil01.hitc.com

   port: 22500

   ***sotimeout**: 60

   path: /OrderStatus

   method: GET

   follow: check the box

   ****proxy**: <hostname>:<port>

   *This is the amount of time in seconds Hyperic will timeout from the http request.

   **If your internal network uses a proxy to get out supply it here.

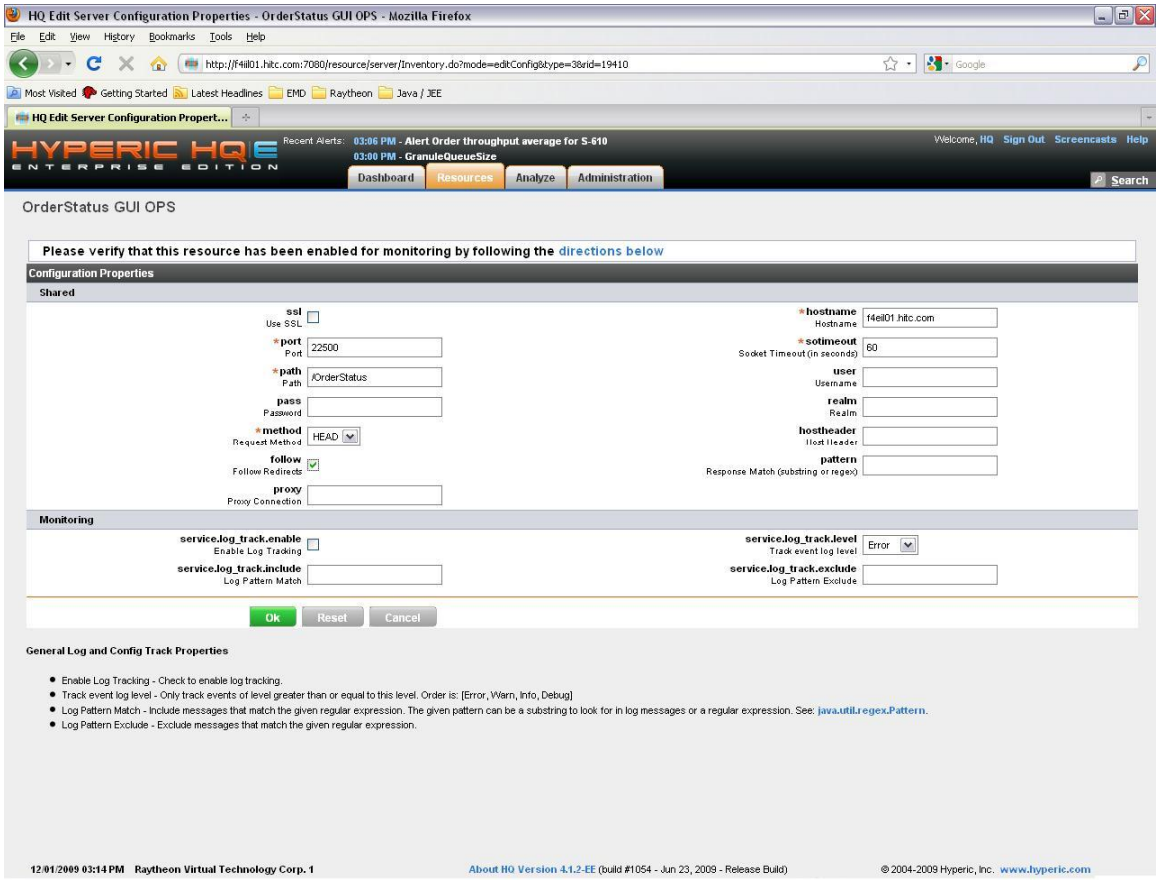

Figure 4.2.2.4.3-13. Web Order Status GUI Configuration Properties Page

6. The Web Order Status GUI HTTP service is now configured and you can go ahead and monitor it, setup alerts, and change the monitoring interval of the metrics it gathers.

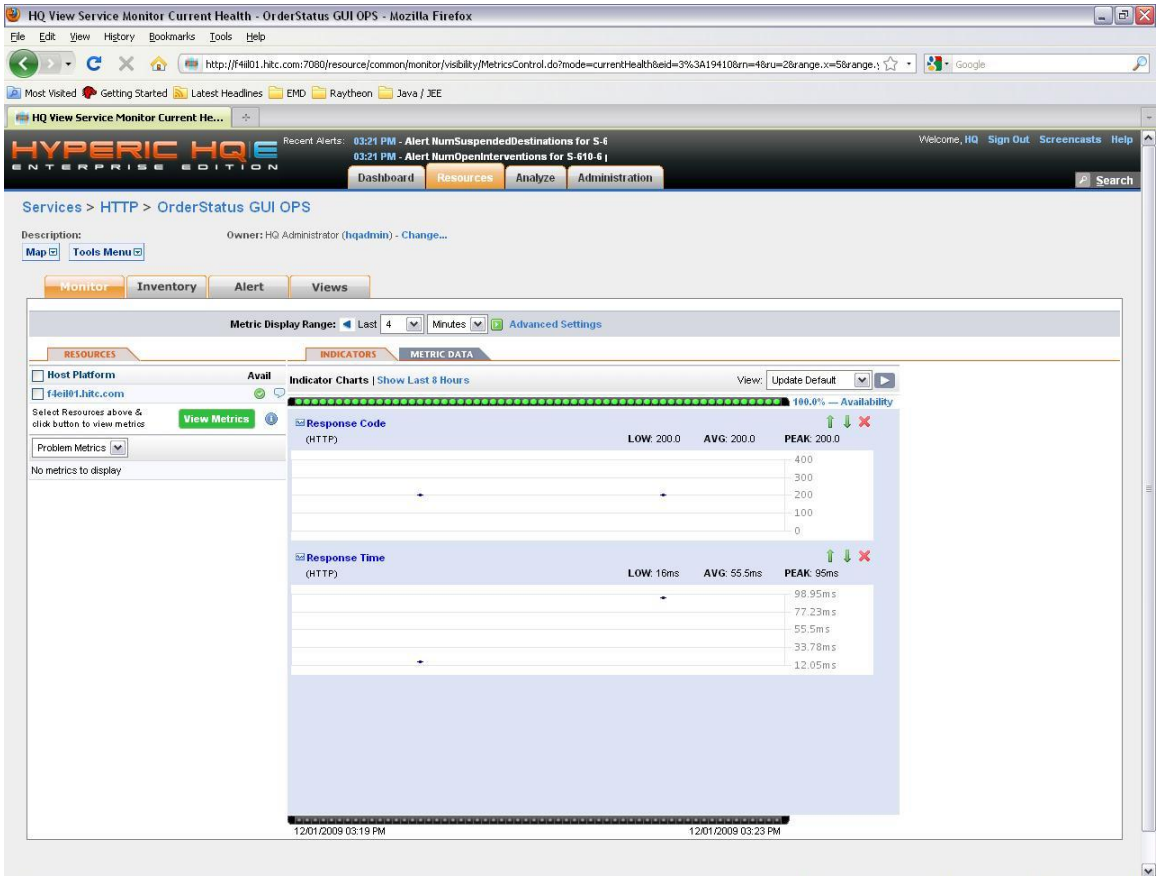

Figure 4.2.2.4.3-14. Web Order Status GUI Monitoring

4.2.2.4.3.6 DataPool File System

See section 4.2.2.4.2.9, DataPool File System.

4.2.2.4.3.7 Sybase

See section 4.2.2.3.1, Configuring Sybase.

4.2.2.4.3.8 SQS Server

See section 4.2.2.4.2.12, it show how to configure Hyperic to monitor each SQS Server.

4.2.2.4.3.9 Apache / Tomcat Web Server

See section 4.2.2.3.2, Configuring Tomcat.

4.2.2.4.3.10 Wu-FTP Server

The Wu-FTP COTS Server runs on the x4eil0x and x4ftl0x hosts servicing ftp requests. A custom server, FtpDataCollector was developed to poll the Wu-FTP log for completed request and the information captured is stored in the Sybase database. In order to monitor the Wu-FTP Server a FTP service is configured to periodically ping the server to determine if it is availability, its response time, and to gather ftp protocol specific metrics. A custom Hyperic plugin was developed to monitor the FtpDataCollector server and to retrieve the information stored in the Sybase database for display in the Hyperic GUI.

4.2.2.4.3.10.1 Configure FTP Service

Please refer to the "Configure FTP" section. The section provides step-by-step instructions on how to configure an ftp service to monitor a host with an ftp server running.

4.2.2.4.3.10.2 FtpDataCollector Server

The EcMsSmFtpDataCollector Server monitoring is done through an xml custom plug-in. There is one plug-in file per host. The plug-in file name is FtpDataCollector-plugin.xml.

The metrics that will be collected for the FtpDataCollector Server includes the standard Hyperic metrics (see section 4.2.2.2.13, Standard Plugin Metrics) and custom metrics (see below).

Table 4.2.2.4.3-1. FtpDataCollector Metrics

Custom Metrics	
Metric Name	**Description**
Number of Ftp Sessions	The total number of ftp sessions for the host.
Number of Ftp Transfers	The total number of ftp transfers over the last five minutes for the host.
Ftp Transfer Rate	The average ftp volume (in MB) throughput per minute over the last five minutes for the host.
Number of Http Transfers	The total number of http transfers over the last five minutes for the host.
Http Transfer Rate	The average http volume (in MB) throughput per minute over the last five minutes for the host.

The FtpDataCollector Server resource will be auto-discovered by Hyperic using the PTQL query defined in the xml plug-in. The standard metrics will be collected by the SIGAR plug-in and the custom metrics will be collected through the SQL plug-in by querying the Ingest database.

Up/down alerts are set up for the FtpDataCollector Server based on its availability metrics. The down alert is used to notify operator of the problem and update the Data Access business process status and trigger the control action that will automatically start the server; the up alert is used as a recovery alert for the down alert to mark the down alert as fixed.

Please refer to screen shots in 7.3 Polling Server for details on the control actions and alerts configurations. The configuration process is basically identical except the resource type, control action, script names and alert names are changed accordingly.

4.2.2.4.4 Order Management Business Process

This section describes the resources that are grouped within the DPL Ingest Business Process.

4.2.2.4.4.1 Order Management Resources

Hosts

- x4oml01

Custom Code

- Order Manager Server
- HEG Server
- EPD Server
- Copy Server
- EWOC
- WIST

File Systems

- Data Pool

Database

- Sybase

HSA

4.2.2.4.4.2 Order Manager Host

A Hyperic agent will be installed on each host used by Order Manager. Each host will be configured to be monitored via Hyperic's auto-discovery service. Once the host is selected to be monitored there are a number of metrics that can be captured including availability, load average, CPU usage, TCP statistics, memory usage, and swap space.

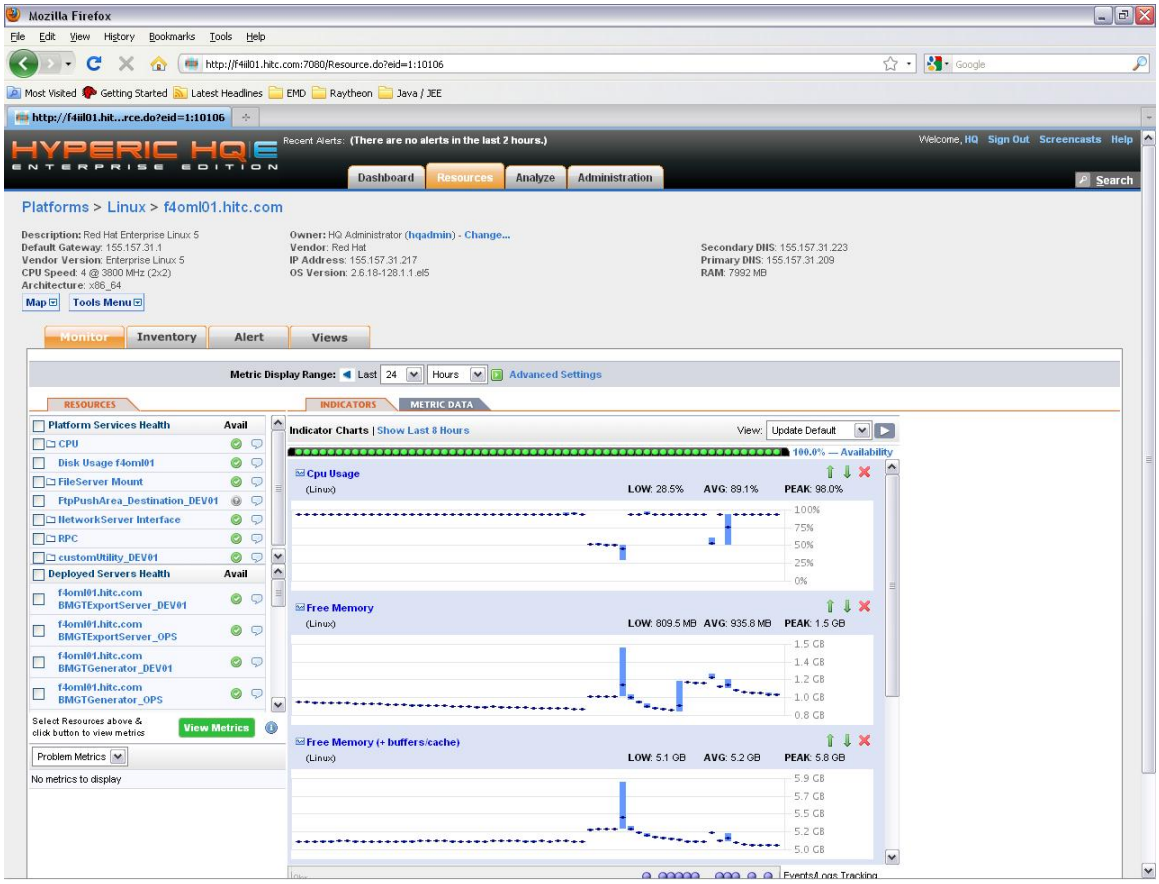

Figure 4.2.2.4.4-1. Order Manager Host Monitor Page

4.2.2.4.4.3 Order Manager Server

The Order Manager Server monitoring is done through an xml custom plug-in. There is one plug-in file per mode. The plug-in file name is <MODE>-OMS-OrderManager-plugin.xml.

The metrics that will be collected for the Order Manager Server include the standard Hyperic metrics (see section 4.2.2.2.13, Standard Plugin Metrics) and custom metrics (see below).

Table 4.2.2.4.4-1. Order Manager Server Metrics (1 of 2)

Custom Metrics	
Metric Name	**Description**
RequestQueueSize	The total number of requests queued in the system.
GranuleQueueSize	The total number of granules queued in the system.

Table 4.2.2.4.4-1. Order Manager Server Metrics (2 of 2)

Custom Metrics	
Metric Name	**Description**
NumOpenAlerts	The total number of open alerts in the server.
NumOpenInterventions	The total number of open interventions in the server.
NumSuspendedDestinations	The total number of suspended destinations in the server
Throughput (gran/min)	The average number of granules throughput per minute over the last five minutes.
Throughput (MB/min)	The average volume (in MB) throughput per minute over the last five minutes.
FtpPushThroughput (gran/min)	The average number of granules throughput per minute over the last five minutes for ftp push orders.
FtpPushThroughput (MB/min)	The average volume (in MB) throughput per minute over the last five minutes for ftp push orders.
FtpPullThroughput (gran/min)	The average number of granules throughput per minute over the last five minutes for ftp pull orders.
FtpPullThroughput (MB/min)	The average volume (in MB) throughput per minute over the last five minutes for ftp pull orders.

The Order Manager server resource will be auto-discovered by Hyperic using the PTQL query defined in the xml plug-in. The standard metrics will be collected by the SIGAR plug-in and the custom metrics will be collected through the SQL plug-in by querying the OMS database.

Up/down alerts are set up for the Order Manager server based on its availability metrics. The down alert is used to notify operator of the problem and update the Order Manager business process status and trigger the control action that will automatically start the server; the up alert is used as a recovery alert for the down alert to mark the down alert as fixed.

Please refer to screen shots in 7.3 Polling Server for details on the control actions and alerts configurations. The configuration process is basically identical except the resource type, control action, script names and alert names are changed accordingly.

4.2.2.4.4.4 OM Destination

Om Destination monitoring is done using the built-in Hyperic group and service metrics monitoring capability through a custom xml plug-in.

We collect the following metrics for each destination:

- Throughput (granules/minutes over the previous five minutes)
- Throughput (MB/minutes over the previous five minutes)

All destinations in a mode are displayed as a group called “<MODE>_DESTINATIONS”. The <MODE>_DESTINATIONS group can be added to the “Favorite Resources” portlet on the Dashboard for easy access. See screen shot below:

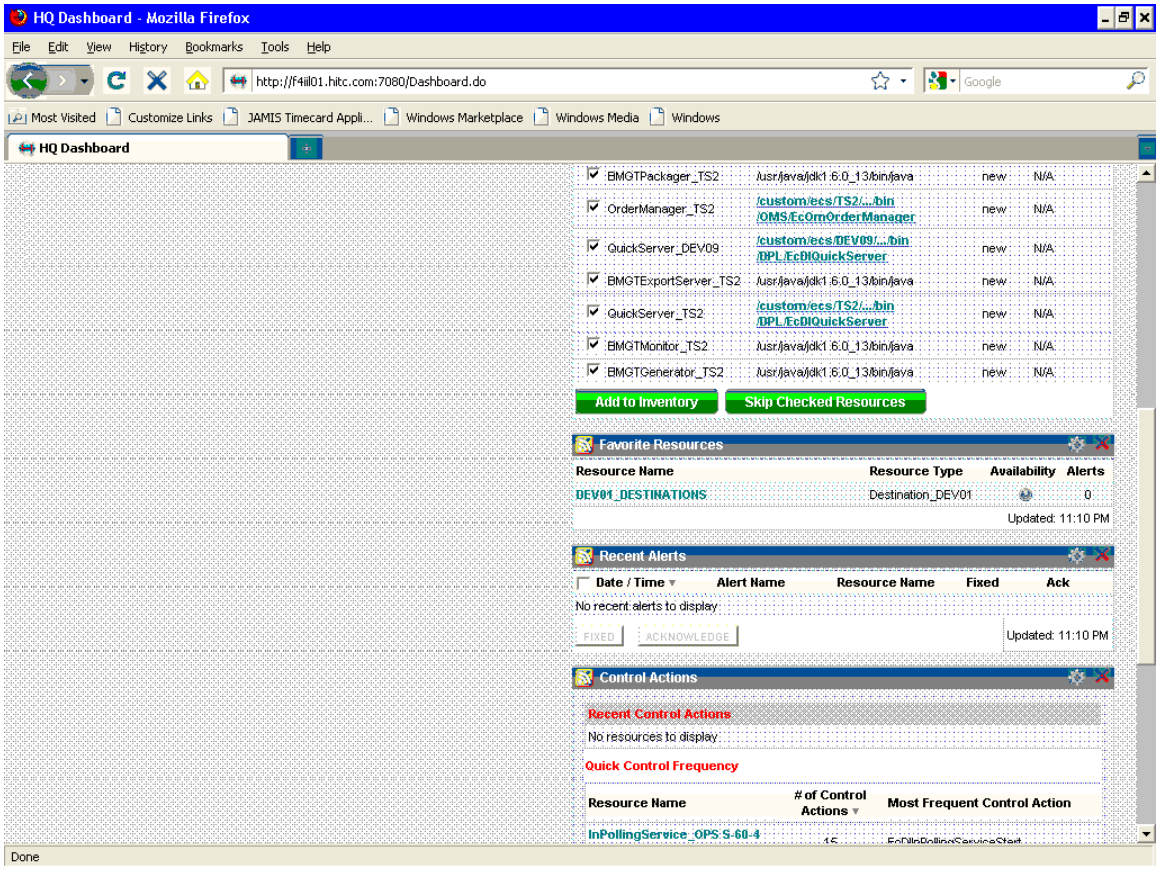

Figure 4.2.2.4.4-2. Destination Dashboard

Click on the OPS_DESTINATIONS will bring up the providers details page which lists the list of defined destinations in the OPS mode and the combined statistic metrics of all the destinations. See screen shot below:

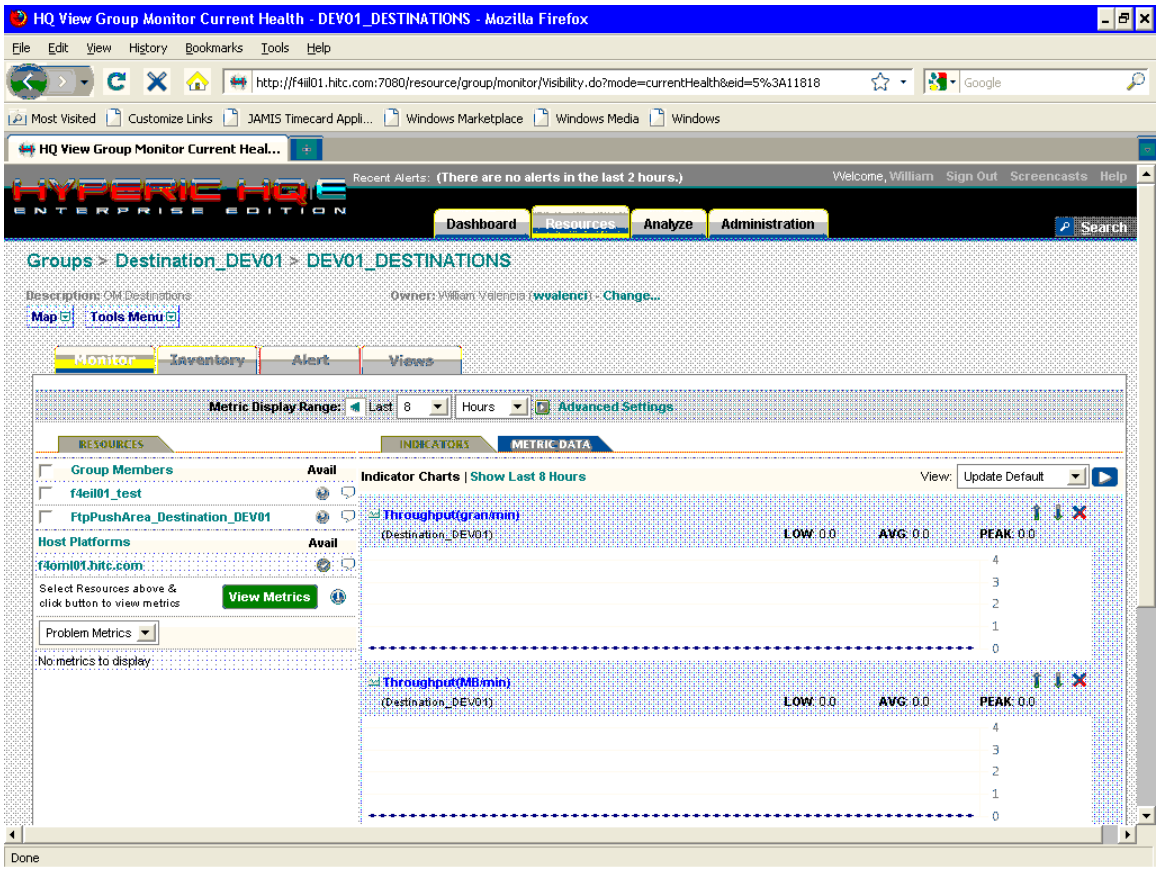

Figure 4.2.2.4.4-3. Data Provider Group Monitor Page

To see the metrics of an individual destination, click on the destination name in the “Resources” window on the left. See below for a screen shot of the destination detail page.

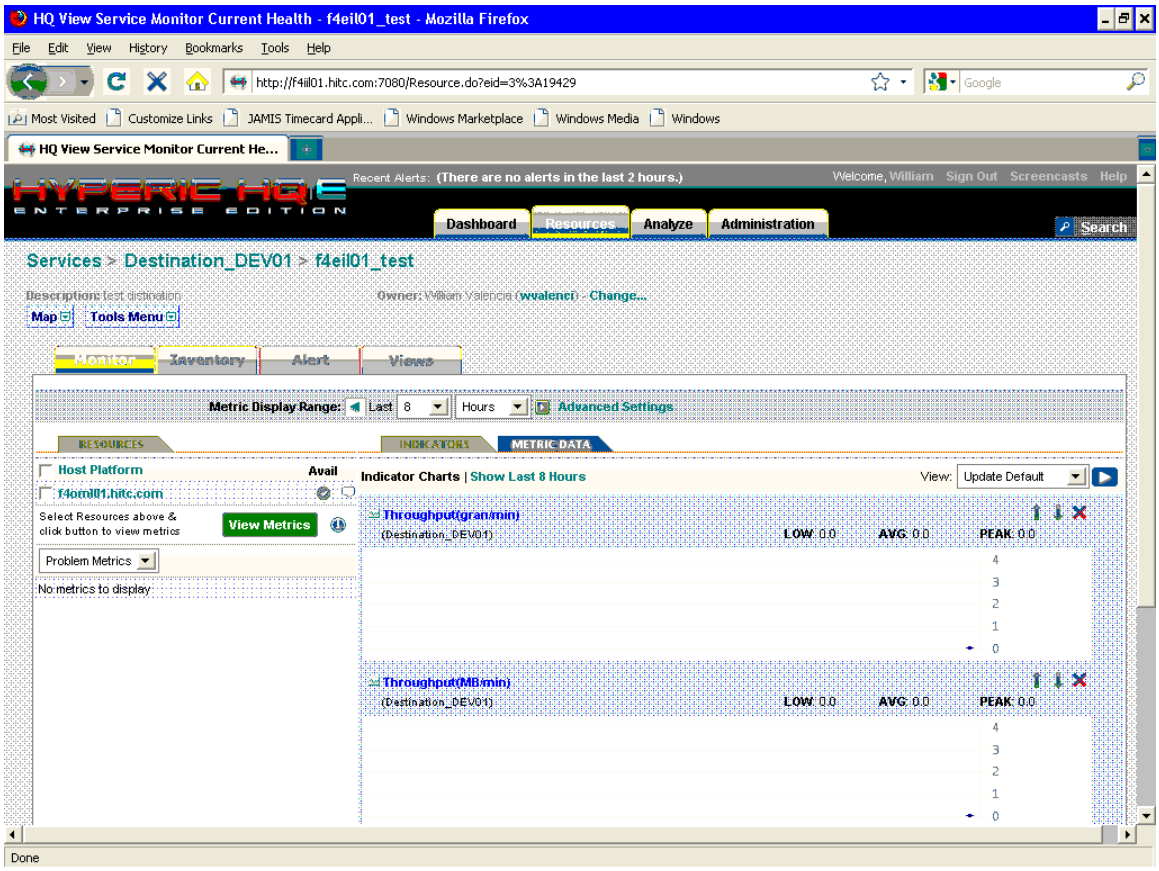

Figure 4.2.2.4.4-4. Individual Destination Monitor Page

Below are the four major steps to set up the OM Destination for monitoring:

1. Define the destination resource through a custom plugin. The destination resource is defined as a platform service type on the OML box. The plugin xml is named <MODE>-OMS-Destination -plugin.xml. It defines the metrics that pertain to the resource type of destination. Please refer to the general instruction on how to deploy a plugin in Hyperic.

2. Define the OPS_DESTINATIONS compatible (not mixed) group. The group type should be destination_OPS. Please refer to the general instruction on how to define a compatible group in Hyperic.

3. Configure individual destinations through Hyperic GUI. See section below for details.

Add the configured individual destination to the OPS_DESTINATIONS group. Please refer to the general instruction on how to add a resource to a group.

Below is a detailed description on how to configure an individual destination using the Hyperic GUI.

Click on the “Resource” tab at the top of the page. Select “Browse” in the drop-down menu. Click on “Platforms” tab to list all the platforms defined in the system. Click on the platform where OMS servers are installed (i.e. X4oml01 box) to bring up the platform detail page. See the screen shot below for an example:

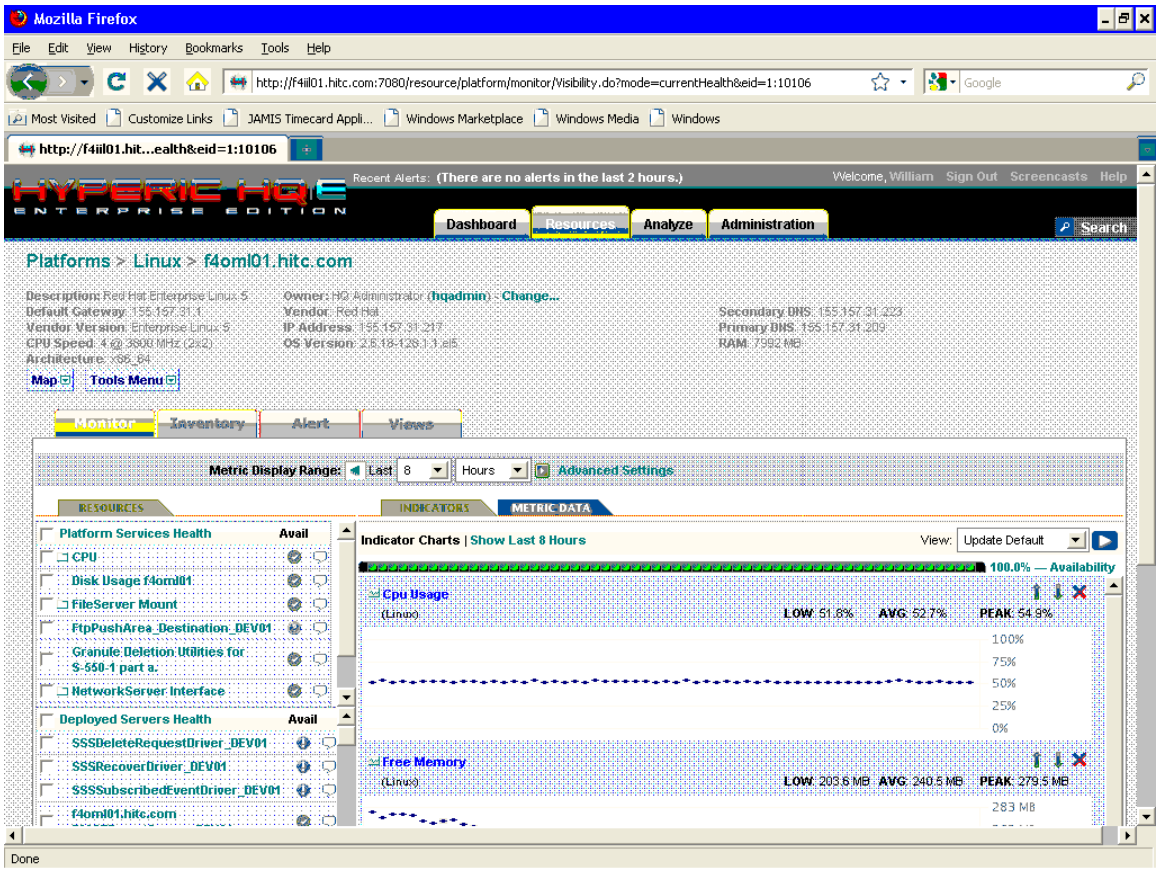

Figure 4.2.2.4.4-5. Host Resource Page

On the platform detail page, click on the “Tools Menu” drop down that is located on the upper left corner of the page, then click on “New Platform Service” in the drop down menu. See Screen shot below:

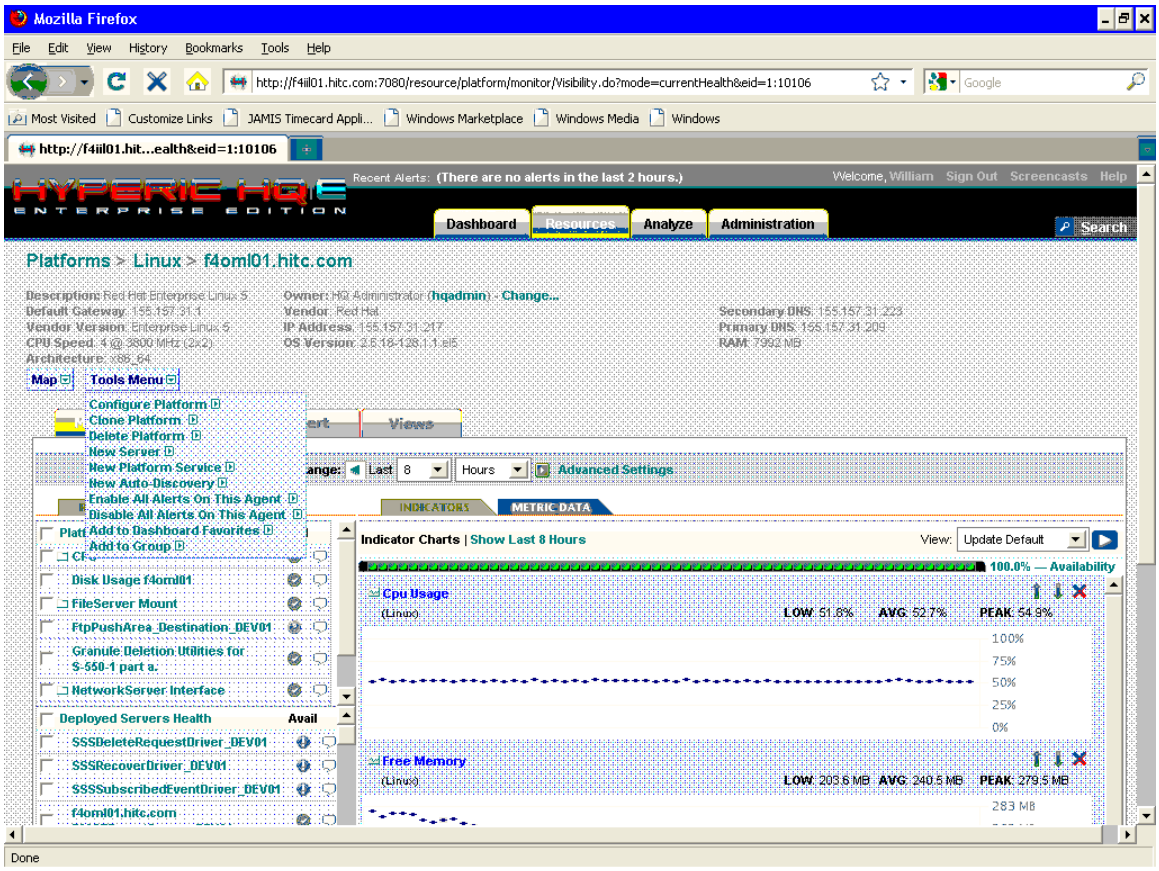

Figure 4.2.2.4.4-6. Add New Platform Service

This brings up the new platform service configuration page. On this page, fill in the name of the destination and description, on the Service Type drop down menu, select “Destination_OPS”. See screen shot below:

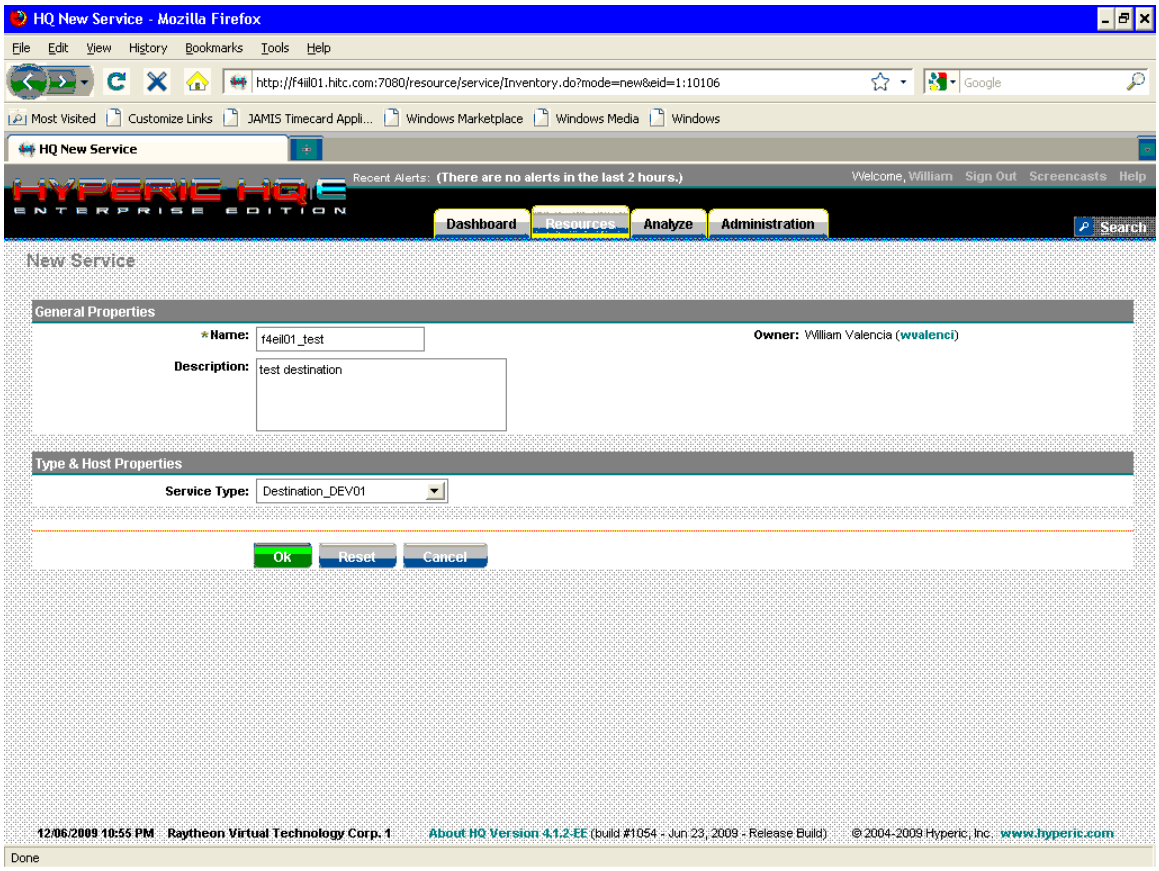

Figure 4.2.2.4.4-7. Configure Individual Destination

Then click “OK”. The service detail page will pop up as a result. See below:

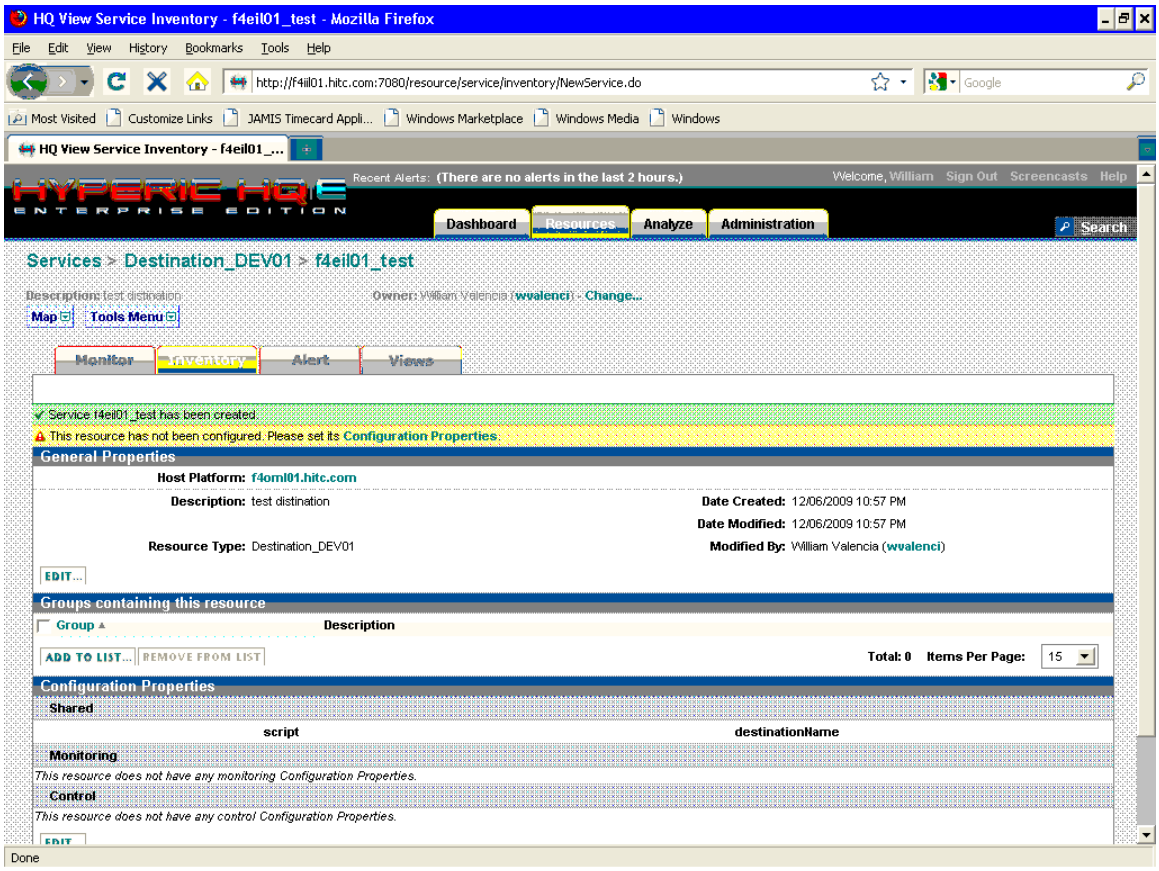

Figure 4.2.2.4.4-8. Individual Destination Inventory Page

Click on the link to configure its configuration properties. On the popped up page, the script input text box should be prepopulated and does not need to be modified. In the destinationName input text box, type in the name of the destination. **Note: the destination name typed in has to exactly match the DestinationName field defined in the OmConfigDestination table in OMS database.** See the screen shot below:

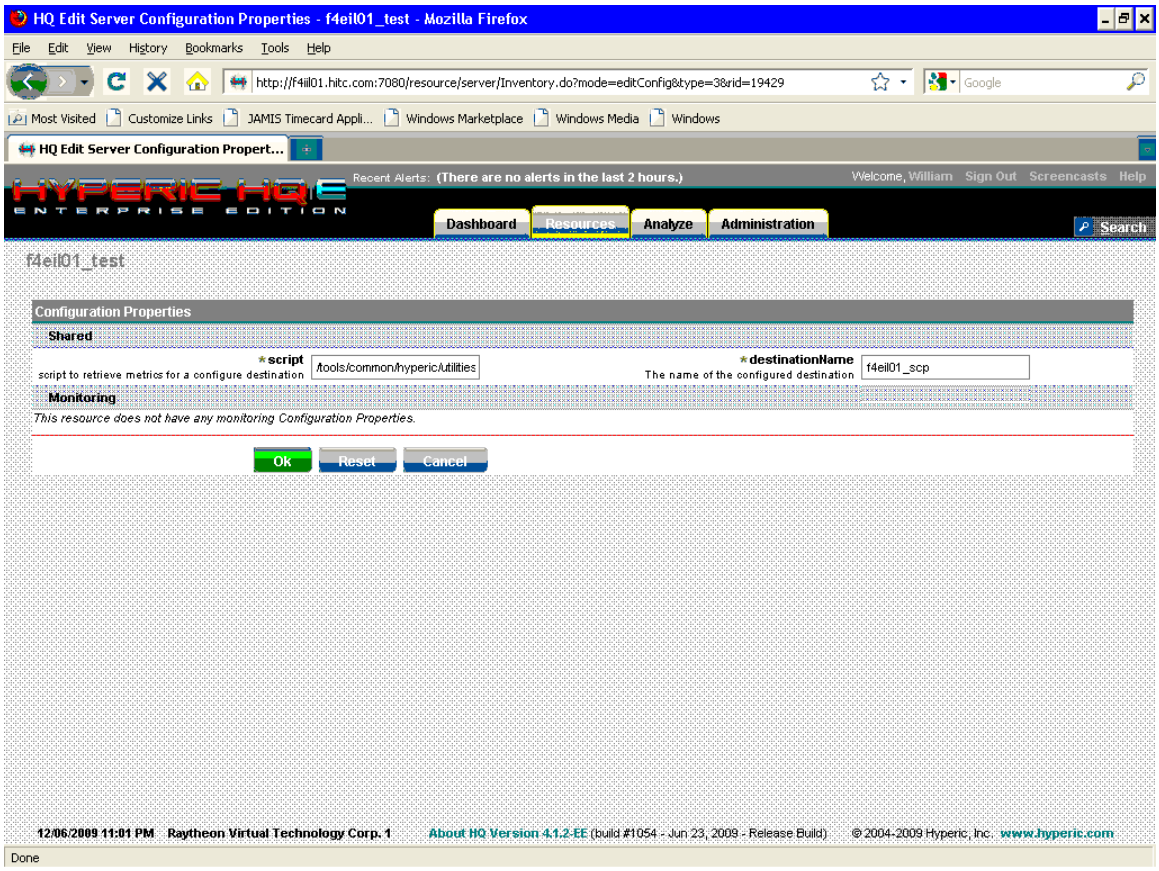

Figure 4.2.2.4.4-9. Individual Destination Configuration Page

Click on “OK” to accept the configuration. The screen will go back to the service detail page. That is the end of configuring an individual destination.

Once the destination is configured, the Hyperic server will be able to collect the defined metrics by executing the script (defined in the plugin xml file) periodically which invokes a Perl script to retrieve the requested statistic information from the Ingest database. The Perl script invokes a stored proc named SmGetDestinationMetrics.sp which takes the destination name as input and outputs the metrics information for that destination.

4.2.2.4.4.5 HEG Server

The HEG Server monitoring is done through an xml custom plug-in. There is one plug-in file per mode. The plug-in file name is <MODE>-HEG-HEGServer-plugin.xml. The HEG Server resource will be auto-discovered by Hyperic using the PTQL query defined in the xml plug-in. The standard metrics will be collected by the SIGAR plug-in.

Up/down alerts are set up for the HEG Server based on its availability metrics. The down alert is used to notify operator of the problem and update the Order Management business process status

and trigger the control action that will automatically start the server; the up alert is used as a recovery alert for the down alert to mark the down alert as fixed.

Please refer to the Polling Server, section 4.2.2.4.2.3 for details on the control actions and alerts configurations. The configuration process is basically identical except the resource type, control action, script names and alert names are changed accordingly.

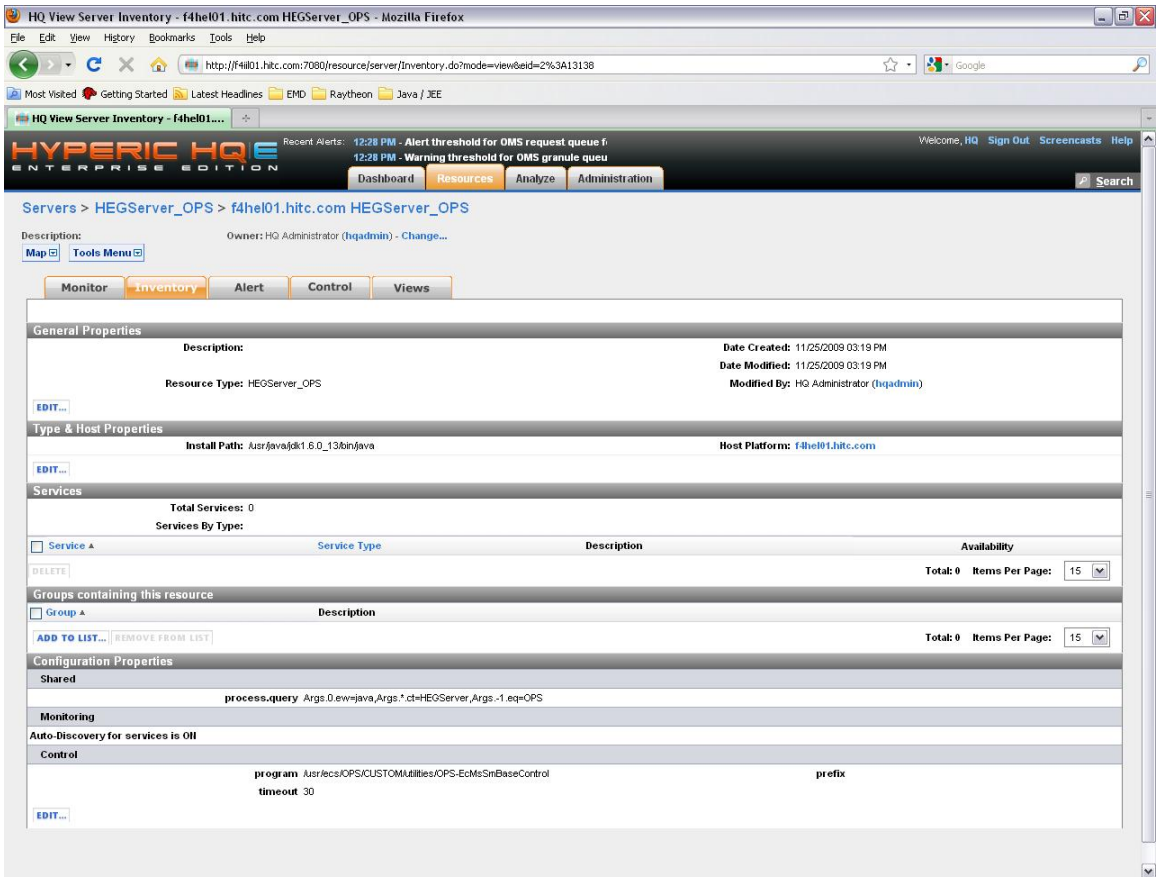

Figure 4.2.2.4.4-10. HEGServer Configuration View

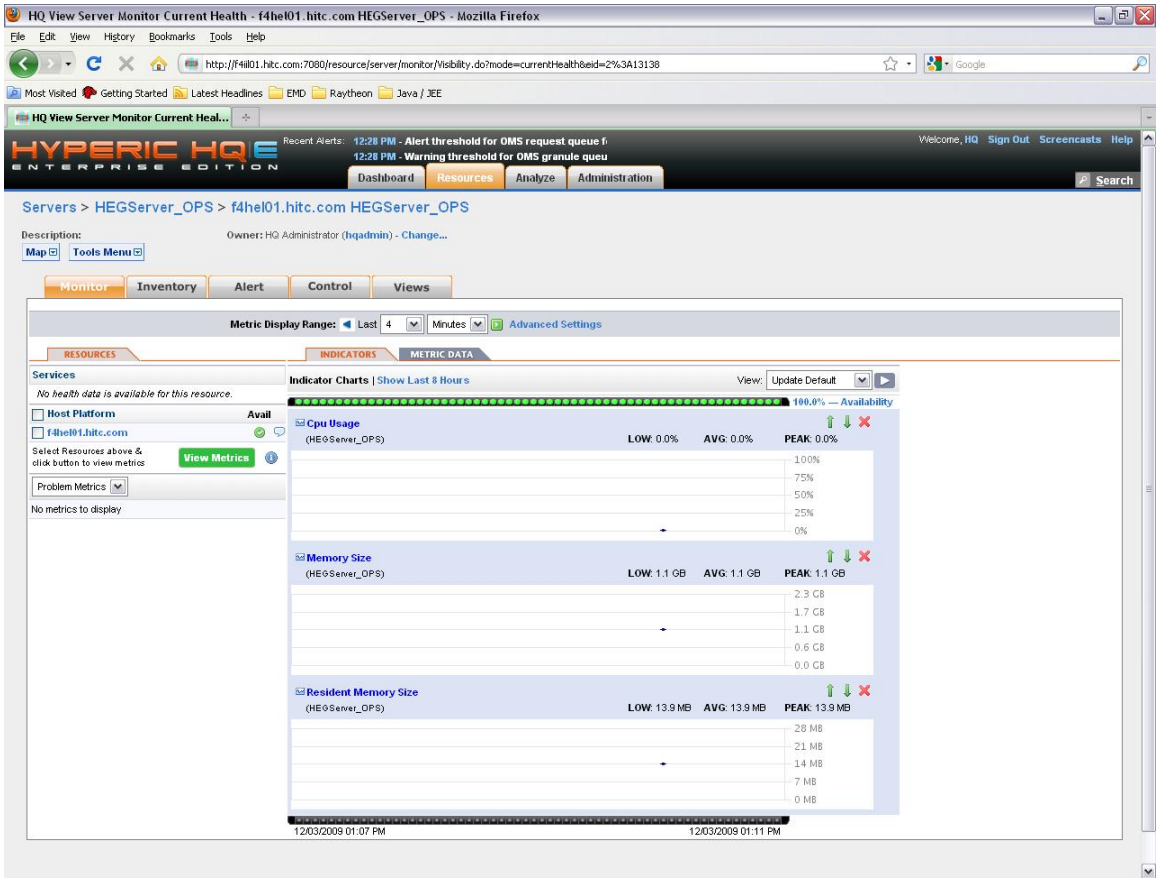

Figure 4.2.2.4.4-11. HEGServer Monitoring

4.2.2.4.4.6 EPD Server

The EPD Server monitoring is done through an xml custom plug-in. There is one plug-in file per mode. The plug-in file name is <MODE>-OMS-EPDServer-plugin.xml. The EPD Server resource will be auto-discovered by Hyperic using the PTQL query defined in the xml plug-in. The standard metrics will be collected by the SIGAR plug-in.

Up/down alerts are set up for the EPD Server based on its availability metrics. The down alert is used to notify operator of the problem and update the Order Management business process status and trigger the control action that will automatically start the server; the up alert is used as a recovery alert for the down alert to mark the down alert as fixed.

Please refer to the Polling Server, section 4.2.2.4.2.3for details on the control actions and alerts configurations. The configuration process is basically identical except the resource type, control action, script names and alert names are changed accordingly.

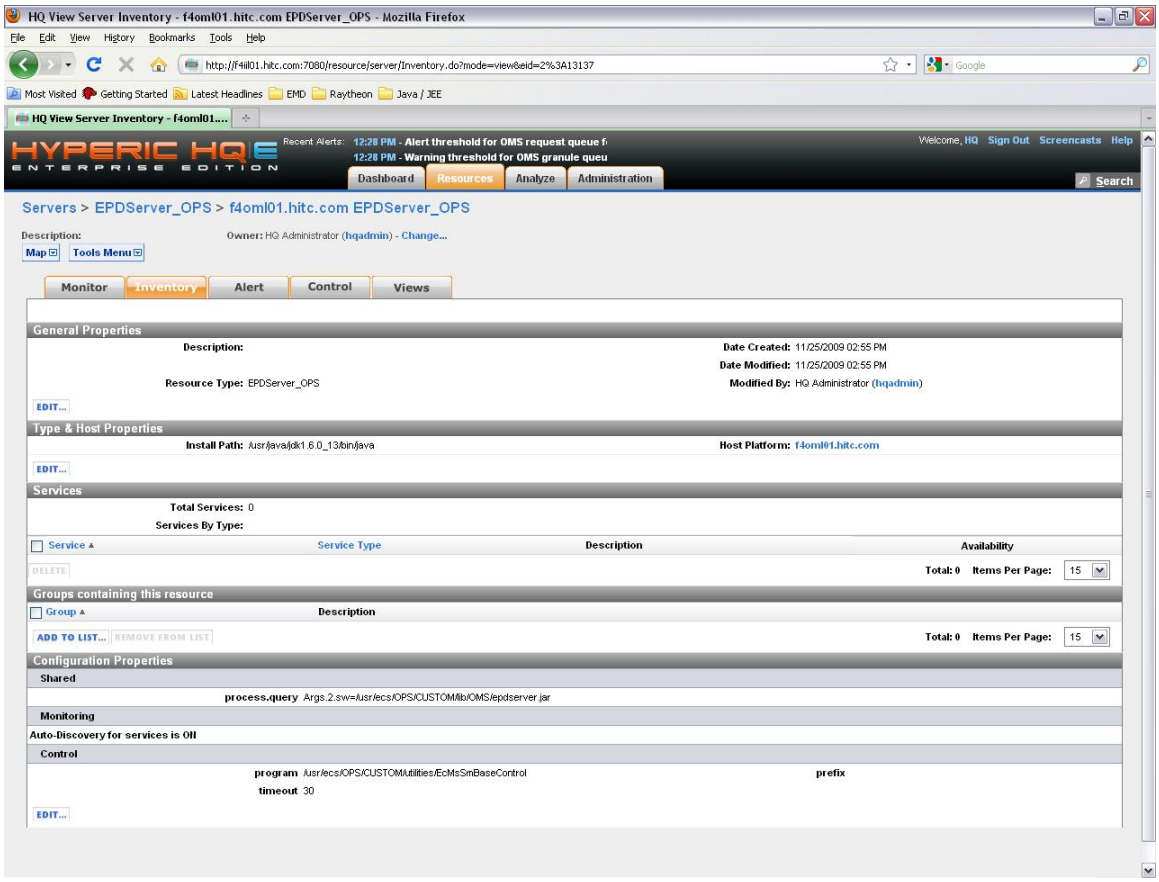

Figure 4.2.2.4.4-12. EPDServer Configuration View

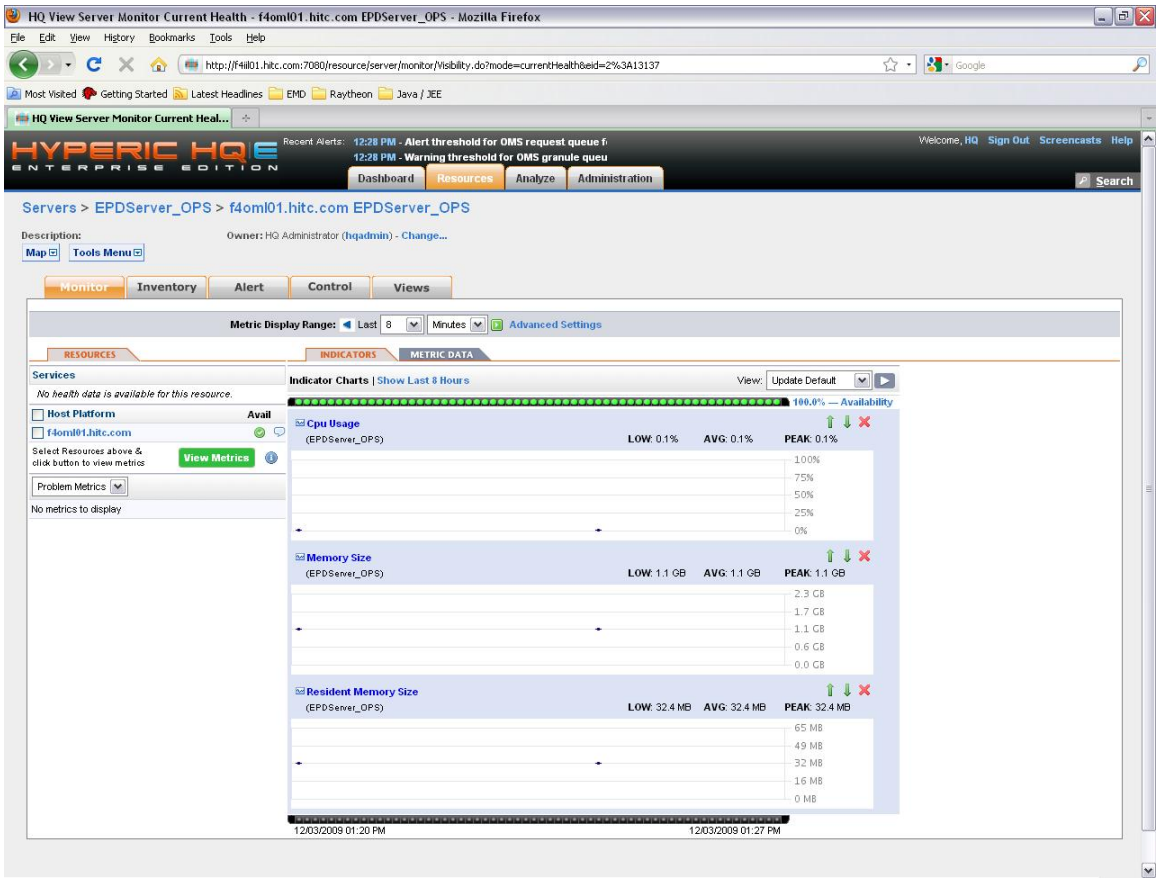

Figure 4.2.2.4.4-13. EPDServer Monitoring

4.2.2.4.4.7 Copy Server

The Order Manager Copy Server monitoring is done through an xml custom plug-in. There is one plug-in file per mode. The plug-in file name is <MODE>-OMS-CopyServer-plugin.xml. The plugin defines the server type and an operator will need to configure Hyperic to monitor the Copy Server. Below are the steps to configure Hyperic to monitor this resource.

1. Select the platform where the Copy Server runs on. The agent on this platform will be in charge of gathering metrics for this resource. Mouse over the 'Resources' tab and select the 'Browse' option. Click on the 'Platforms' link if it isn't highlighted and select a platform.

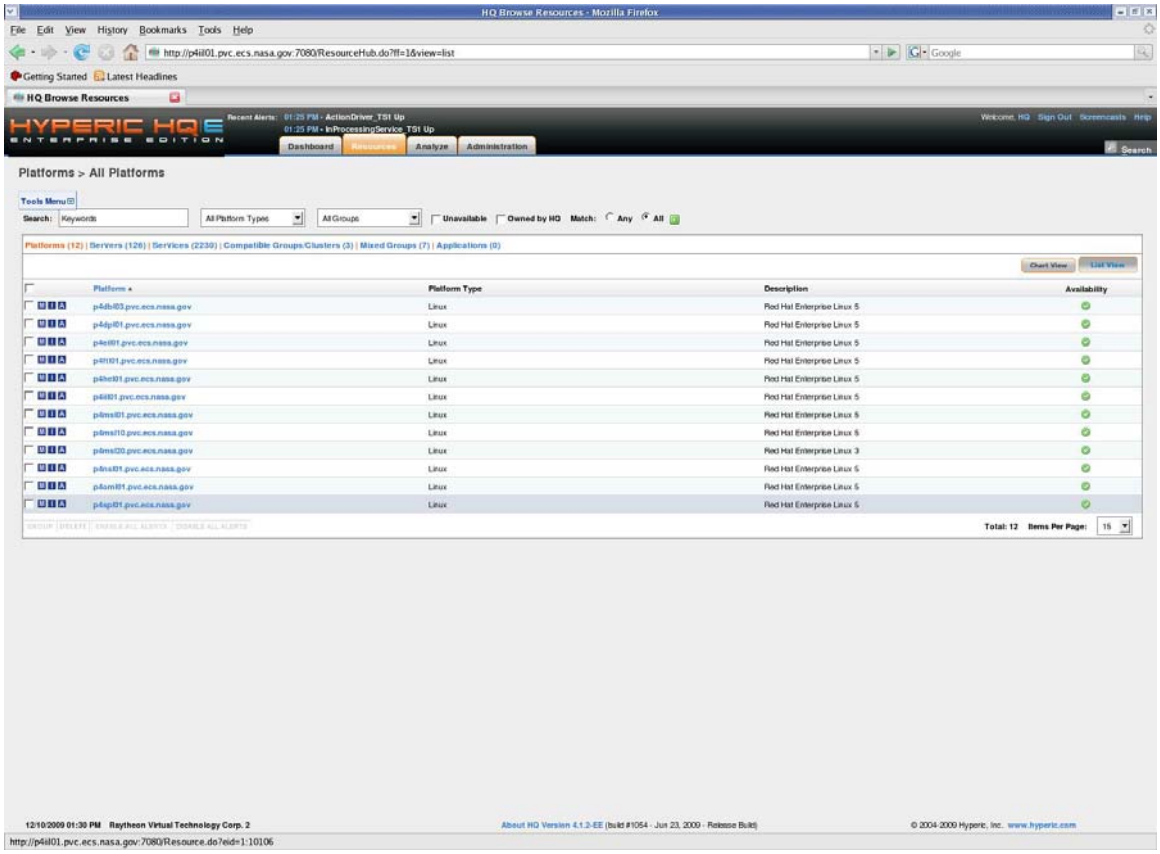

Figure 4.2.2.4.4-14. Copy Server Select Platform

2. In the platform's detailed page, click on the 'Tools Menu' button and select the 'New Server' option.

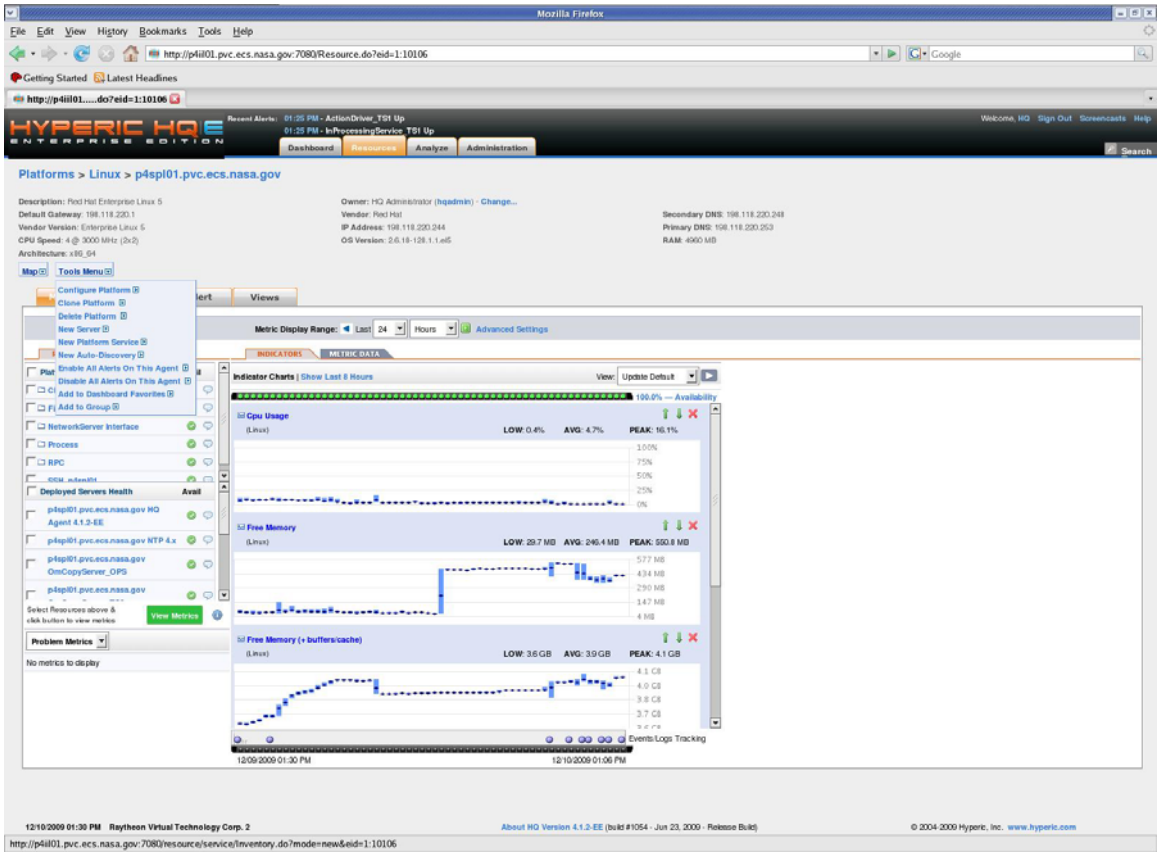

Figure 4.2.2.4.4-15. Copy Server Add New Server

3. A new server general properties page will be displayed. Fill in the name, description, server type, and installpath. Afterward, click the 'OK' button.

   Name: OmCopyServer_OPS

   Description: This service monitors the OmCopyServer in the OPS mode.

   Server Type: OmCopyServer_OPS

   Installpath: /usr/ecs/OPS/CUSTOM/bin/OMS/EcOmSrCopyServer.pl

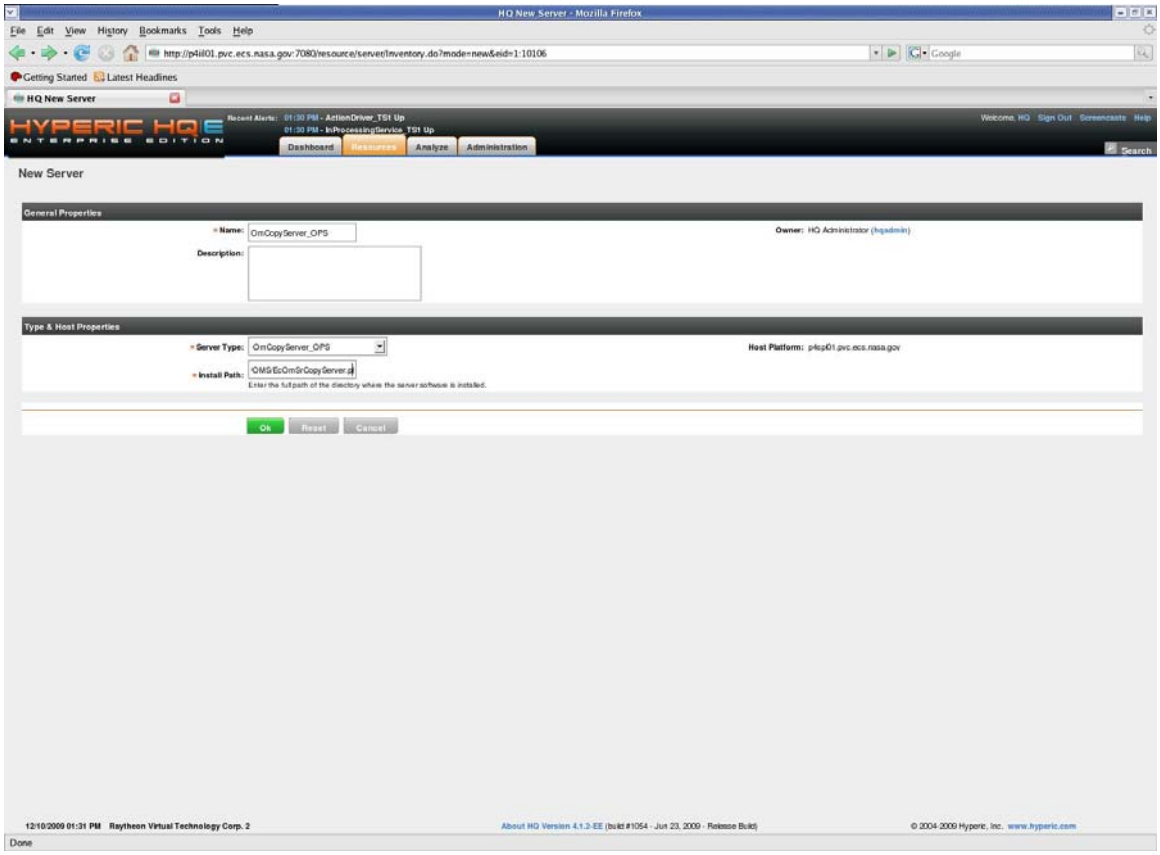

Figure 4.2.2.4.4-16. Copy Server General Properties

4. You will be taken to the server detail page, but the server has not been configured to monitor the Copy Server yet. Click on the 'Configuration Properties' link.

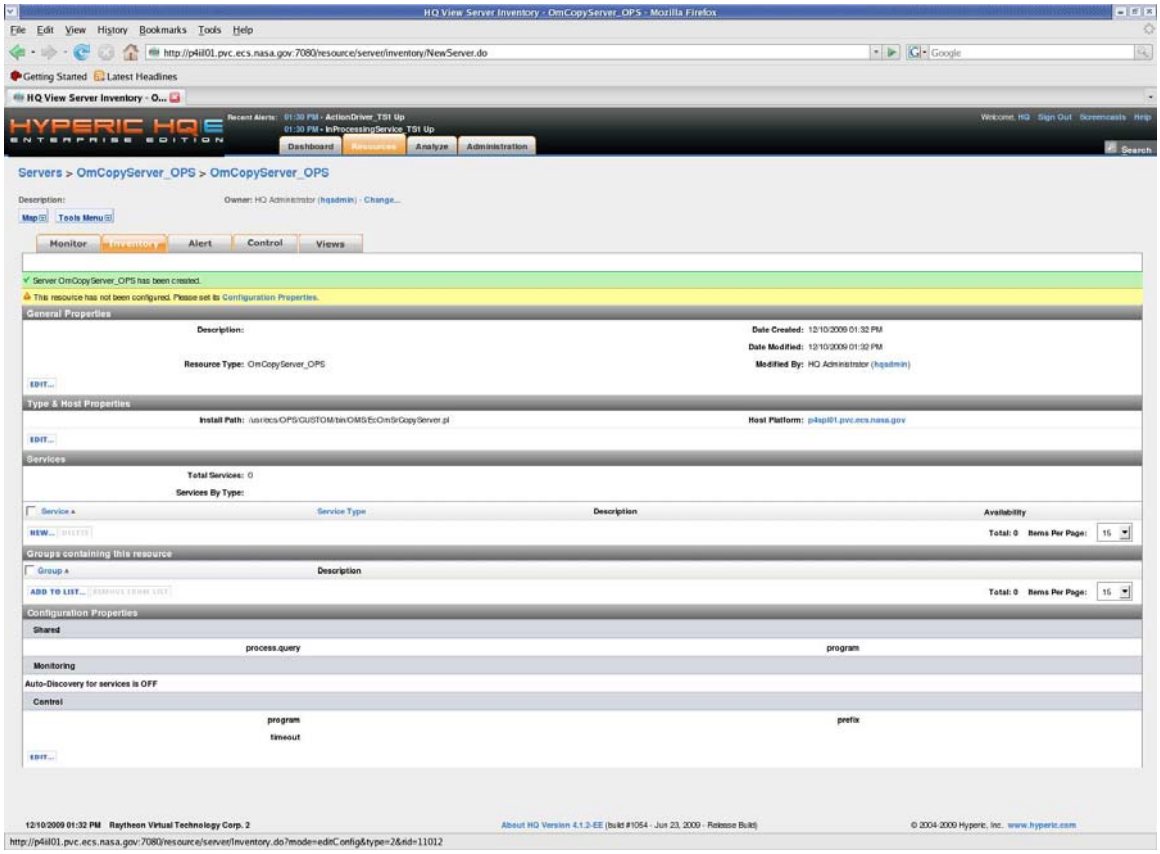

Figure 4.2.2.4.4-17. Copy Server Configuration Properties Link

5. In the configuration properties page, leave all the properties as it is and verify that the control program is pointing to /tools/common/hyperic/utilities/<MODE>-EcMsSmBaseControl (where MODE is fill in) and click the 'OK' button.

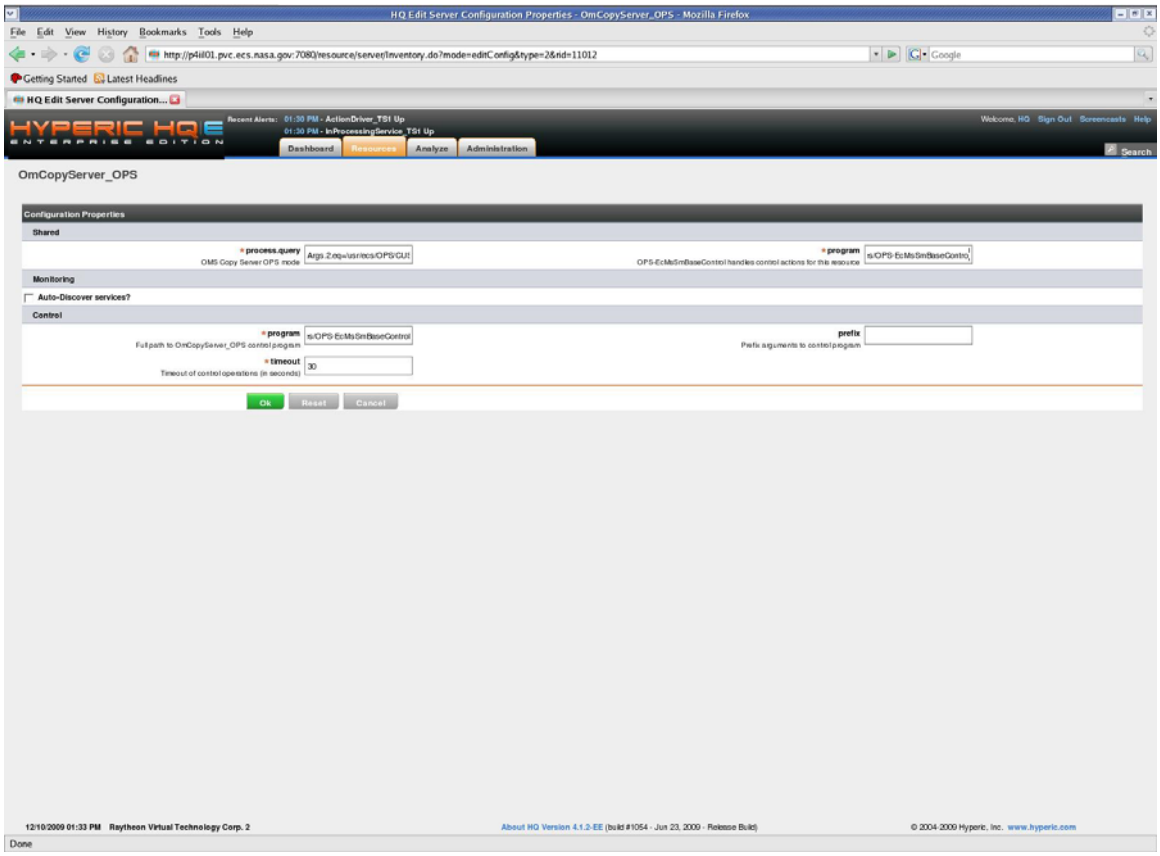

Figure 4.2.2.4.4-18. Copy Server Configuration Properties

6. The Copy Server is now configured and you can go ahead and monitor it, setup alerts, and change the monitoring interval of the metrics it gathers.

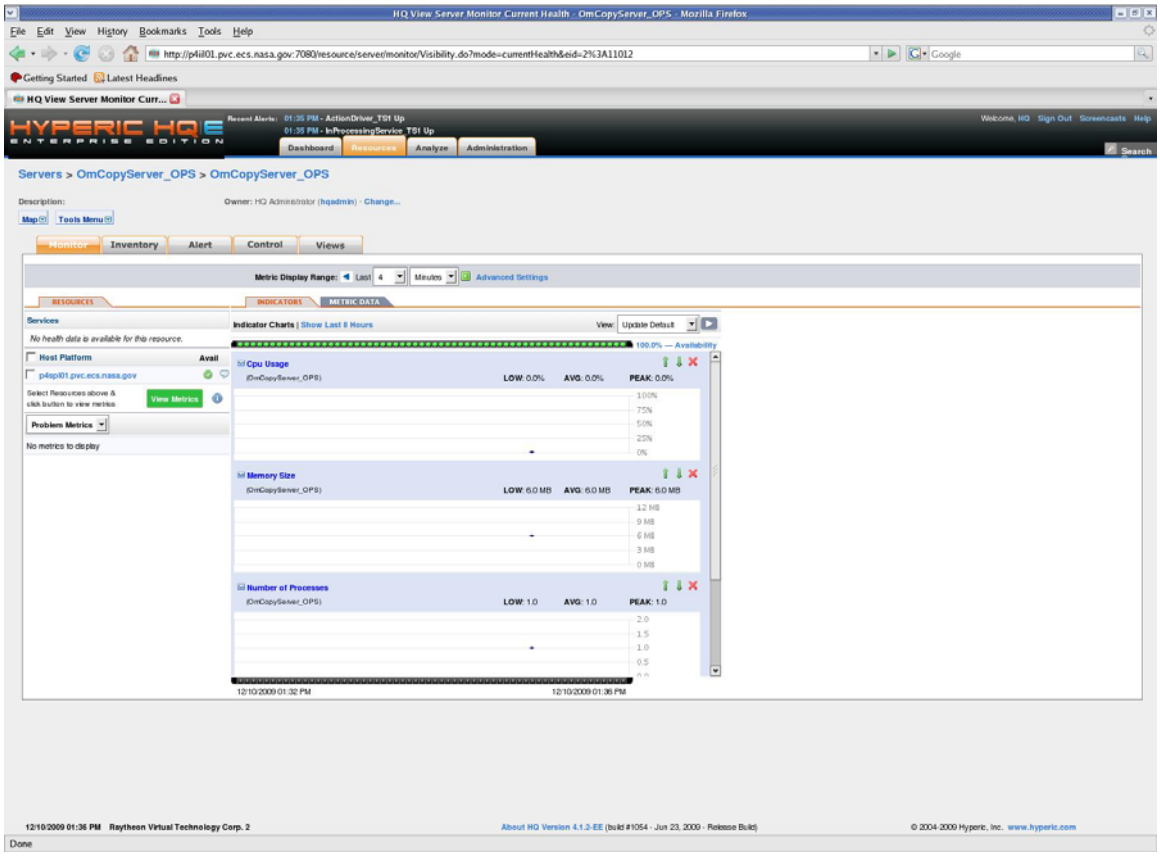

Figure 4.2.2.4.4-19. Copy Server Monitoring

The metrics that will be collected for the Copy Server include the standard Hyperic metrics (see section 4.2.2.2.13: Standard Plugin Metrics) and custom metrics (see below).

Table 4.2.2.4.4-2. Copy Server Metrics

Custom Metrics	
Metric Name	**Description**
Number of Processes	The number of Copy Server instances running.

Up/down alerts are set up for the Copy Server based on its availability metrics. The down alert is used to notify operator of the problem and update the Order Management business process status and trigger the control action that will automatically start the server; the up alert is used as a recovery alert for the down alert to mark the down alert as fixed.

Please refer to the Polling Server, section 4.2.2.4.2.3for details on the control actions and alerts configurations. The configuration process is basically identical except the resource type, control action, script names and alert names are changed accordingly.

4.2.2.4.4.8 EWOC

The URL to the EWOC Apache Axis homepage is http://<hostname>:<port number>/EWOC/. EWOC provides two services, OrderStatusUpdate and OrderFullfillment. The example below will show you how to configure HTTP services via Hyperic to monitor the EWOC OrderStatusUpdatePort and OrderFullfillmentPort wsdl.

Note: WSDL is an xml-based language which describes a web service.

1. Select a platform where you want to define your HTTP service. The agent on this platform will be in charge of submitting an http request and reading the responses. Mouse over the 'Resources' tab and select the 'Browse' option. Click on the 'Platforms' link if it isn't highlighted and select a platform.

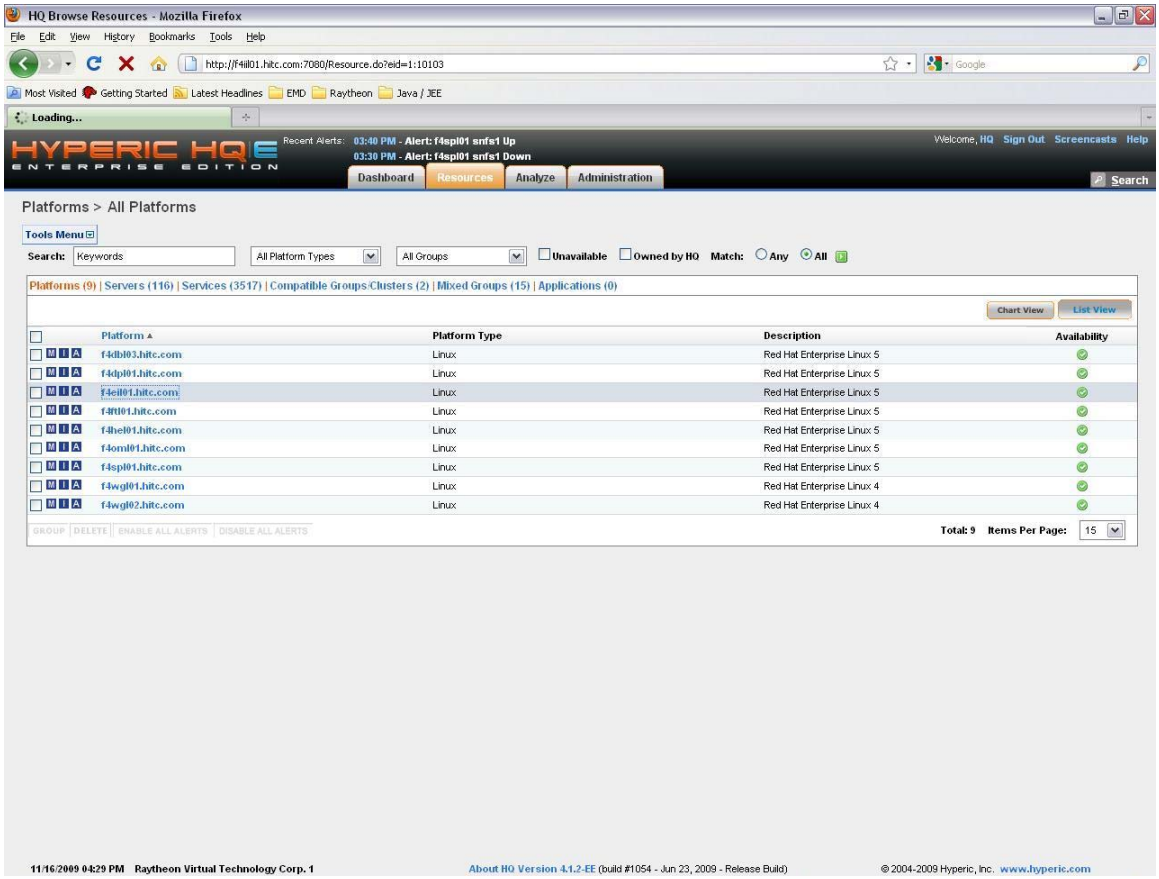

Figure 4.2.2.4.4-21. Select EWOC HTTP Service Platform

2. In the platform's detailed page, click on the 'Tools Menu' button and select the 'Add Platform Service' option.

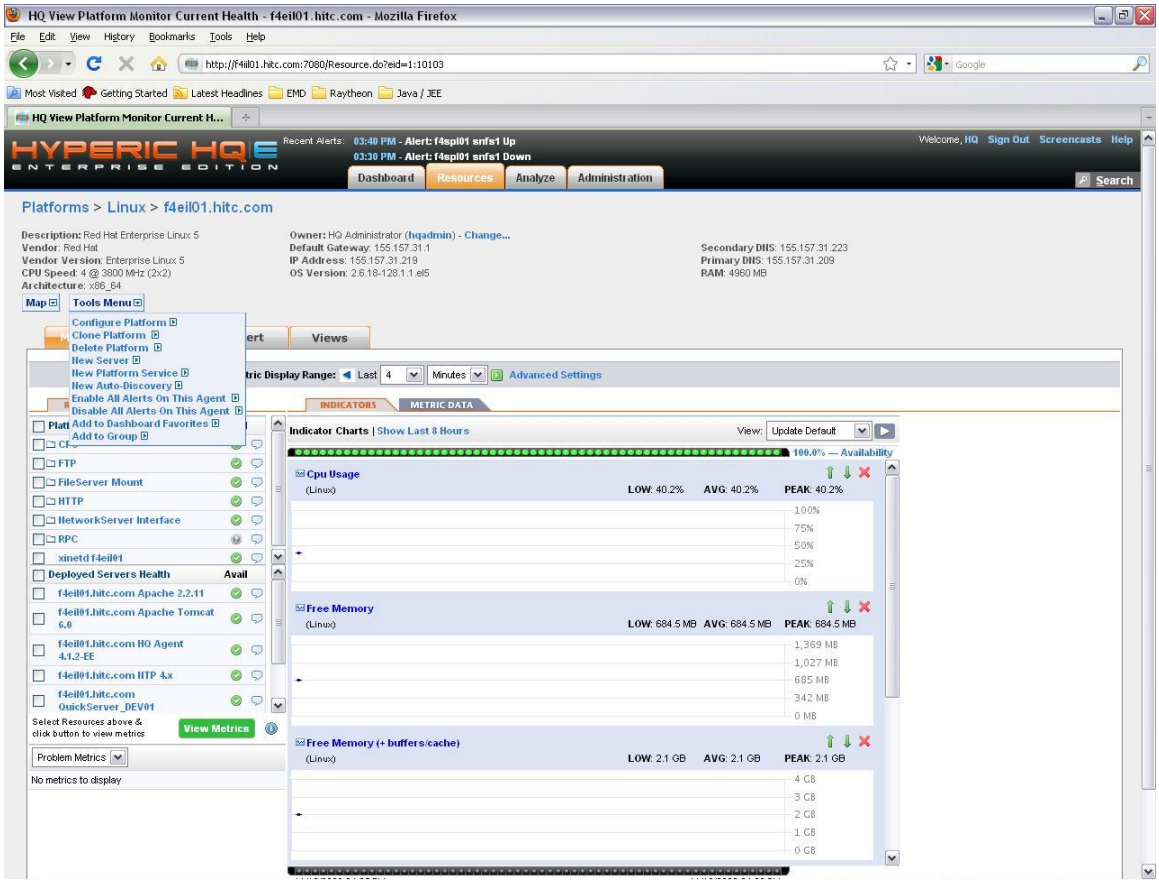

Figure 4.2.2.4.4-22. EWOC Add New Service

3. A new service general properties page will be displayed. Fill in the name, description, and service type. Afterward, click the 'OK' button.

   Name: Monitor EWOC_OrderStatusUpdatePort_OPS

   Description: This service monitors EWOC OrderStatusUpdatePort wsdl.

   Service Type: HTTP

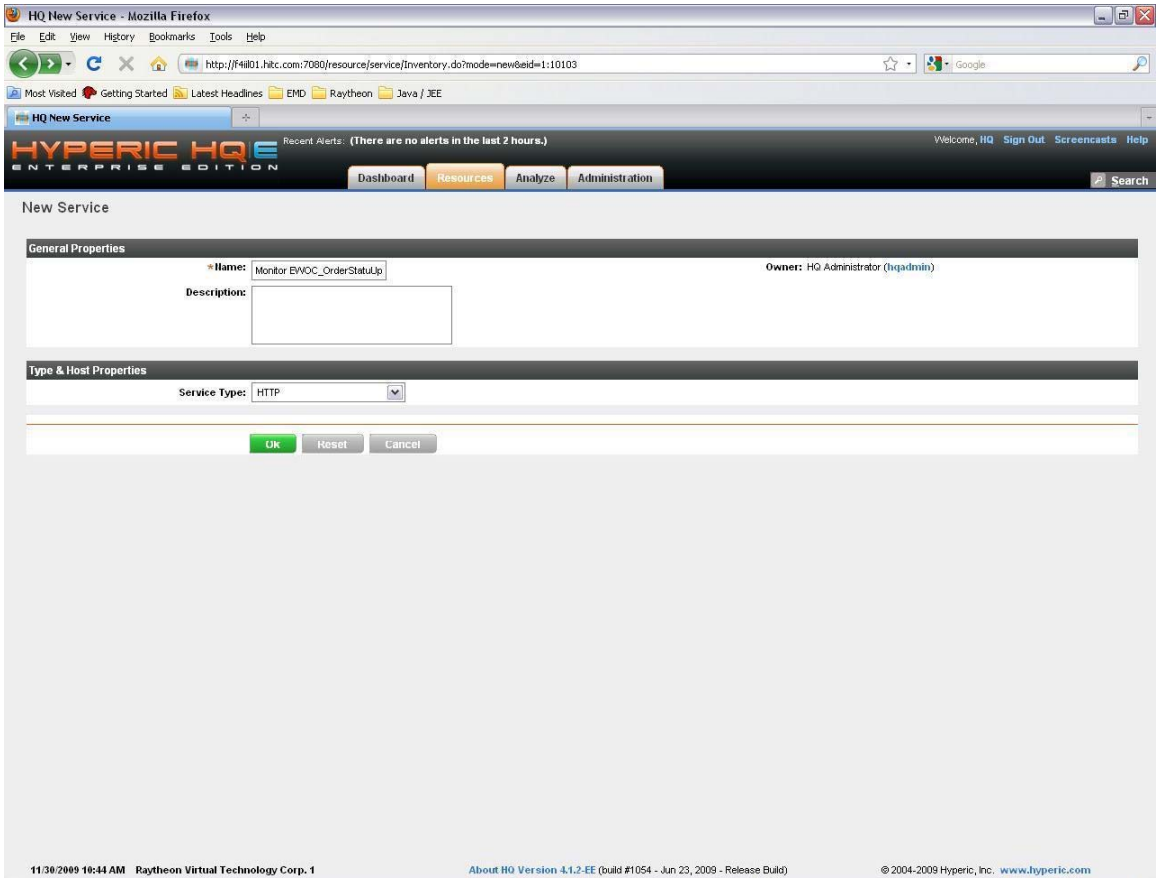

Figure 4.2.2.4.4-23. EWOC OrderStatusUpdatePort General Properties

4. You will be taken to the service detail page, but the service has not been configured to monitor the EWOC OrderStatusUpdatePort wsdl yet. Click on the 'Configuration Properties' link.

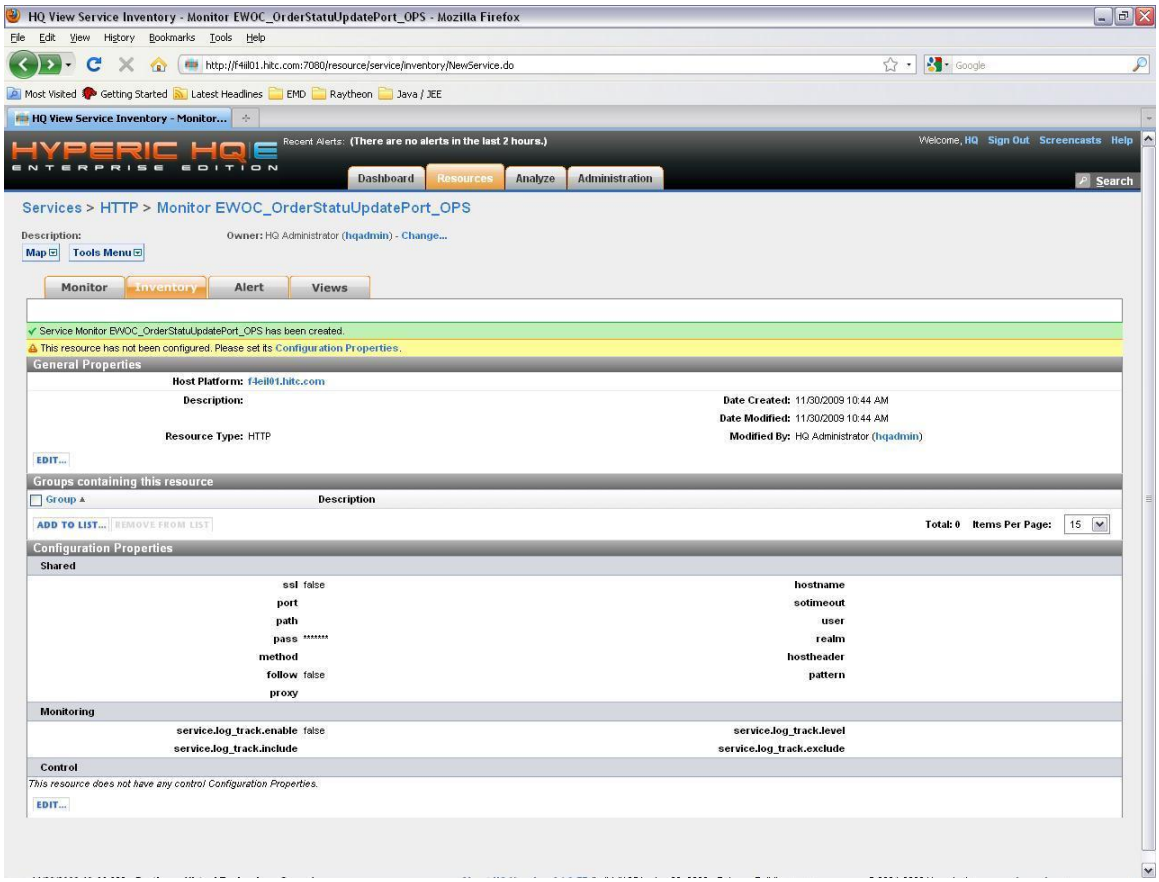

Figure 4.2.2.4.4-24. EWOC OrderStatusUpdatePort Configuration Properties Link

5. In the configuration properties page fill in the hostname, port, path, and method fields. When finish click the 'OK' button. Here's an example configuration of the EWOC OrderStatusUpdatePort wsdl in the EDF.

   hostname: f4eil01.hitc.com

   port: 22500

   sotimeout: 10

   path: /EWOC/services/OrderStatusUpdatePort?wsdl

   method: GET

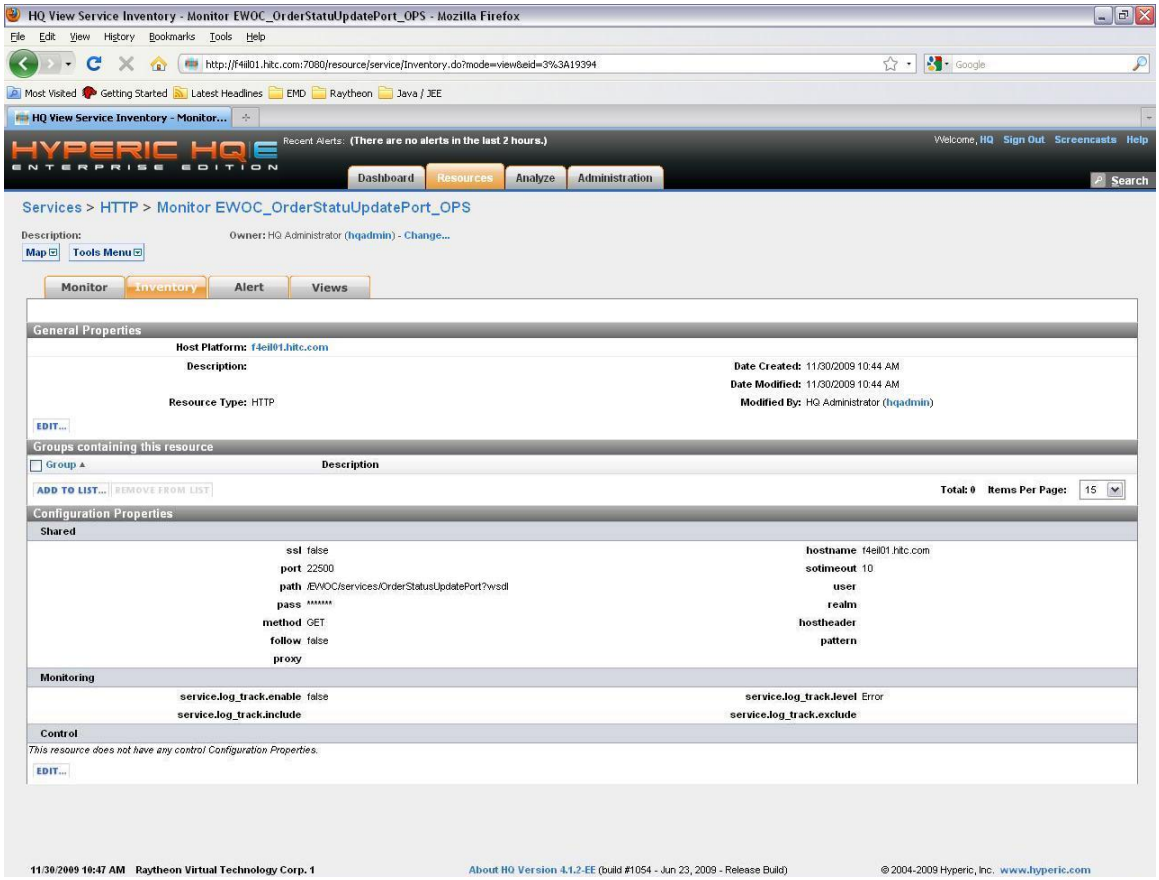

Figure 4.2.2.4.4-25. EWOC OrderStatusUpdatePort Configuration Properties Page

6. The EWOC OrderStatusUpdatePort HTTP service is now configured and you can go ahead and monitor it, setup alerts, and change the monitoring interval of the metrics it gathers.

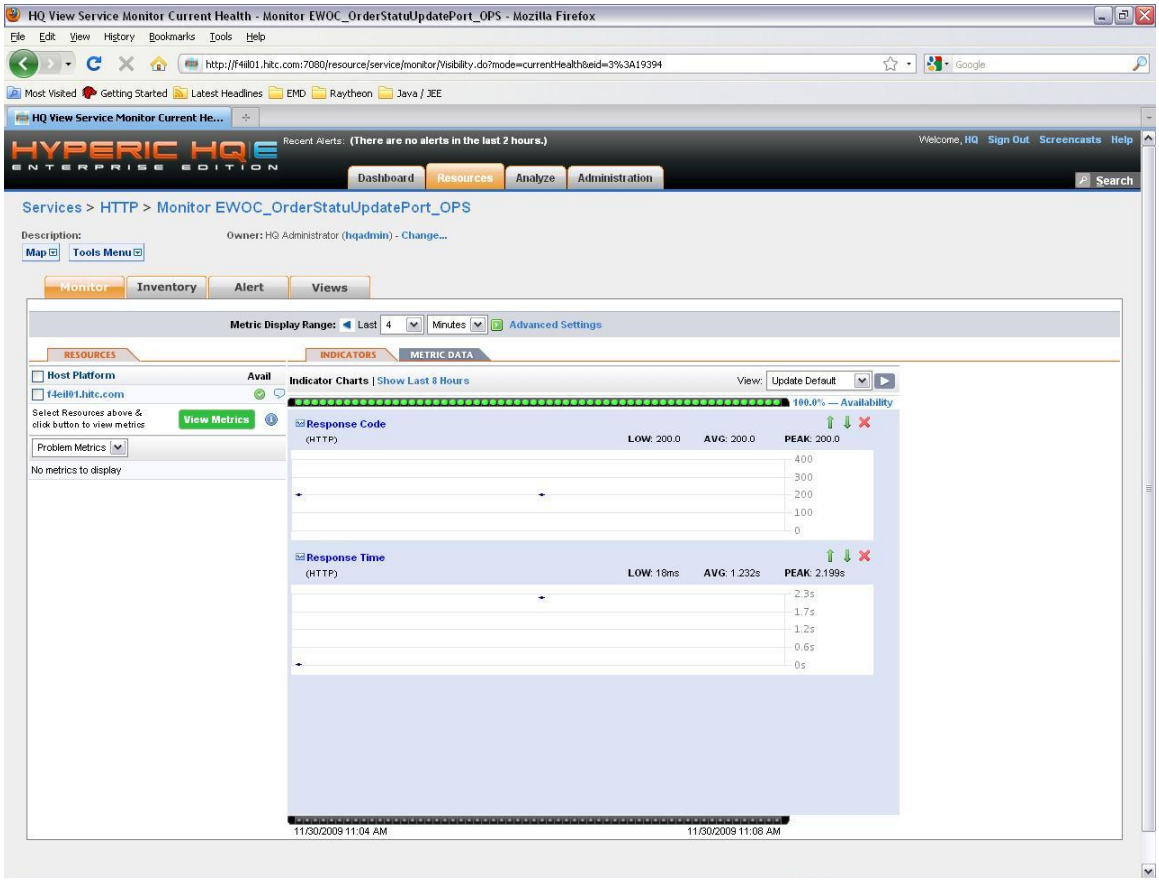

Figure 4.2.2.4.4-26. EWOC OrderStatusUpdatePort Monitoring

7. Perform steps 1 – 6 to configure Hyperic to monitor the OrderFullfillmentPort wsdl. The image below shows an example of the configuration in EDF.

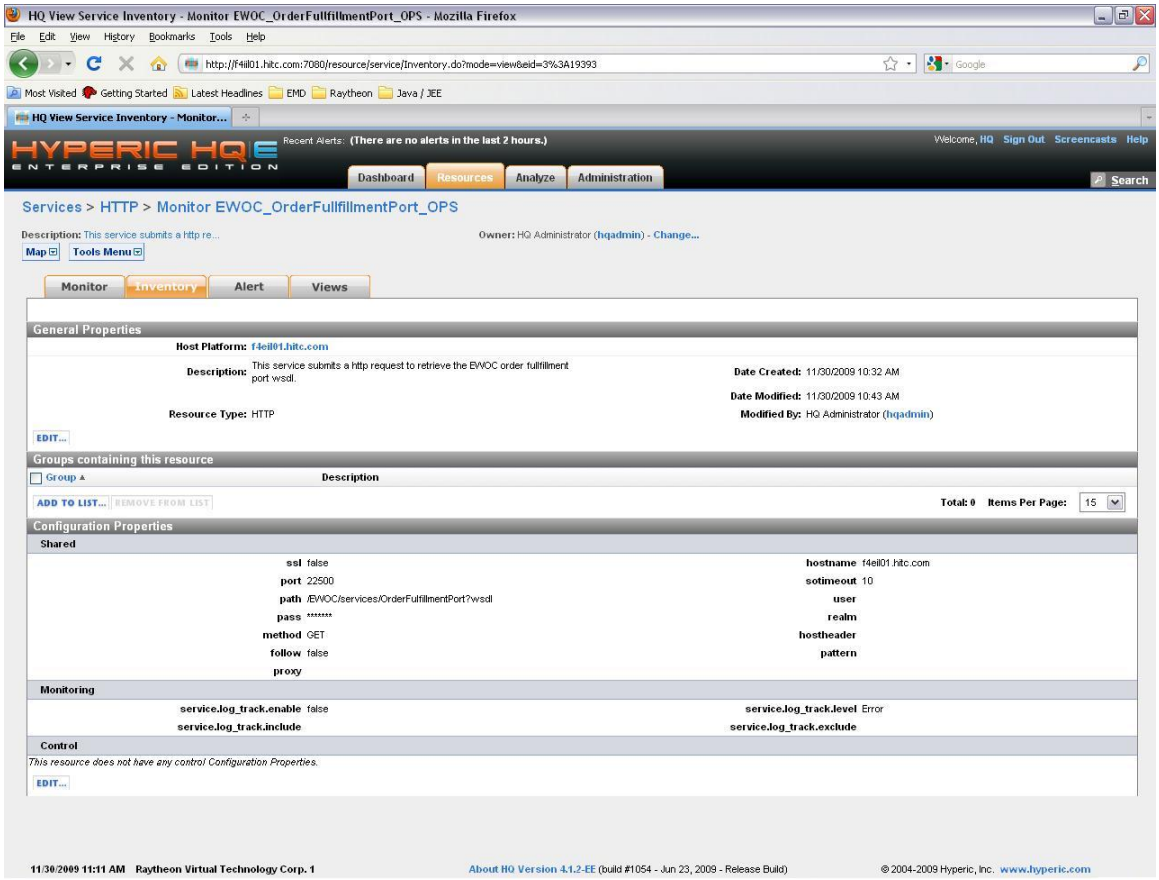

Figure 4.2.2.4.4-27. EWOC OrderFullfillmentPort Configuration Properties Page

4.2.2.4.4.9 WIST

See section 4.2.2.5, ECHO WIST.

4.2.2.4.4.10 DataPool File System

See section 4.2.2.4.2.9, DataPool File System.

4.2.2.4.4.11 Sybase

See section 4.2.2.3.1, Configuring Sybase.

4.2.2.4.4.12 HSA

A Hyperic agent will not be installed on the HSA host, thus it is not possible to directly monitor the HSA service. Instead, Hyperic will be configured to ping the HSA host to ensure it is available. The following steps describe how to configure Hyperic to monitor the HSA host using the InetAddress Ping service.

1. Select a platform where you want to define your InetAddress Ping service. The agent on this platform will ping the HSA host periodically. Mouse over the ‘Resources’ tab and select the ‘Browse’ option. Click on the ‘Platforms’ link if it isn’t highlighted and select a platform.

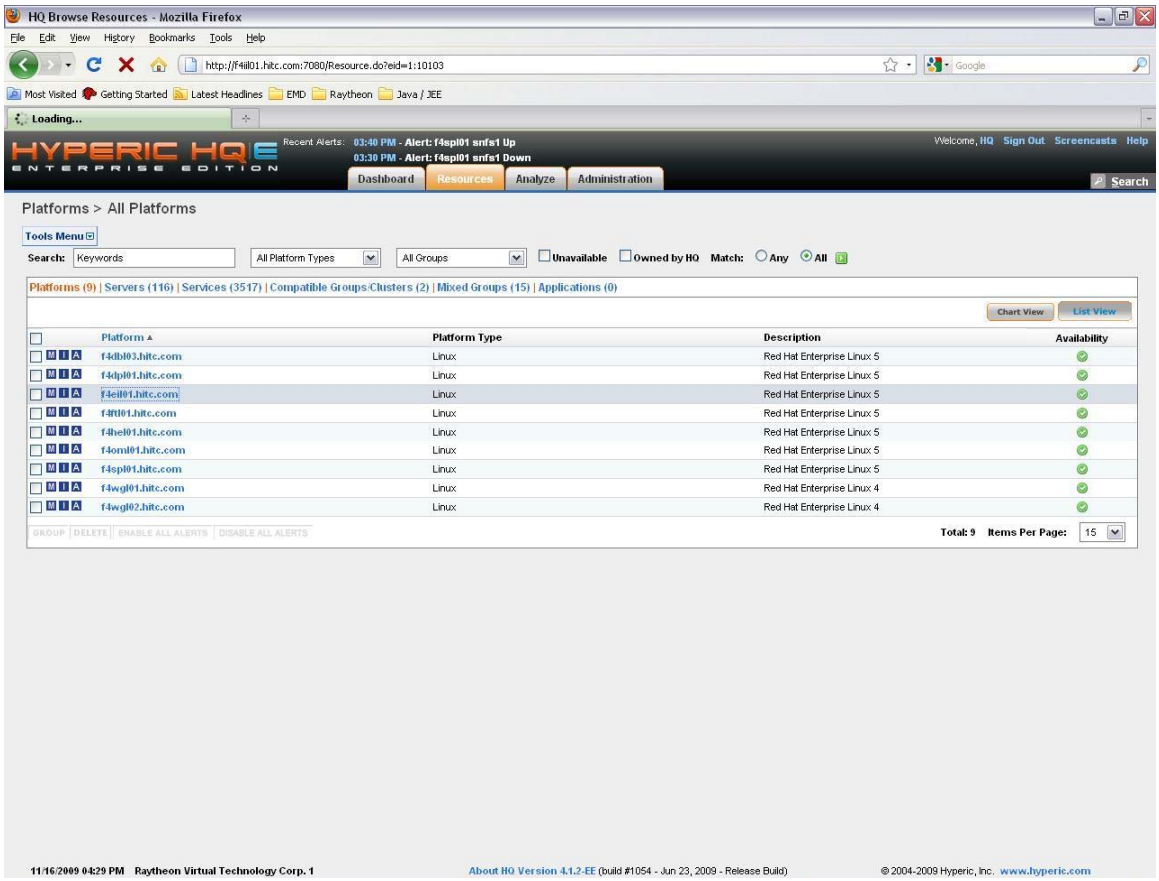

Figure 4.2.2.4.4-28. Select HSA InetAddress Service Platform

2. In the platform’s detailed page, click on the ‘Tools Menu’ button and select the ‘Add Platform Service’ option.

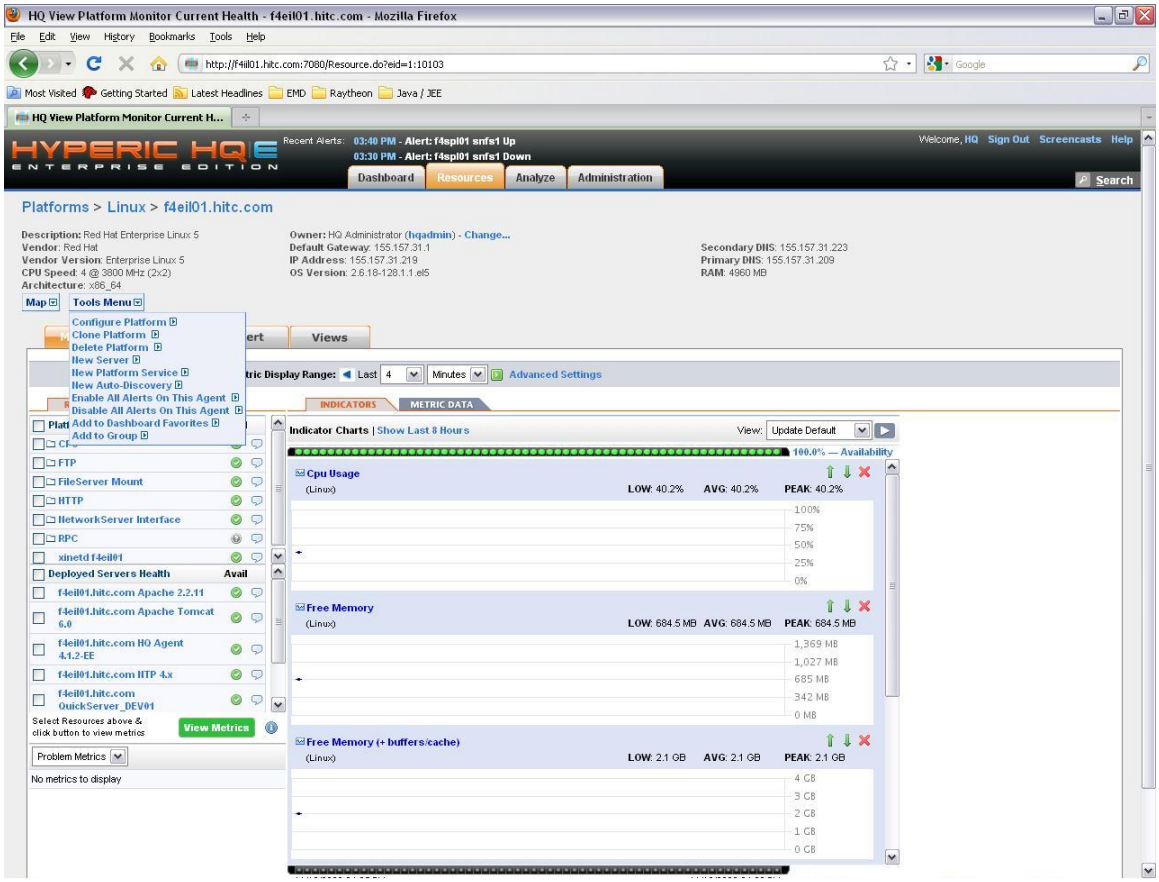

Figure 4.2.2.4.4-29. HSA Add New Service

3. A new service general properties page will be displayed. Fill in the name, description, and service type. Afterward, click the 'OK' button.

   Name: Monitor HSA Host

   Description: This service monitors the HSA host.

   Service Type: InetAddress Ping

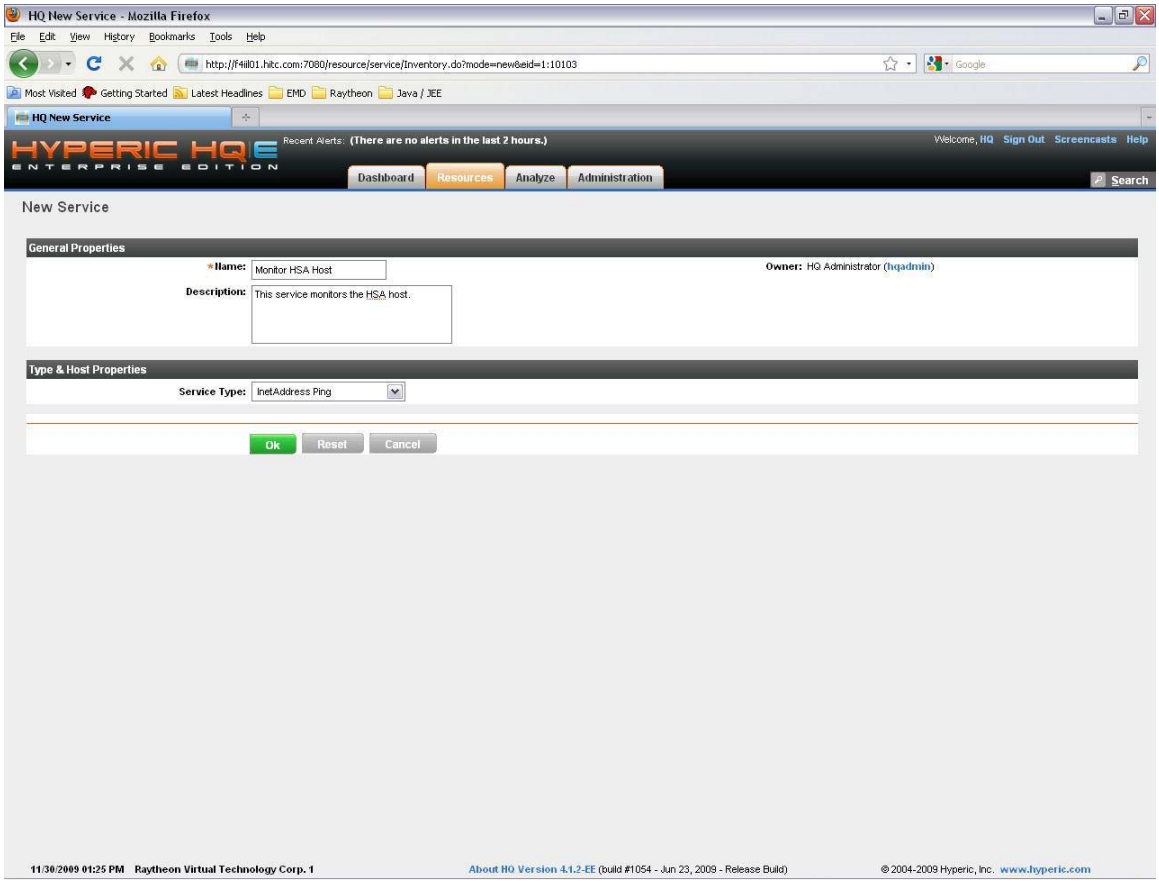

Figure 4.2.2.4.4-30. HSA General Properties

4. You will be taken to the service detail page, but the service has not been configured to ping the HSA host. Click on the ‘Configuration Properties’ link.

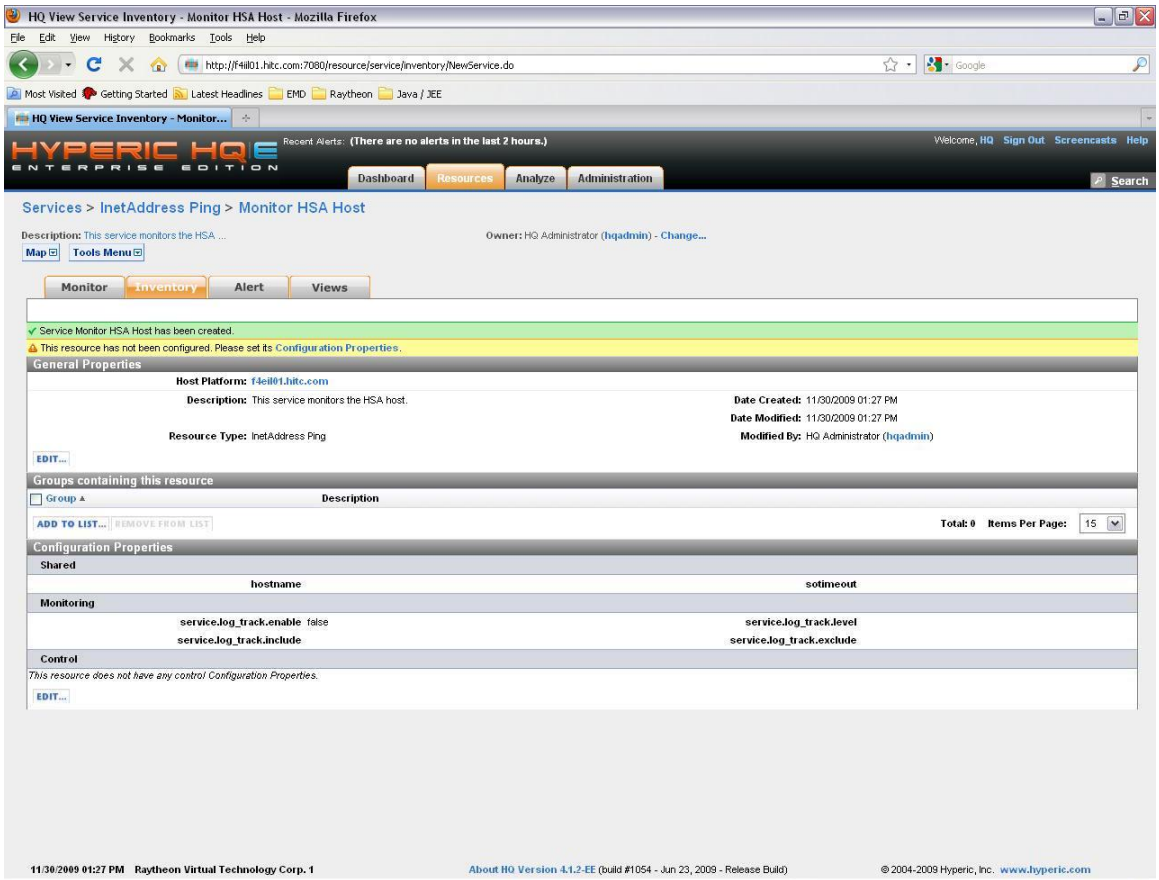

Figure 4.2.2.4.4-31. HSA Configuration Properties Link

5. In the configuration properties page fill in the hostname field. When finish click the ‘OK’ button.

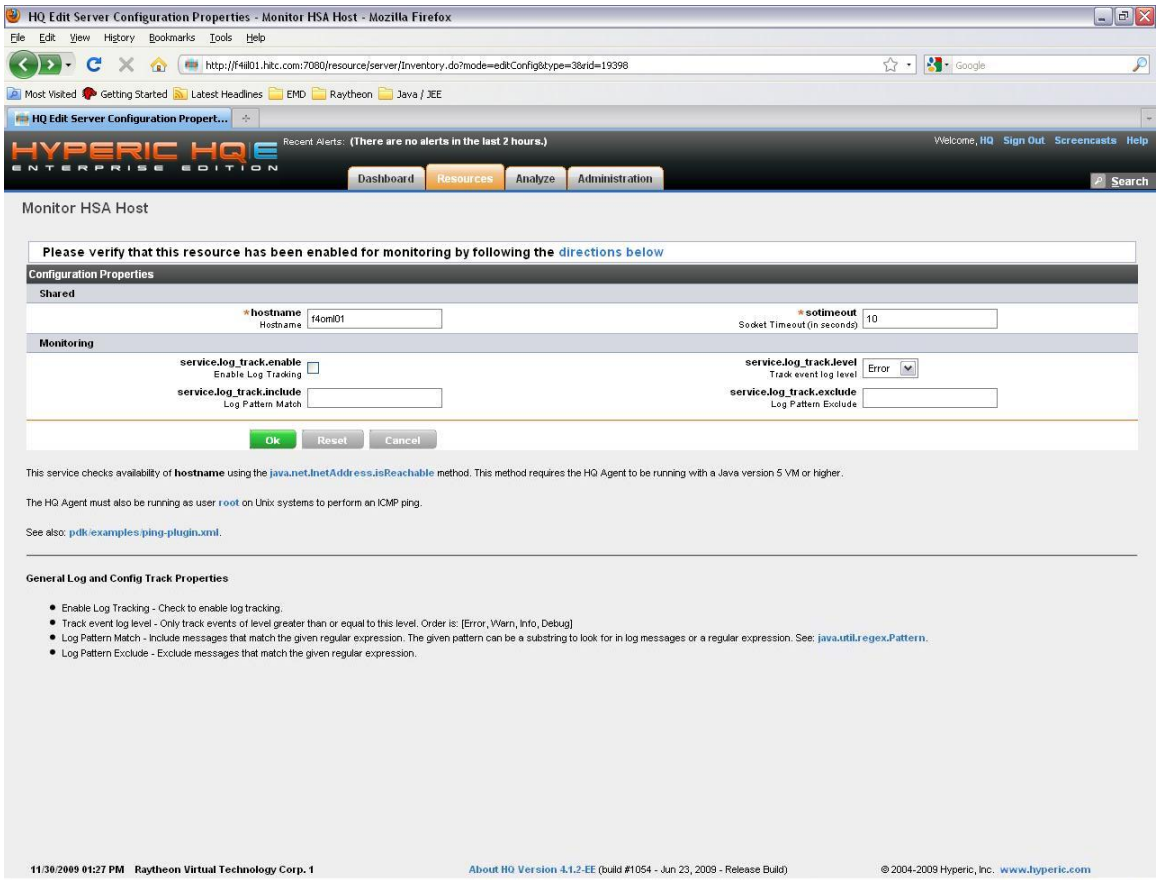

Figure 4.2.2.4.4-32. HSA Configuration Properties Page

The HSA Host InetAddress Ping service is now configured and you can go ahead and monitor it, setup alerts, and change the monitoring interval of the metrics it gathers

4.2.2.5 ECHO Resources

This section describes how to configure Hyperic to monitor the ECHO website, WIST, ECHO API, and the ECHO FTP site.

4.2.2.5.1 ECHO Website

The URL to the ECHO website is http://www.echo.nasa.gov. To monitor a website, Hyperic provides a HTTP service that can be configured to periodically submit an http request to a specified URL. Hyperic will read the responses and determine if the website is available or not. Below are the steps to configure Hyperic to monitor the ECHO website.

1. Select a platform where you want to define your HTTP service. The agent on this platform will be in charge of submitting a http request and reading the responses. Mouse

over the ‘Resources’ tab and select the ‘Browse’ option. Click on the ‘Platforms’ link if it isn’t highlighted and select a platform.

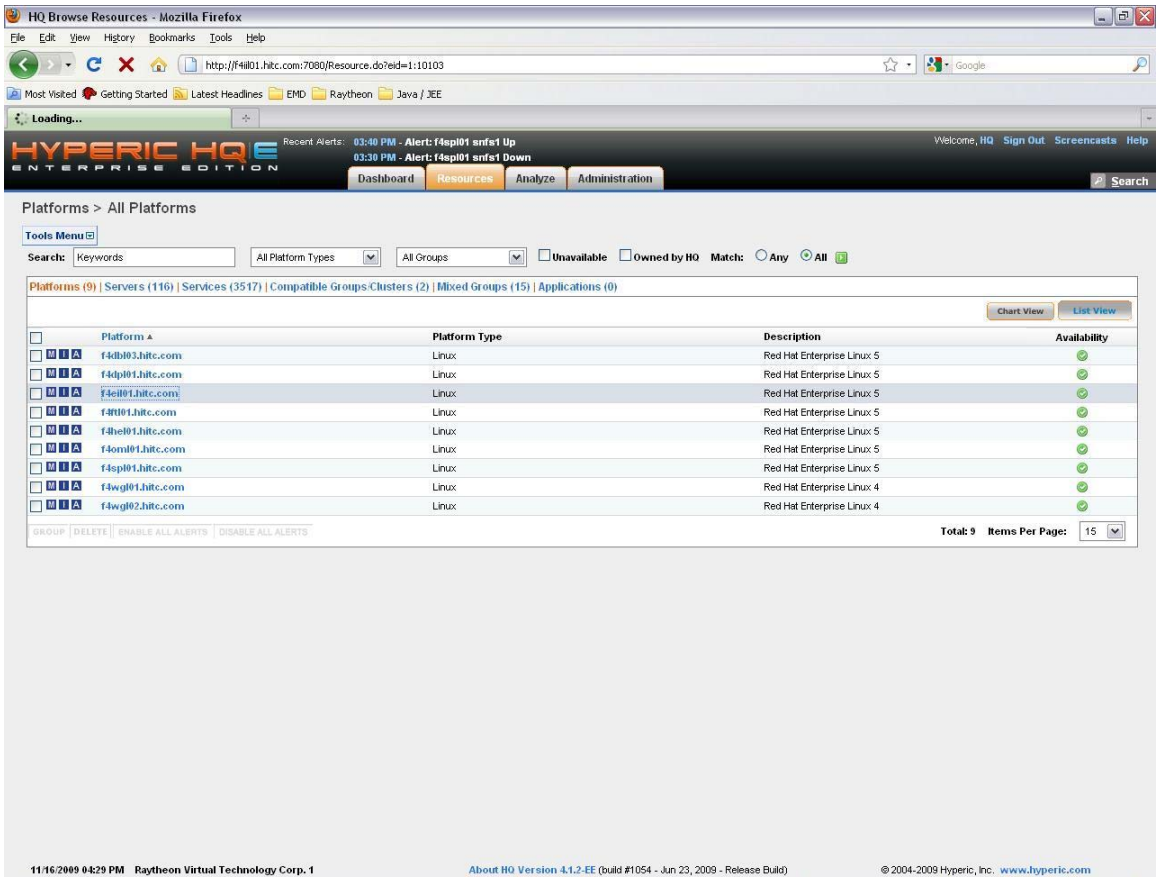

Figure 4.2.2.5.1-1. Select ECHO HTTP Service Platform

2. In the platform’s detailed page, click on the ‘Tools Menu’ button and select the ‘Add Platform Service’ option.

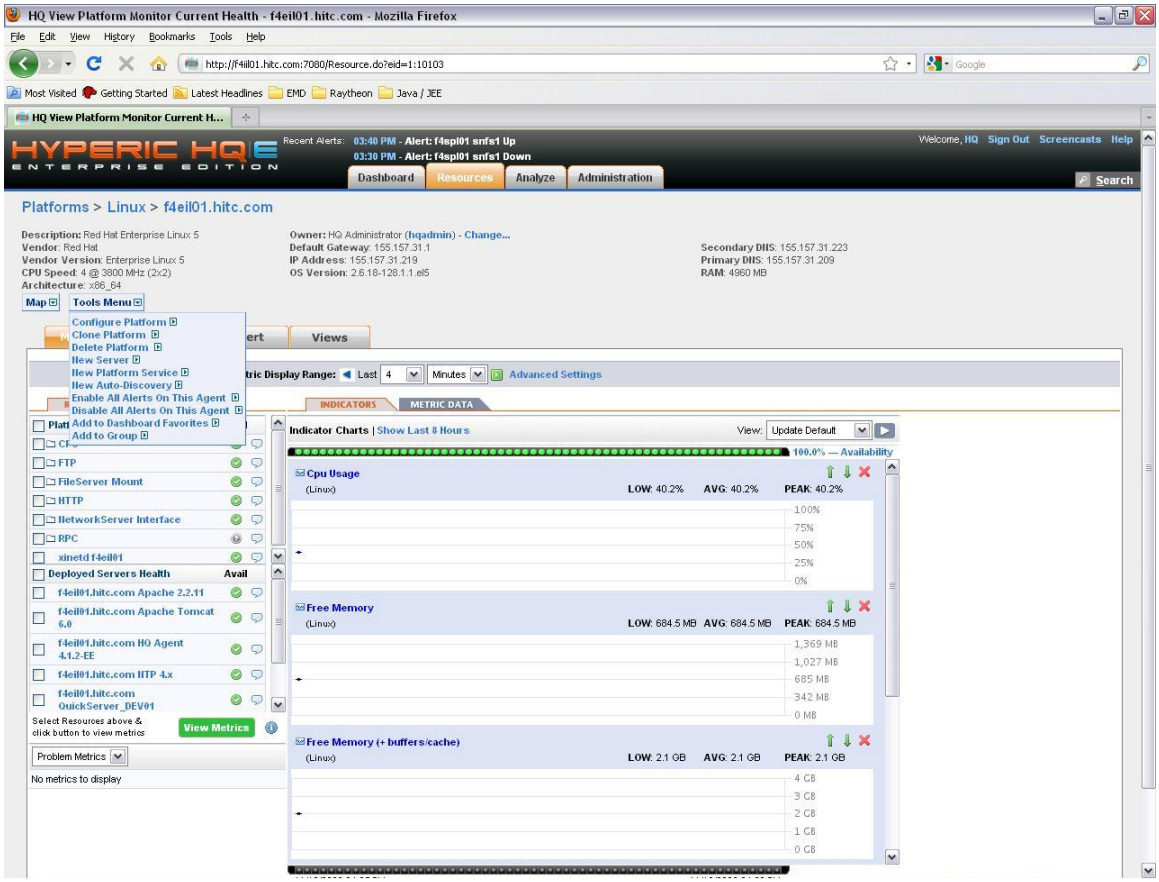

Figure 4.2.2.5.1-2. ECHO Website Add New Service

3. A new service general properties page will be displayed. Fill in the name, description, and service type. Afterward, click the 'OK' button.

   Name: ECHO Website

   Description: This service monitors the ECHO website.

   Service Type: HTTP

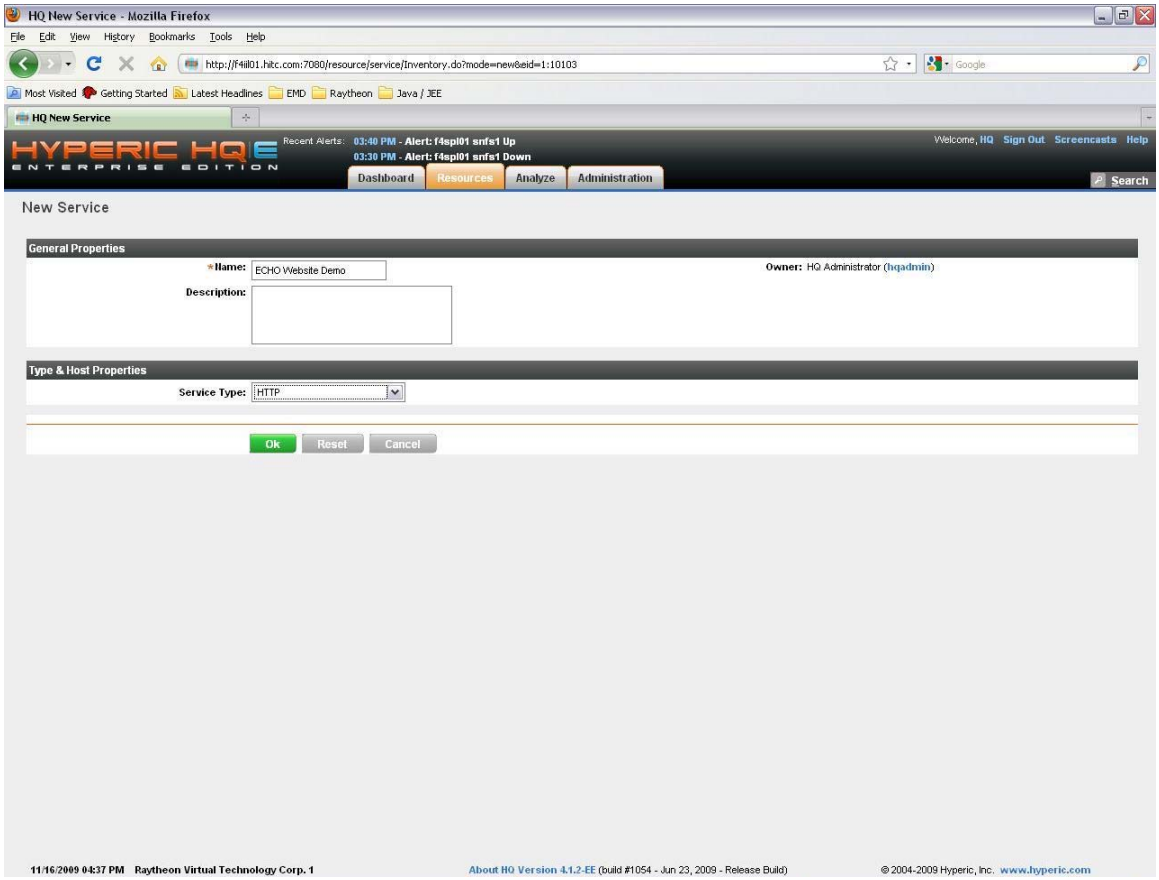

Figure 4.2.2.5.1-3. ECHO Website General Properties

4. You will be taken to the service detail page, but the service has not been configured to monitor the ECHO website yet. Click on the 'Configuration Properties' link.

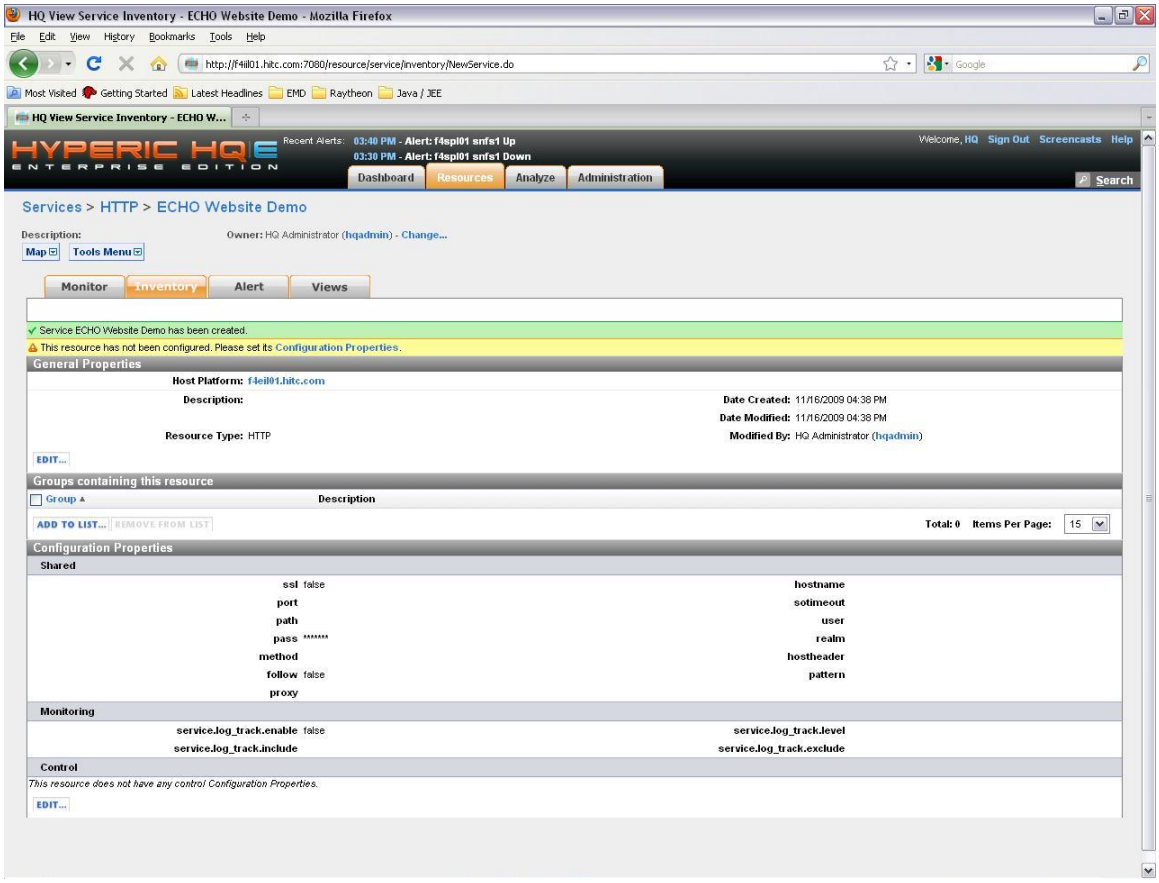

Figure 4.2.2.5.1-4. ECHO Website Configuration Properties Link

5. In the configuration properties page fill in the following and click the 'OK' button.

hostname: www.echo.nasa.gov

port: 80

*sotimeout: 10

path: /

method: GET

**proxy: <hostname>:<port>

*This is the amount of time in seconds Hyperic will timeout from the http request.

**If your internal network uses a proxy to get out supply it here.

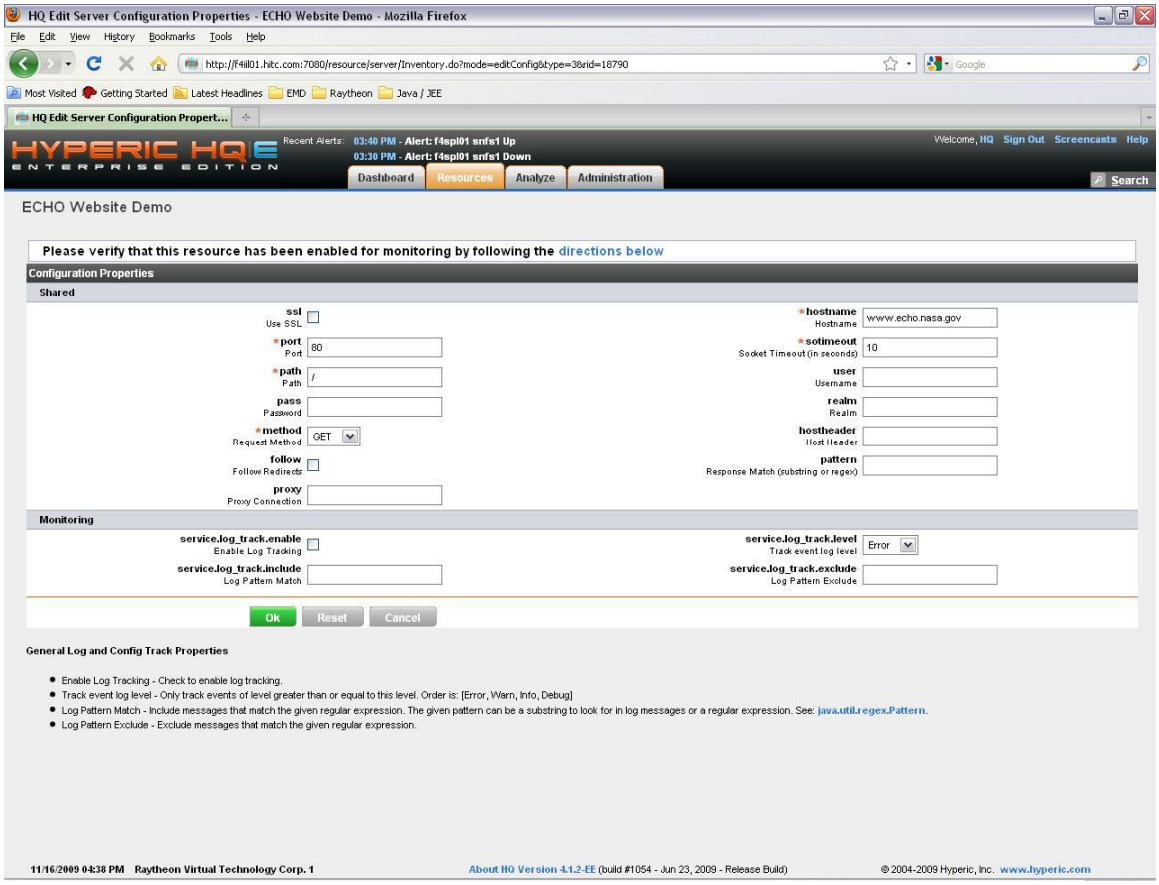

Figure 4.2.2.5.1-5. ECHO Website Configuration Properties Page

6. The ECHO Website HTTP service is now configured and you can go ahead and monitor it, setup alerts, and change the monitoring interval of the metrics it gathers.

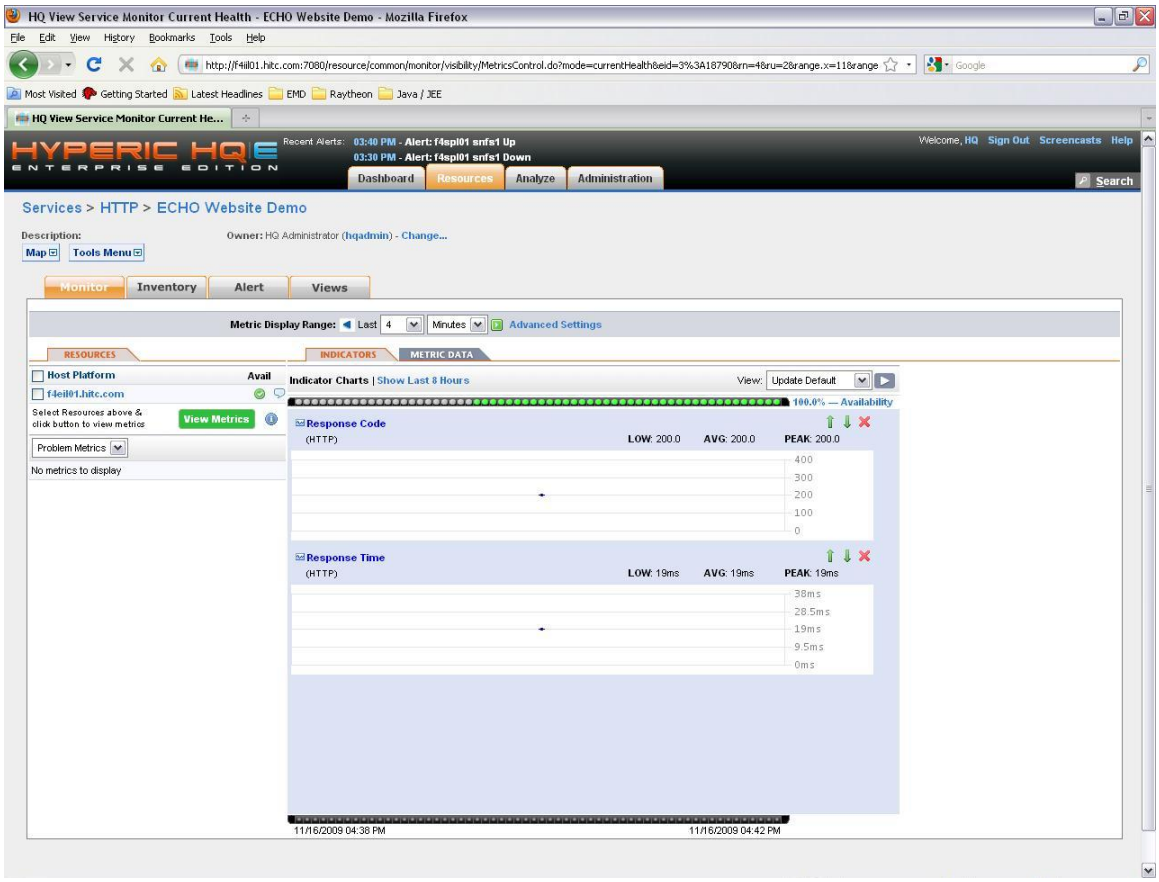

Figure 4.2.2.5.1-6. ECHO Website Monitoring

4.2.2.5.2 ECHO WIST

The URL to the ECHO WIST is https://wist.echo.nasa.gov. To monitor a website, Hyperic provides a HTTP service that can be configured to periodically submit an http request to a specified URL. Hyperic will read the responses and determine if the website is available or not. Below are the steps to configure Hyperic to monitor ECHO WIST.

1. Select a platform where you want to define your HTTP service. The agent on this platform will be in charge of submitting an http request and reading the responses. Mouse over the ‘Resources’ tab and select the ‘Browse’ option. Click on the ‘Platforms’ link if it isn’t highlighted and select a platform.

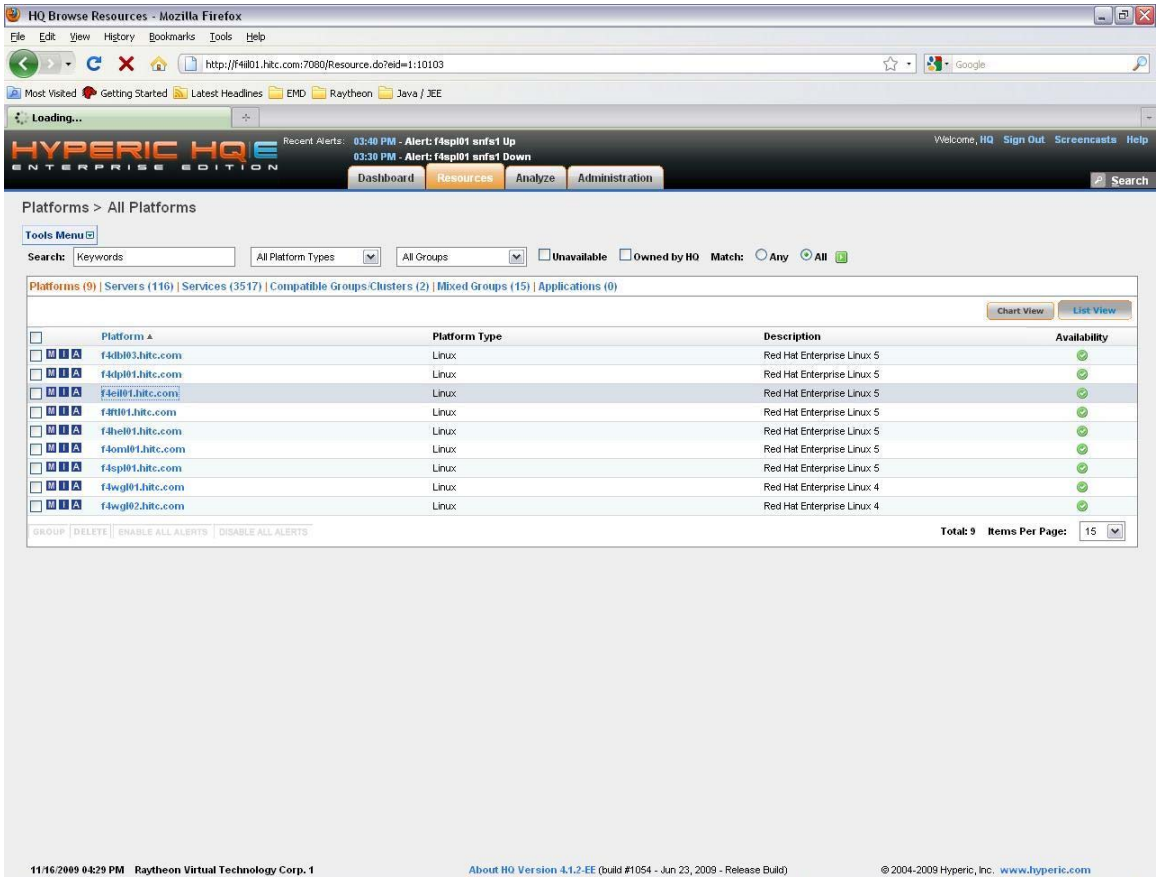

Figure 4.2.2.5.2-1. Select ECHO WIST Service Platform

2. In the platform's detailed page, click on the 'Tools Menu' button and select the 'Add Platform Service' option.

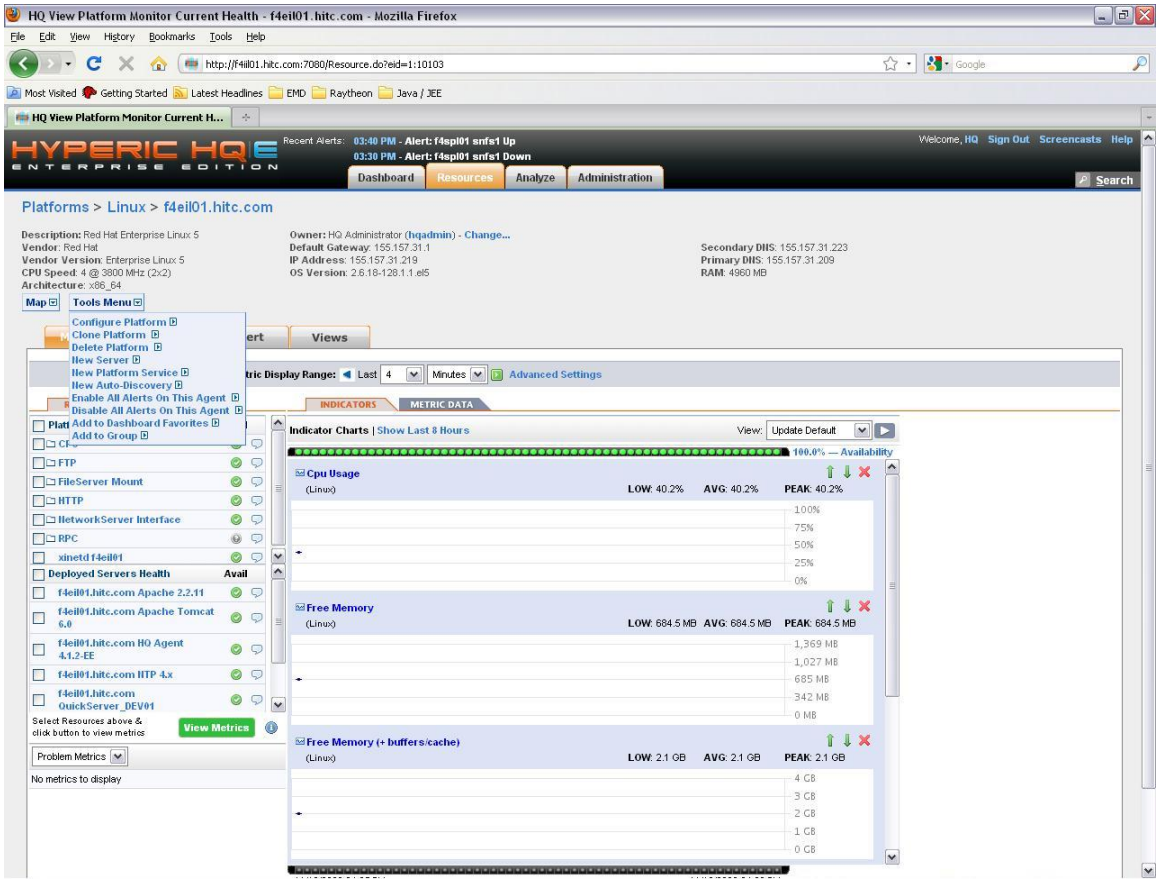

Figure 4.2.2.5.2-2. ECHO WIST Add New Service

3. A new service general properties page will be displayed. Fill in the name, description, and service type. Afterward, click the 'OK' button.

   Name: ECHO WIST

   Description: This service monitors ECHO WIST.

   Service Type: HTTP

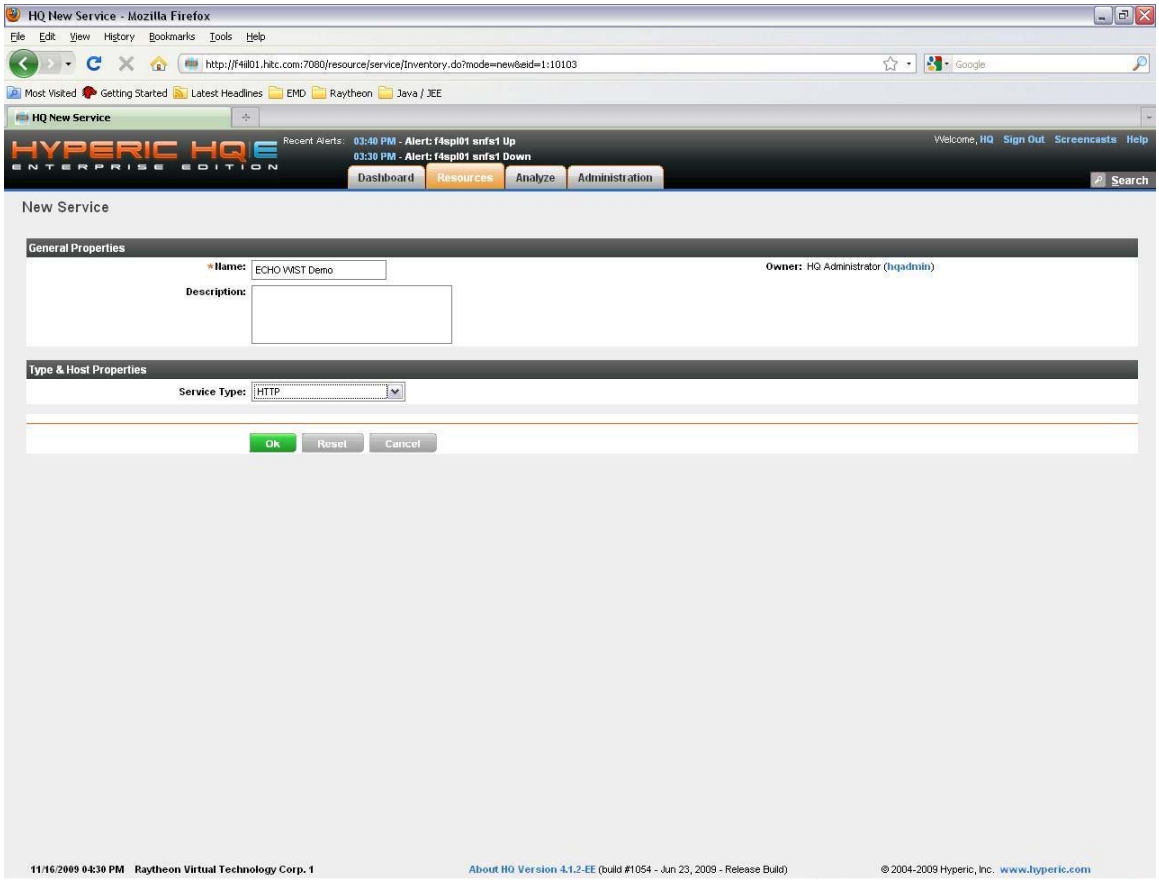

Figure 4.2.2.5.2-3. ECHO WIST General Properties

4. You will be taken to the service detail page, but the service has not been configured to monitor ECHO WIST yet. Click on the ‘Configuration Properties’ link.

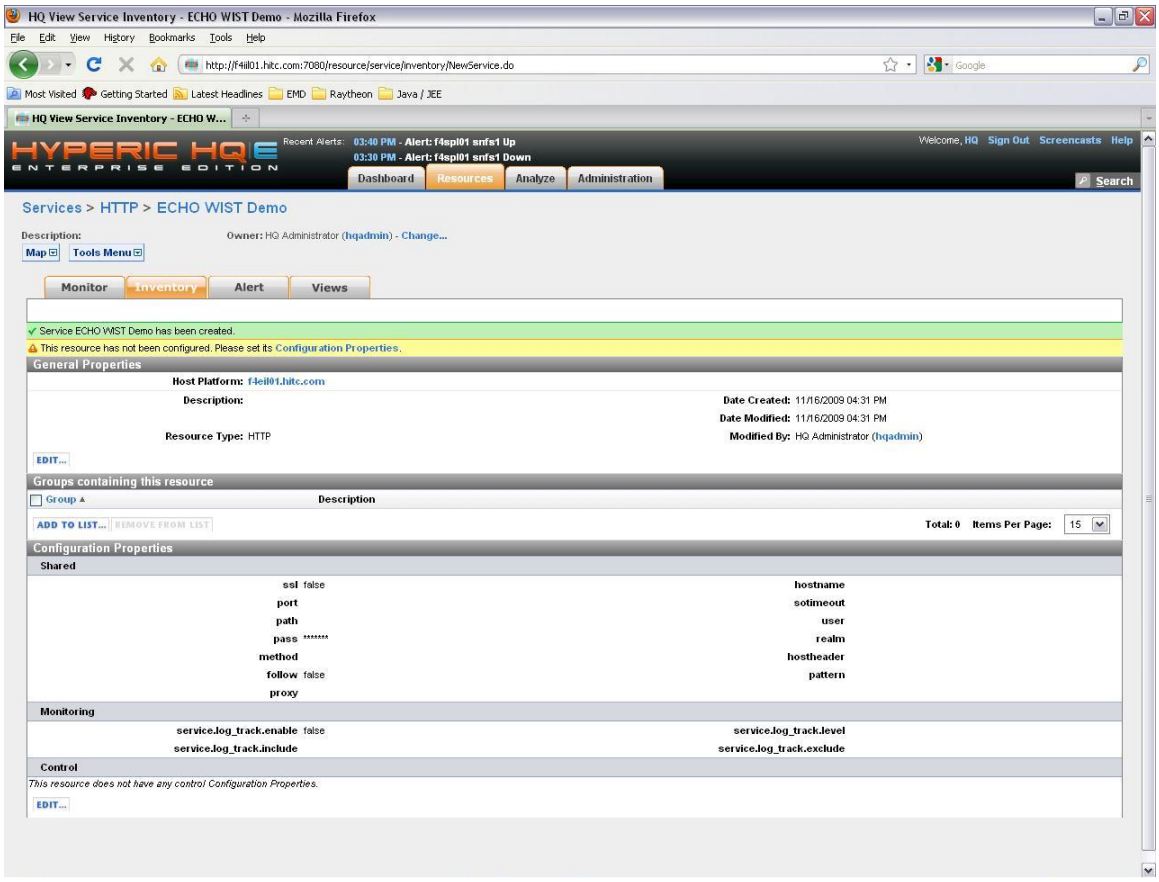

Figure 4.2.2.5.2-4. ECHO WIST Configuration Properties Link

5. In the configuration properties page fill in the following and click the ‘OK’ button.

   ssl: check the checkbox

   hostname: wist.echo.nasa.gov

   port: 443

   *sotimeout: 10

   path: /

   method: GET

   follow: check the checkbox

   **proxy: <hostname>:<port>

   *This is the amount of time in seconds Hyperic will timeout from the http request.

**If your internal network uses a proxy to get out supply it here.

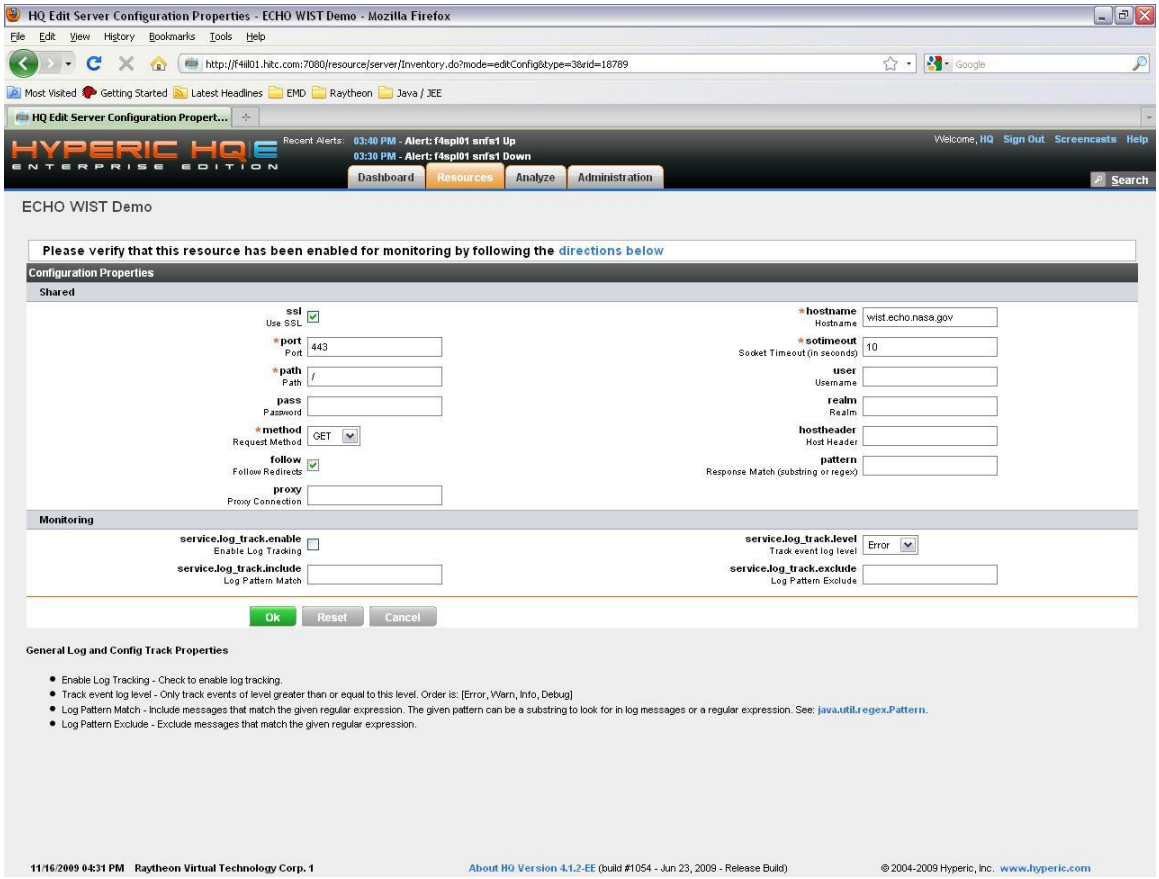

Figure 4.2.2.5.2-5. ECHO WIST Configuration Properties Page

6. The ECHO WIST HTTP service is now configured and you can go ahead and monitor it, setup alerts, and change the monitoring interval of the metrics it gathers.

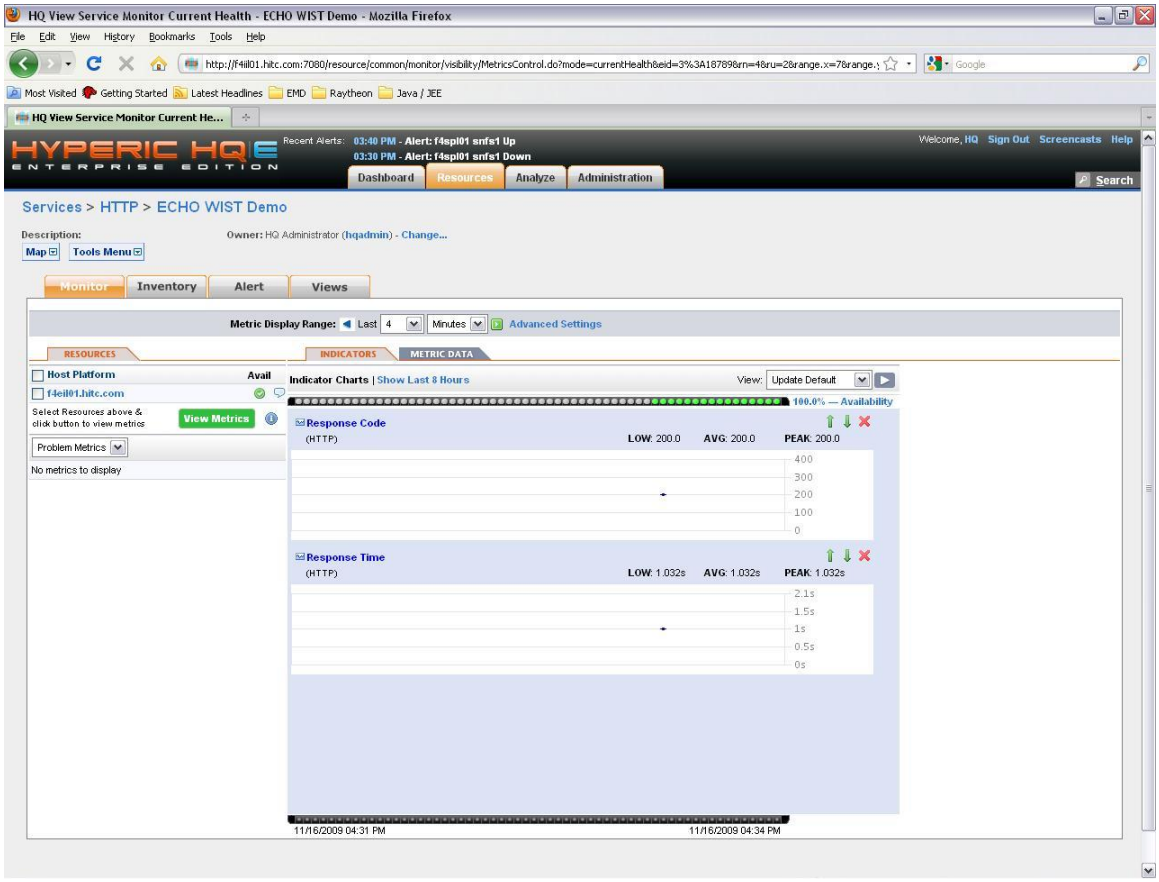

Figure 4.2.2.5.2-6. ECHO WIST Monitoring

4.2.2.5.3 ECHO API

The URL to the ECHO API is http://api.echo.nasa.gov/echo/apis.html. To monitor the API site, Hyperic provides a HTTP service that can be configured to periodically submit an http request to a specified URL. Hyperic will read the responses and determine if the site is available or not. Below are the steps to configure Hyperic to monitor the ECHO API.

1. Select a platform where you want to define your HTTP service. The agent on this platform will be in charge of submitting an http request and reading the responses. Mouse over the ‘Resources’ tab and select the ‘Browse’ option. Click on the ‘Platforms’ link if it isn’t highlighted and select a platform.

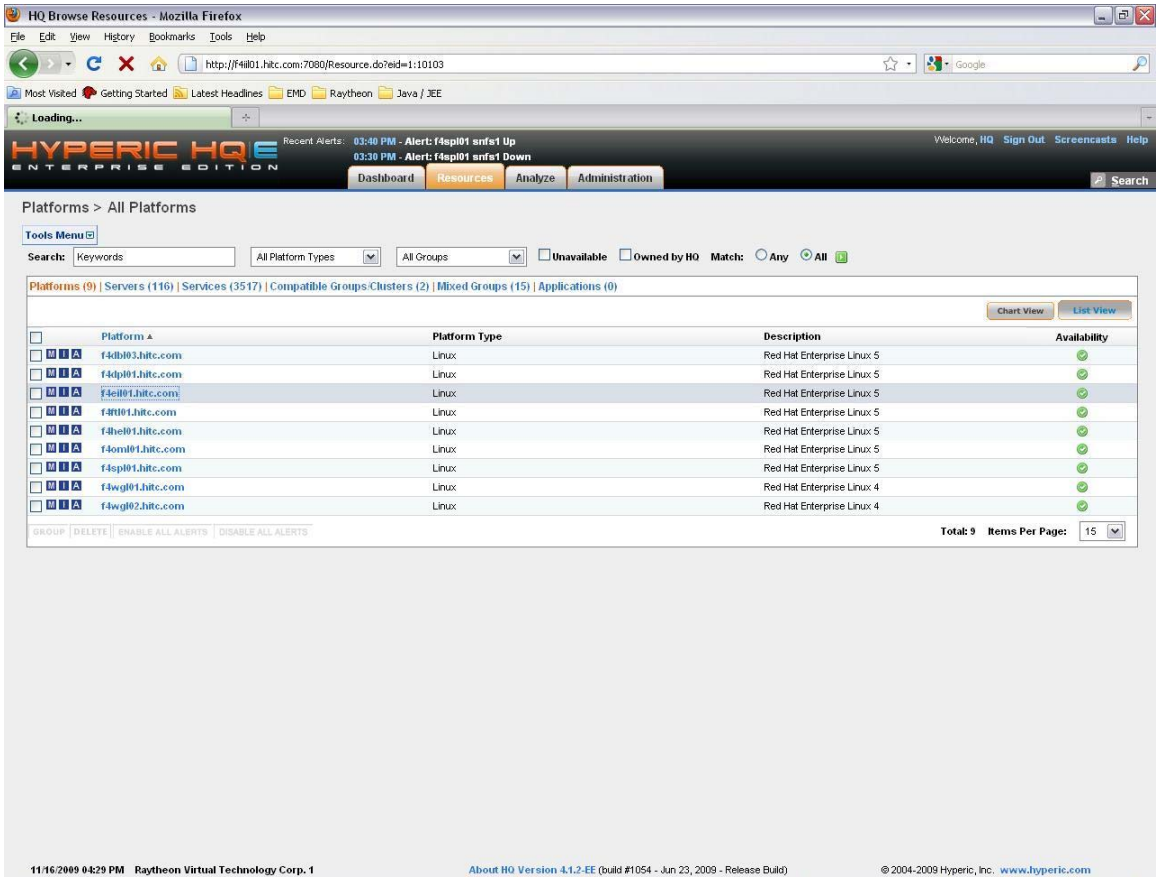

Figure 4.2.2.5.3-1. Select ECHO WIST Service Platform

2. In the platform's detailed page, click on the 'Tools Menu' button and select the 'Add Platform Service' option.

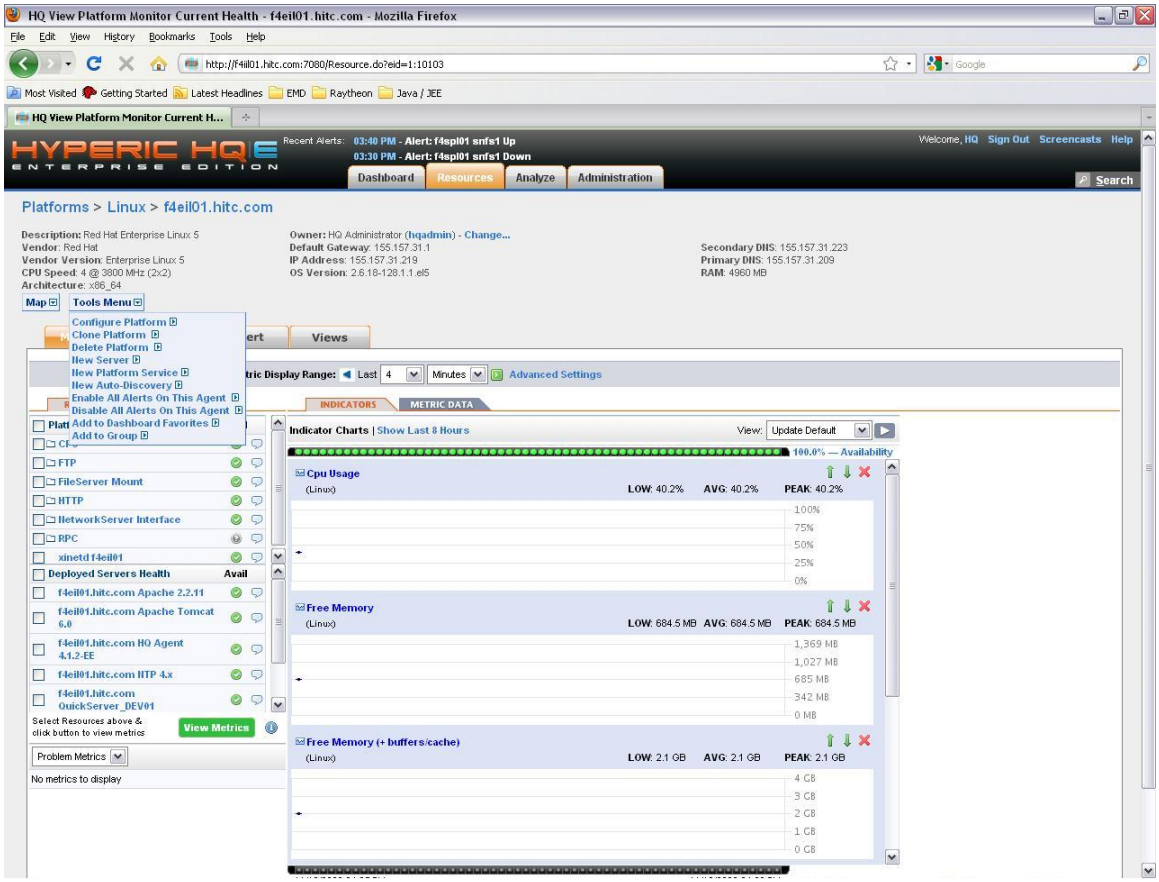

Figure 4.2.2.5.3-2. ECHO WIST Add New Service

3. A new service general properties page will be displayed. Fill in the name, description, and service type. Afterward, click the 'OK' button.

   Name: ECHO API

   Description: This service monitors ECHO API.

   Service Type: HTTP

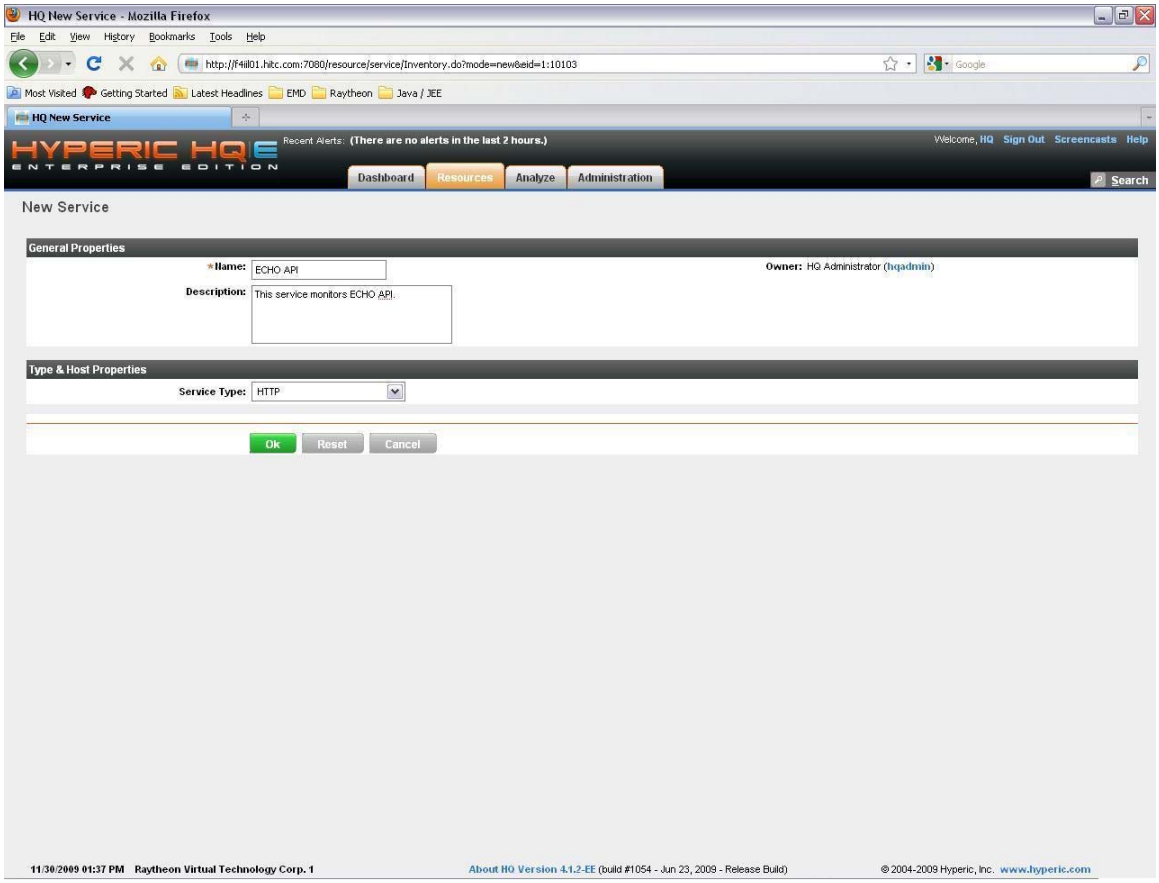

Figure 4.2.2.5.3-3. ECHO API General Properties

4. You will be taken to the service detail page, but the service has not been configured to monitor the ECHO API yet. Click on the 'Configuration Properties' link.

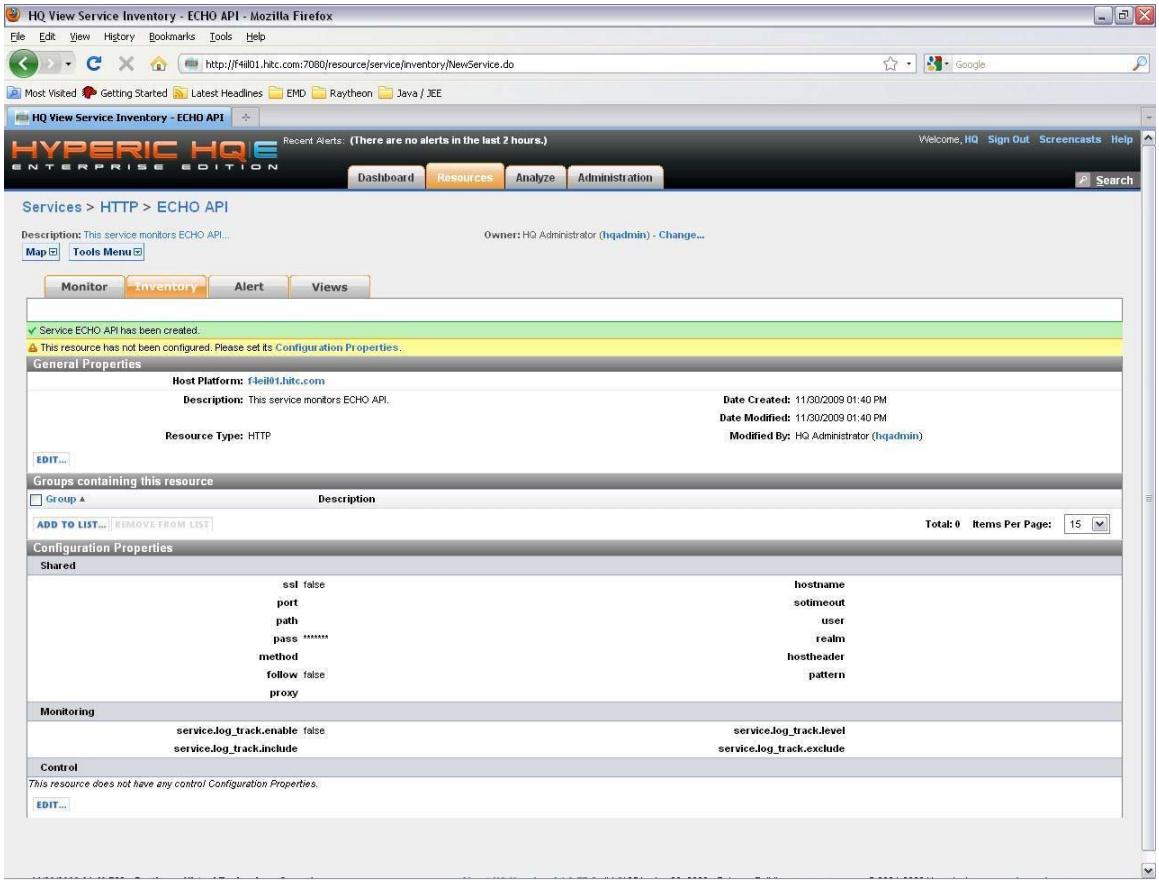

Figure 4.2.2.5.3-4. ECHO API Configuration Properties Link

5. In the configuration properties page fill in the following and click the ‘OK’ button.

   hostname: api.echo.nasa.gov

   port: 80

   *sotimeout: 60

   path: /echo/apis.html

   method: GET

   **proxy: <hostname>:<port>

   *This is the amount of time in seconds Hyperic will timeout from the http request.

   **If your internal network uses a proxy to get out supply it here.

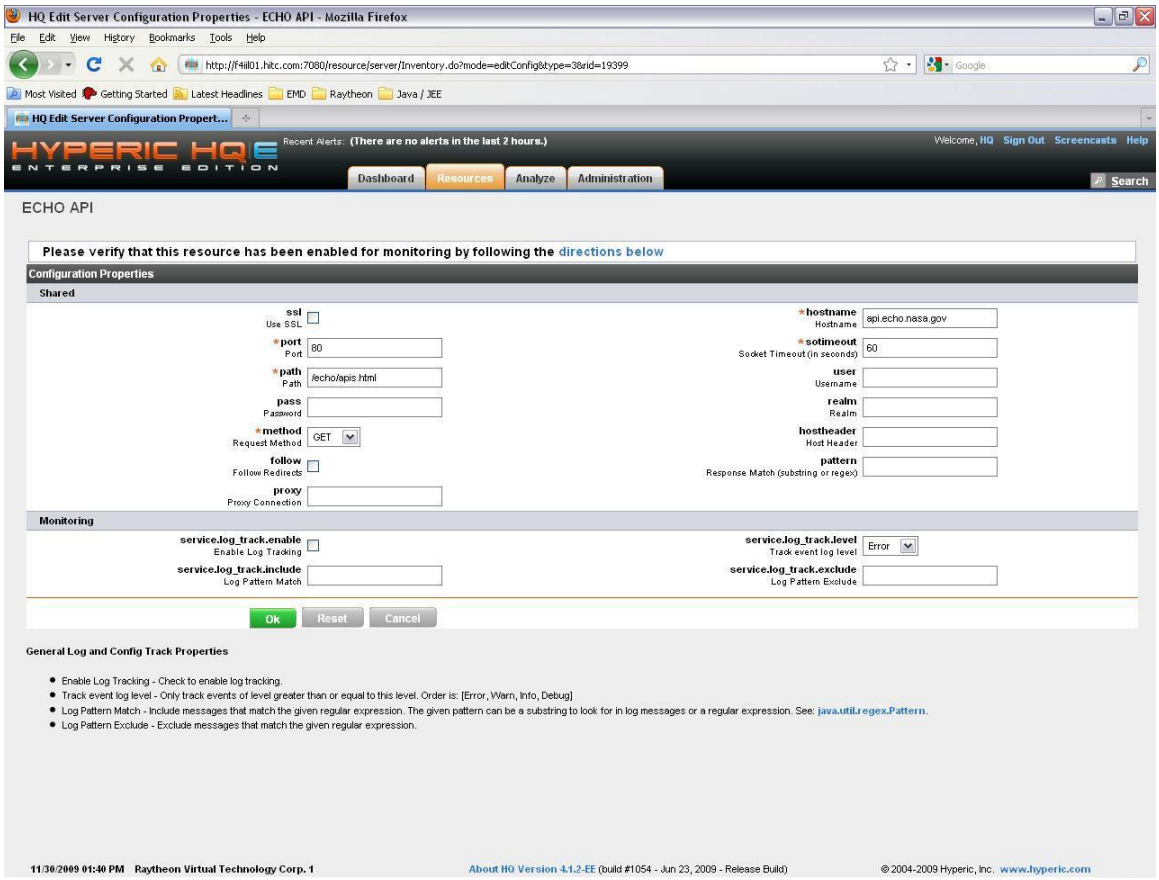

Figure 4.2.2.5.3-5. ECHO API Configuration Properties Page

6. The ECHO API HTTP service is now configured and you can go ahead and monitor it, setup alerts, and change the monitoring interval of the metrics it gathers.

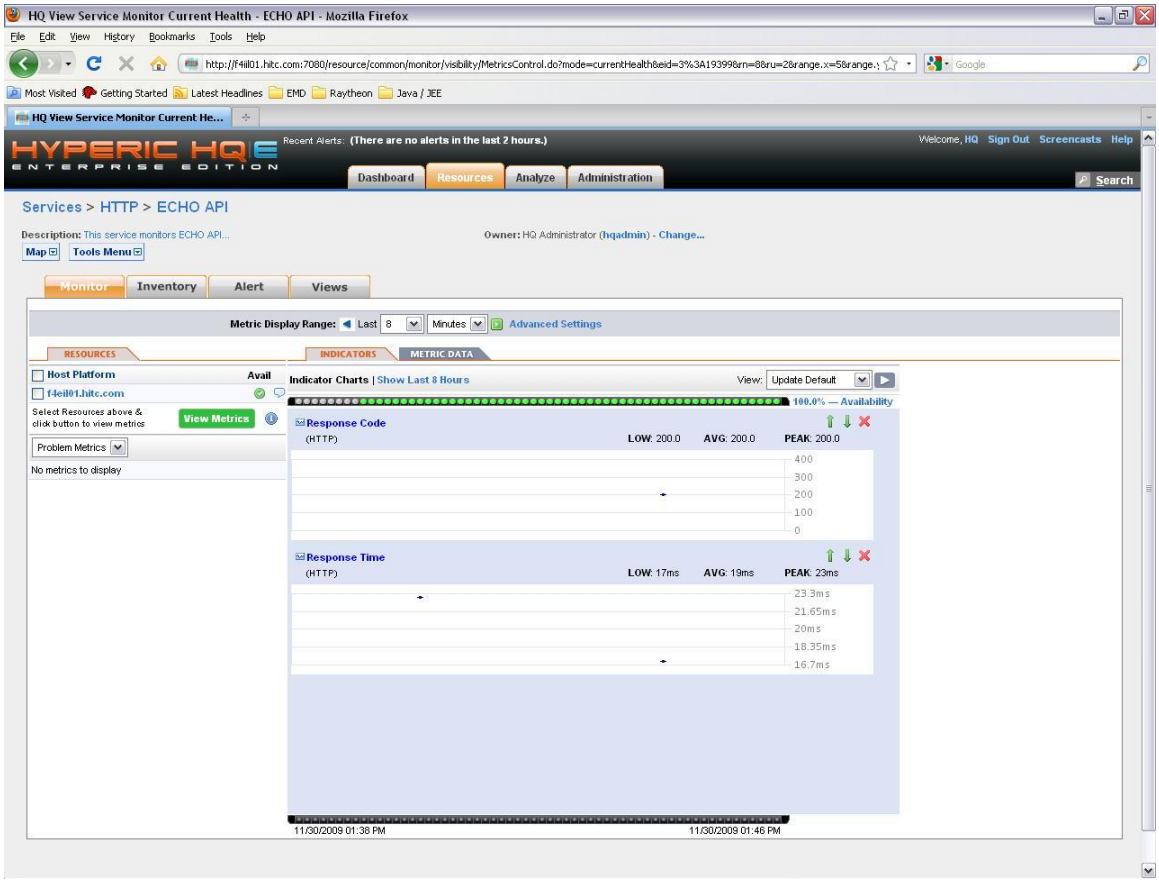

Figure 4.2.2.5.3-6. ECHO API Monitoring

4.2.2.5.4 ECHO FTP

Hyperic provides a FTP service that can be configured to periodically log into an ftp site to see whether it is up or down. This approach will be taken to monitor the ECHO FTP site. Below are the steps to configure Hyperic to perform this task.

1. Select a platform where you want to define your FTP service. The agent on this platform will be in charge of logging into the ftp site. Mouse over the 'Resources' tab and select the 'Browse' option. Click on the 'Platforms' link if it isn't highlighted and select a platform.

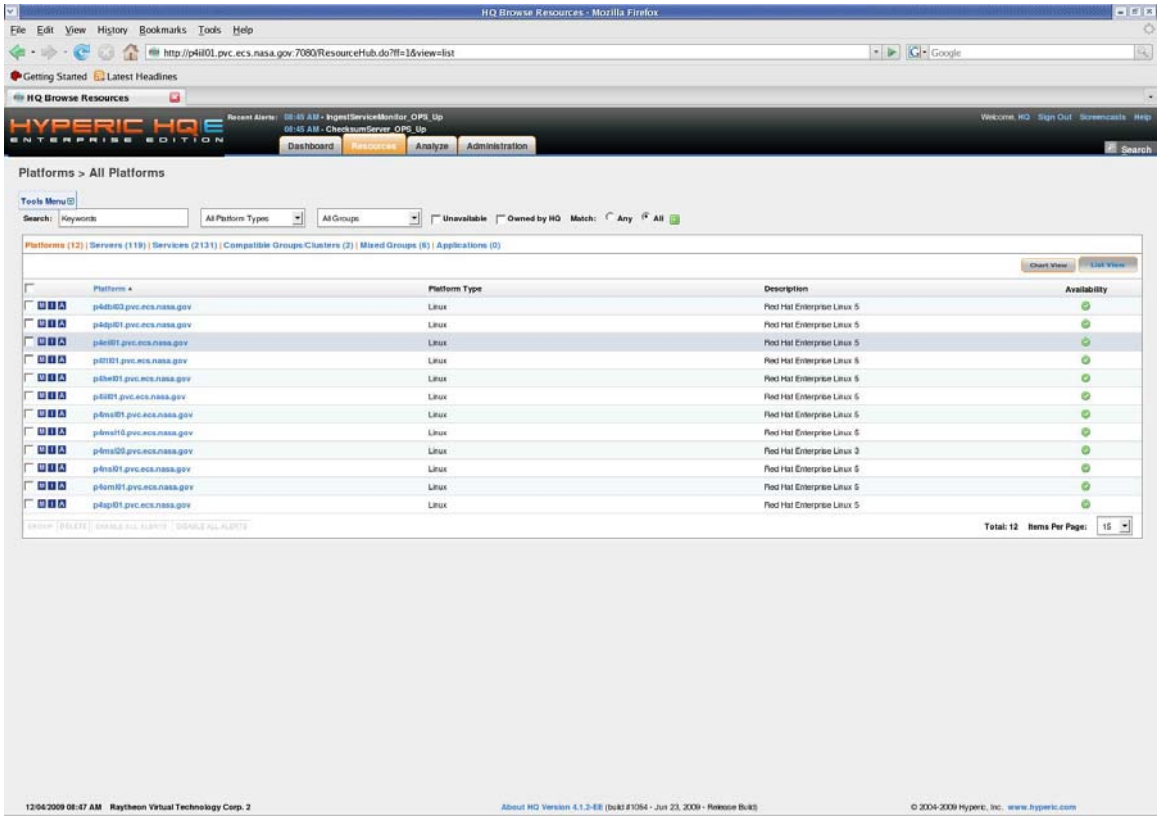

Figure 4.2.2.5.4-1. Select ECHO FTP Service Platform

2. In the platform's detailed page, click on the 'Tools Menu' button and select the 'Add Platform Service' option.

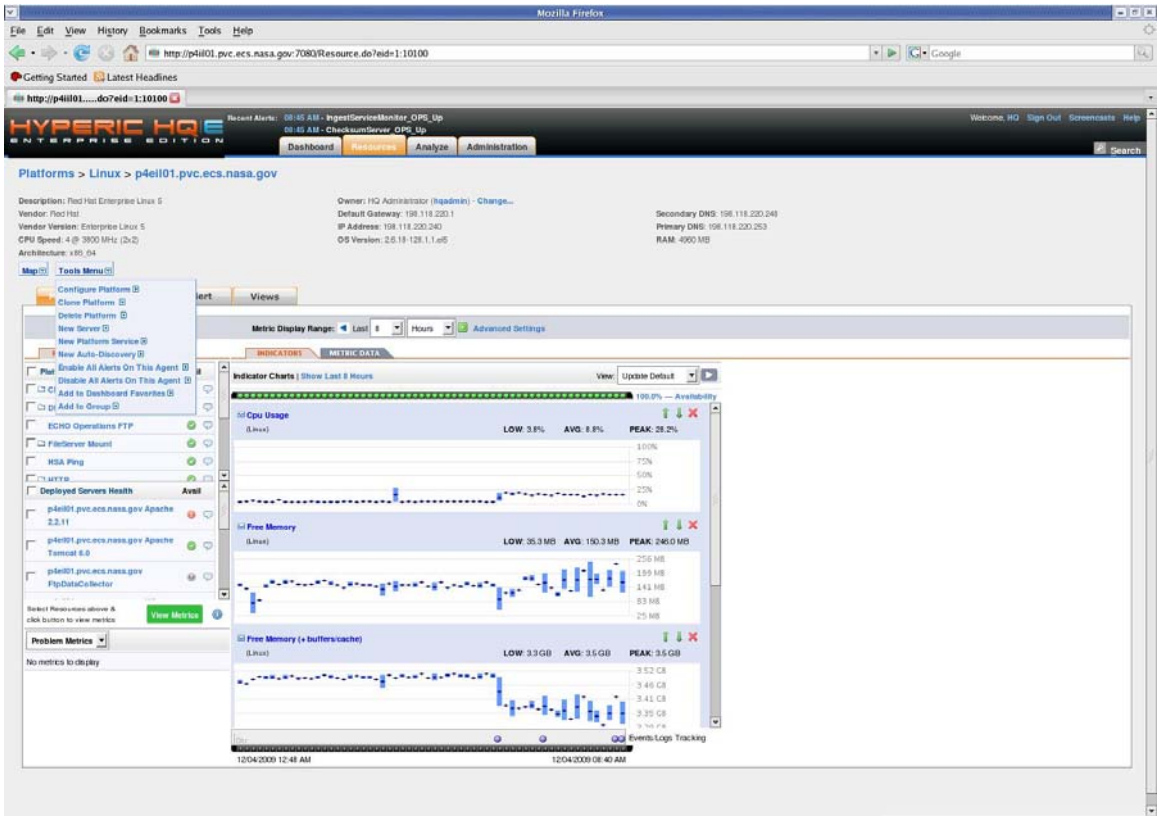

Figure 4.2.2.5.4-2. ECHO FTP Add New Service

3. A new service general properties page will be displayed. Fill in the name, description, and service type. Afterward, click the ‘OK’ button.

   Name: ECHO FTP Site

   Description: This monitors the ECHO FTP site.

   Service Type: FTP

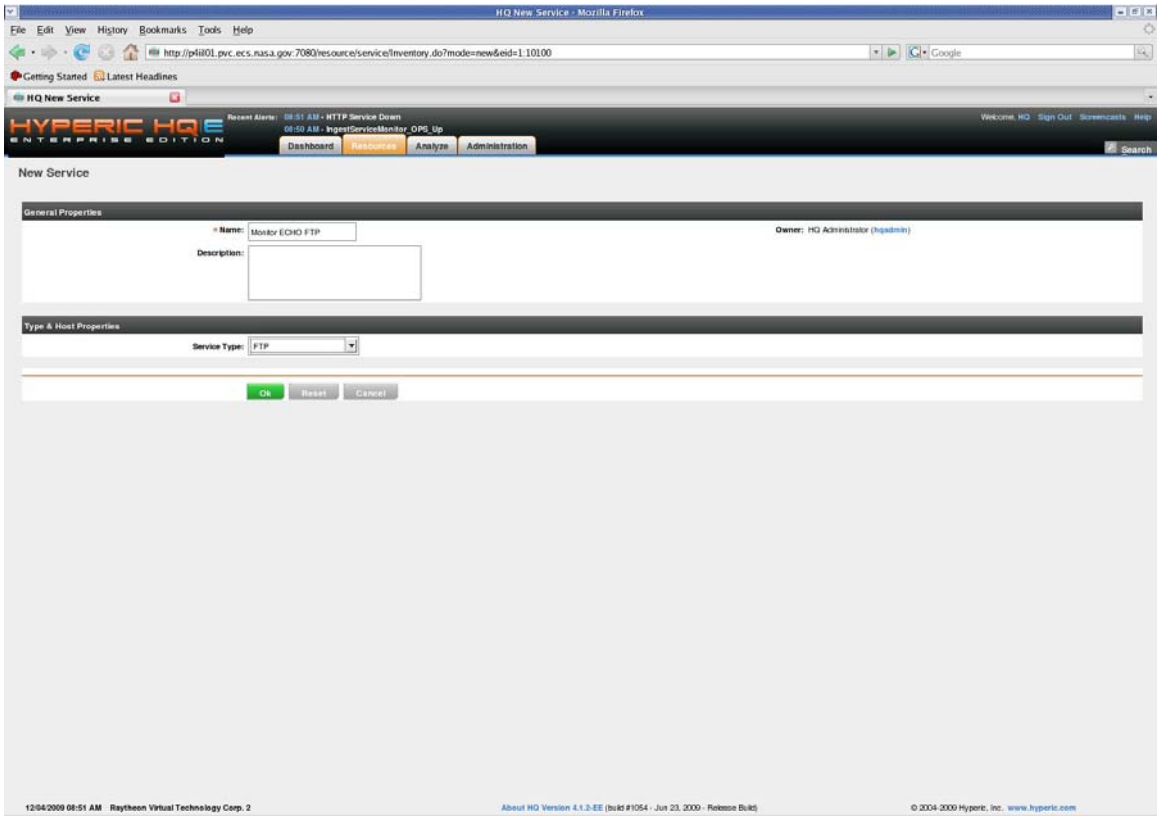

Figure 4.2.2.5.4-3. ECHO FTP General Properties

4. You will be taken to the service detail page, but the service has not been configured to monitor the ECHO FTP yet. Click on the 'Configuration Properties' link.

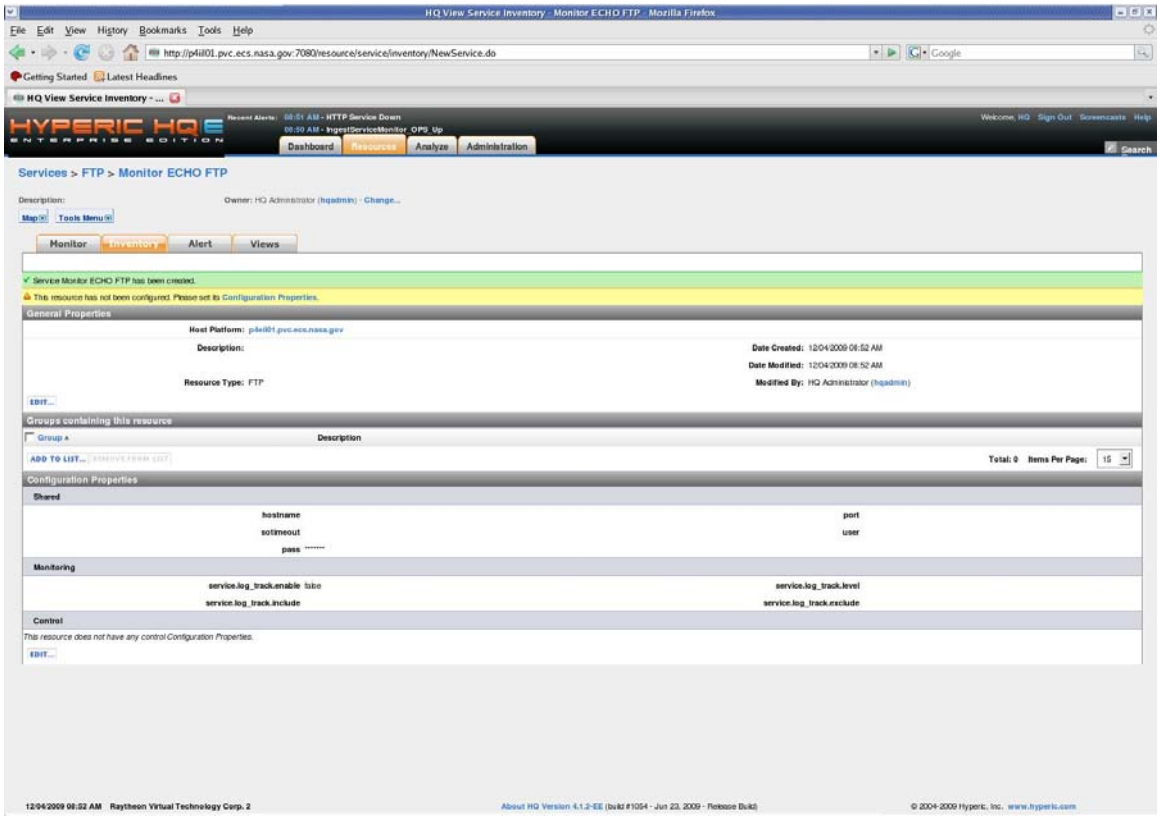

Figure 4.2.2.5.4-4. ECHO FTP Configuration Properties Link

5. In the configuration properties page fill in the following (below is an example configuration from Riverdale PVC) and click the 'OK' button.

   hostname: p0fwi09.pvc.ecs.nasa.gov

   port: 21

   *sotimeout: 20

   user: pvc_ts2@ingest-test.echo.nasa.gov

   pass: <password>

   *This is the amount of time in seconds Hyperic will timeout from the ftp request.

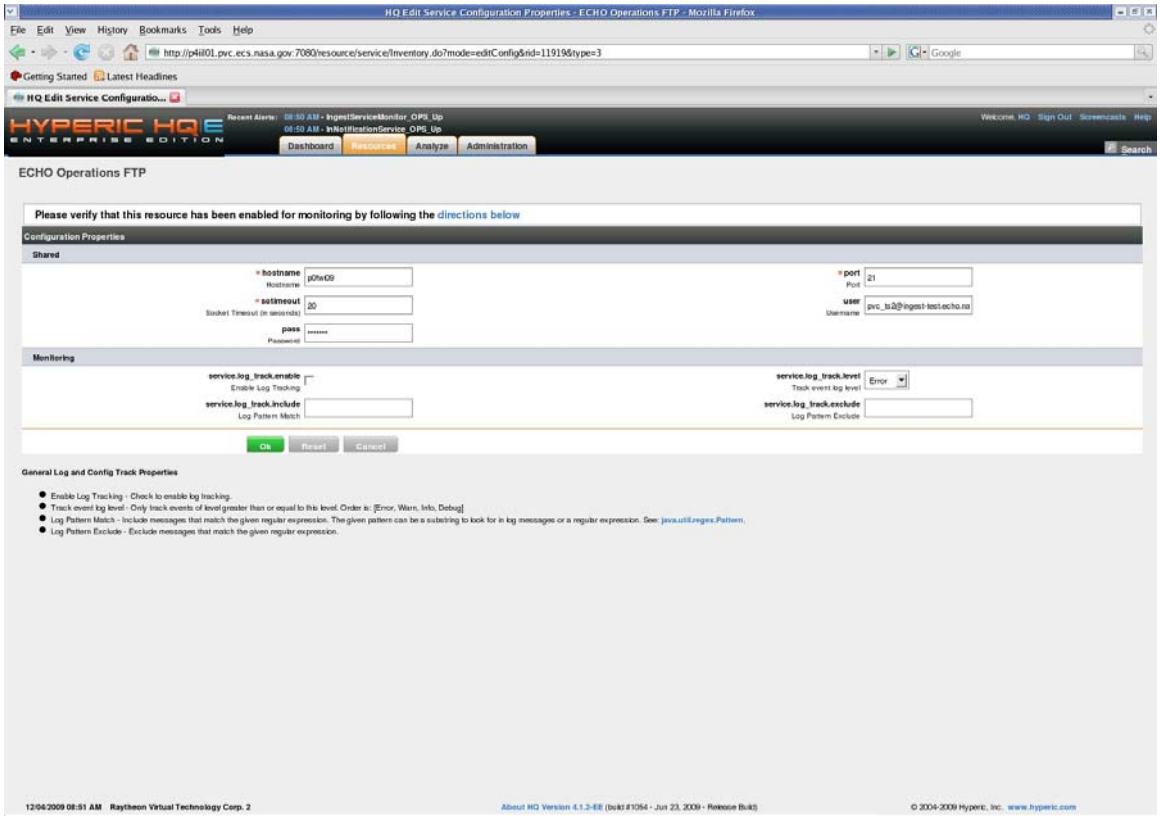

Figure 4.2.2.5.4-5. ECHO FTP Configuration Properties Page

6. The ECHO FTP service is now configured and you can go ahead and monitor it, setup alerts, and change the monitoring interval of the metrics it gathers.

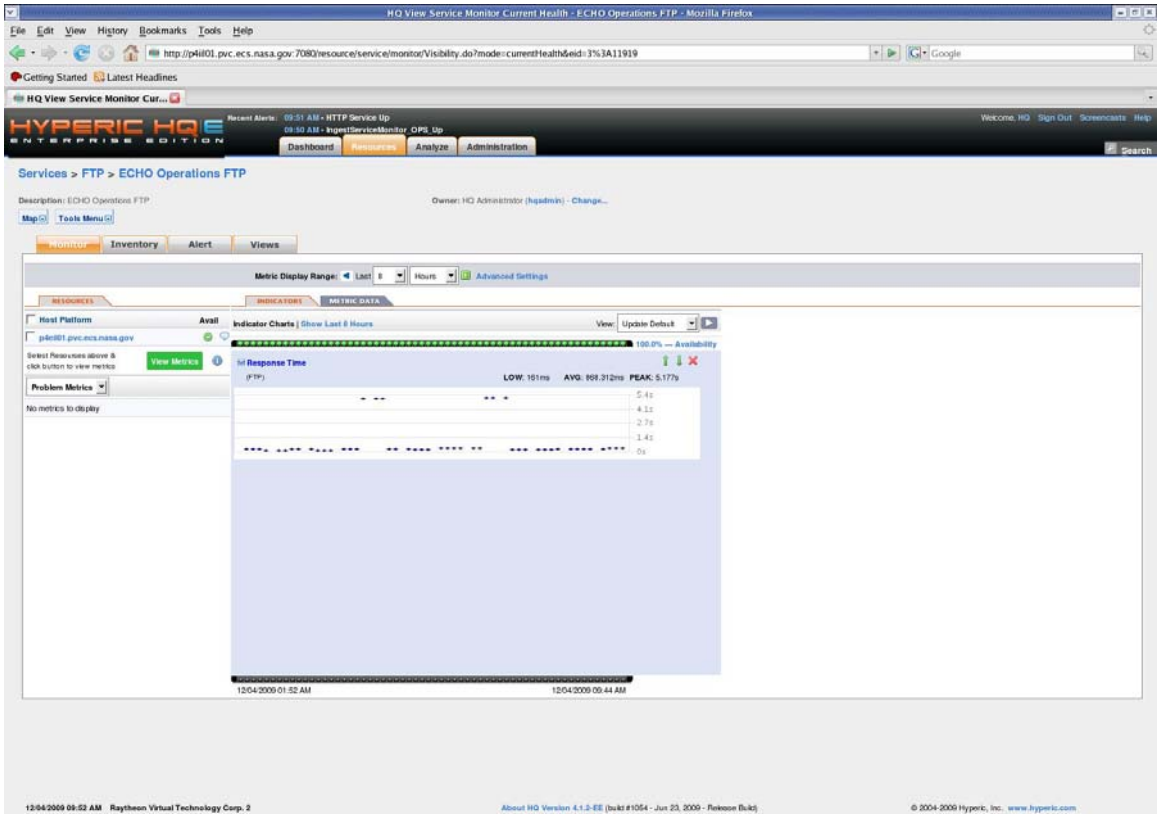

Figure 4.2.2.5.4-6. ECHO FTP Monitoring

4.2.2.6 Other Custom Resources

This section describes configure and/or monitor EEB custom resources via Hyperic that are not covered in the previous section.

4.2.2.6.1 BMGT Monitor Server

The BMGT Monitor Server monitoring is done through an xml custom plug-in. There is one plug-in file per mode. The plug-in file name is <MODE>-BMGT-Monitor-plugin.xml. The BMGT Monitor Server resource will be auto-discovered by Hyperic using the PTQL query defined in the xml plug-in.

The metrics that will be collected for the BMGT Monitor Server include the standard Hyperic metrics *(see section 4.2.2.2.13, Standard Plugin Metrics)* and custom metrics (see below).

Table 4.2.2.6.1-1. BMGT Custom Metrics

Custom Metrics	
Metric Name	**Description**
PercentIncVerComplete	The percent that Incremental Verification is complete.
NumNewPackages	The total number of new packages.
NumExportedPackages	The total number of exported packages.
NumPkgGenFailed	The total number of packages that failed Package Generation.
NumPrdtGenFailed	The total number of packages that failed Product Generation.
NumRetransmit	The total number of packages that are being retransmitted.

Up/down alerts are set up for the BMGT Monitor Server based on its availability metrics. The down alert is used to notify operator of the problem and trigger the control action that will automatically start the server; the up alert is used as a recovery alert for the down alert to mark the down alert as fixed.

Please refer to the Polling Server, section 4.2.2.4.2.3 for details on the control actions and alerts configurations. The configuration process is basically identical except the resource type, control action, script names and alert names are changed accordingly.

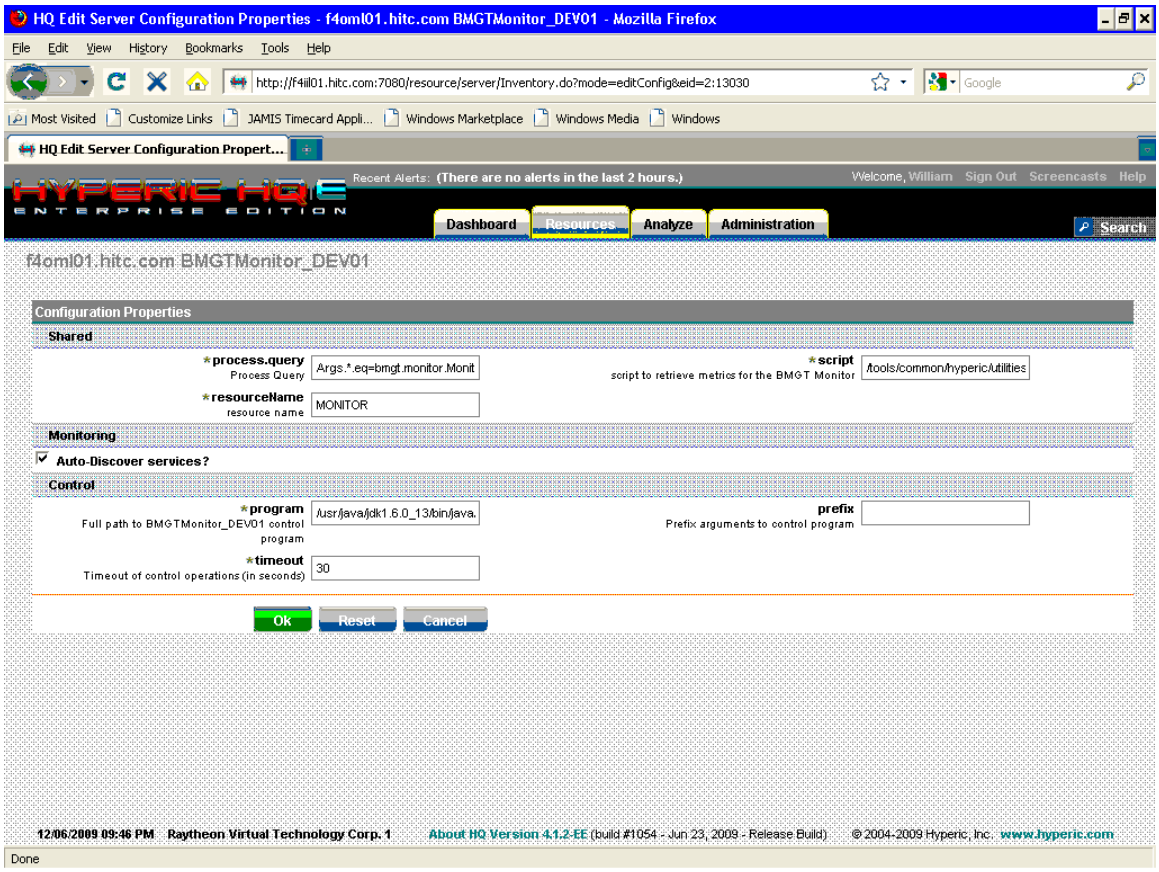

Figure 4.2.2.6.1-1. BMGT Monitor Server Configuration View

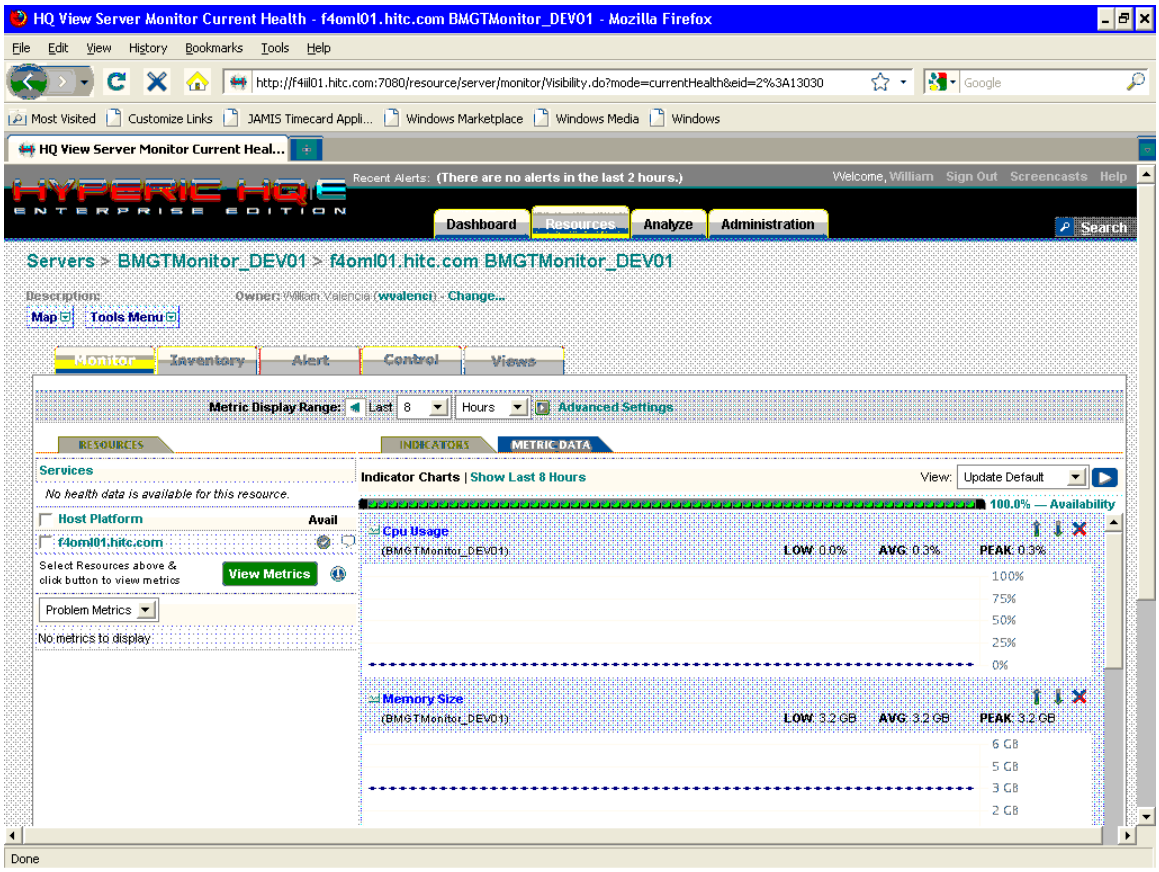

Figure 4.2.2.6.1-2. BMGT Monitor Server Monitoring

4.2.2.6.2 BMGT Generator Server

The BMGT Generator Server monitoring is done through an xml custom plug-in. There is one plug-in file per mode. The plug-in file name is <MODE>-BMGT-Generator-plugin.xml. The BMGT Generator Server resource will be auto-discovered by Hyperic using the PTQL query defined in the xml plug-in. The standard metrics will be collected by the SIGAR plug-in.

Up/down alerts are set up for the BMGT Generator Server based on its availability metrics. The down alert is used to notify operator of the problem and trigger the control action that will automatically start the server; the up alert is used as a recovery alert for the down alert to mark the down alert as fixed.

Please refer to the Polling Server, section 4.2.2.4.2.3 for details on the control actions and alerts configurations. The configuration process is basically identical except the resource type, control action, script names and alert names are changed accordingly.

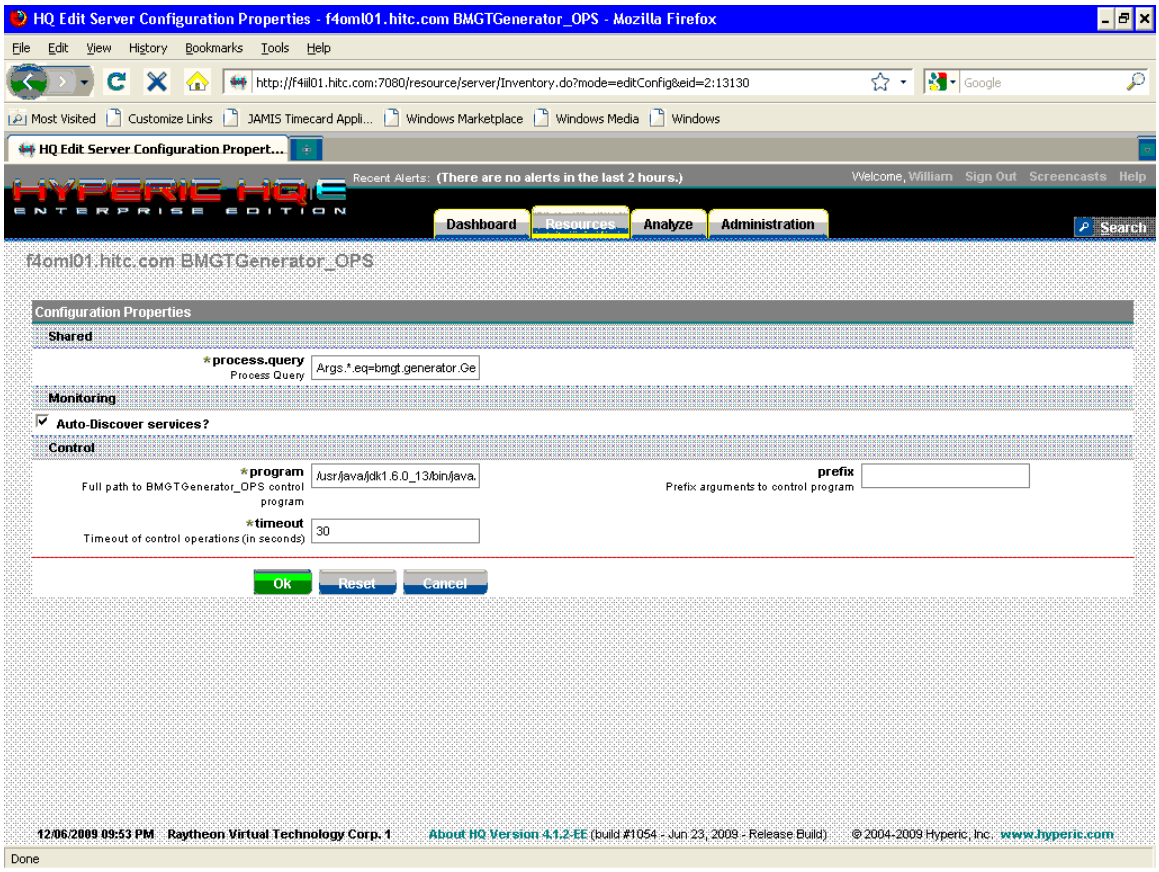

Figure 4.2.2.6.2-1. BMGT Generator Server Configuration View

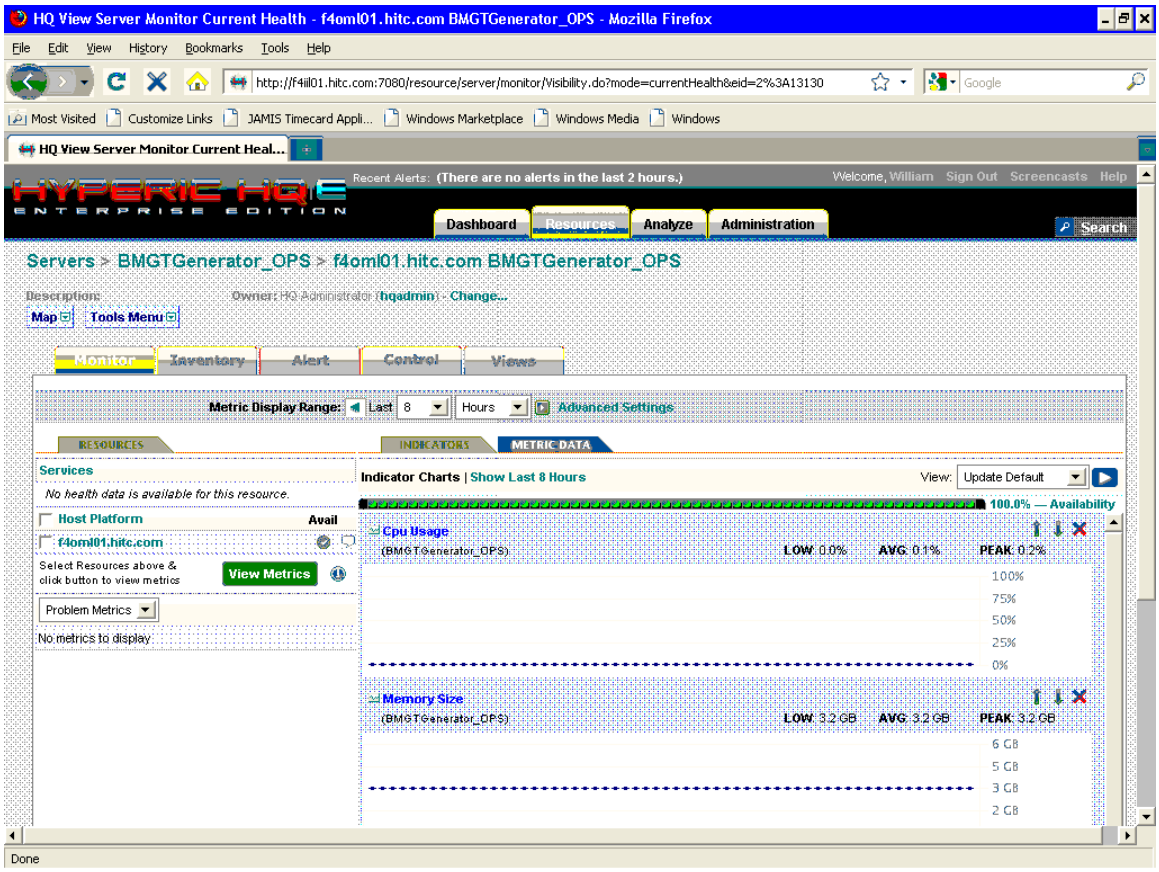

Figure 4.2.2.6.2-2. BMGT Generator Server Monitoring

4.2.2.6.3 BMGT Packager Server

The BMGT Packager Server monitoring is done through an xml custom plug-in. There is one plug-in file per mode. The plug-in file name is <MODE>-BMGT-Packager-plugin.xml. The BMGT Packager Server resource will be auto-discovered by Hyperic using the PTQL query defined in the xml plug-in. The standard metrics will be collected by the SIGAR plug-in.

Up/down alerts are set up for the BMGT Packager Server based on its availability metrics. The down alert is used to notify operator of the problem and trigger the control action that will automatically start the server; the up alert is used as a recovery alert for the down alert to mark the down alert as fixed.

Please refer to the Polling Server, section 4.2.2.4.2.3 for details on the control actions and alerts configurations. The configuration process is basically identical except the resource type, control action, script names and alert names are changed accordingly.

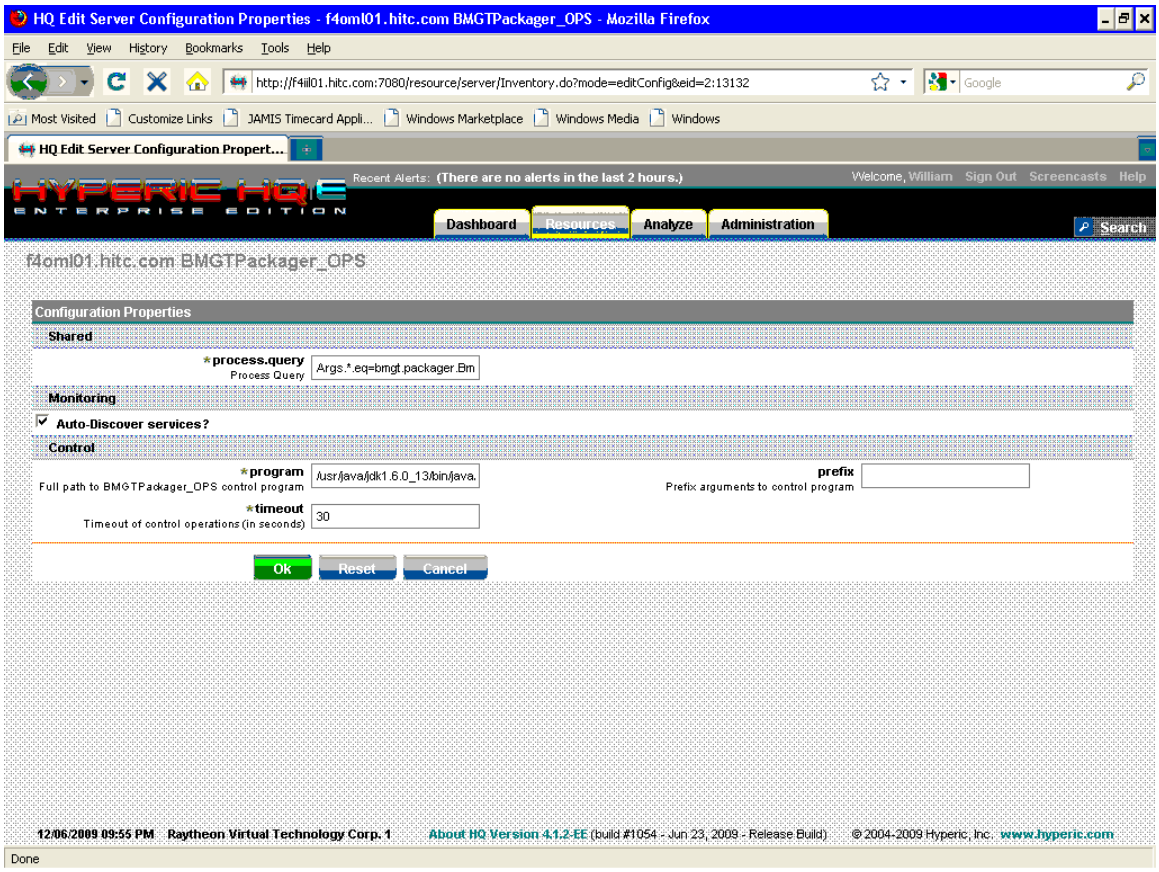

Figure 4.2.2.6.3-1. BMGT Packager Server Configuration View

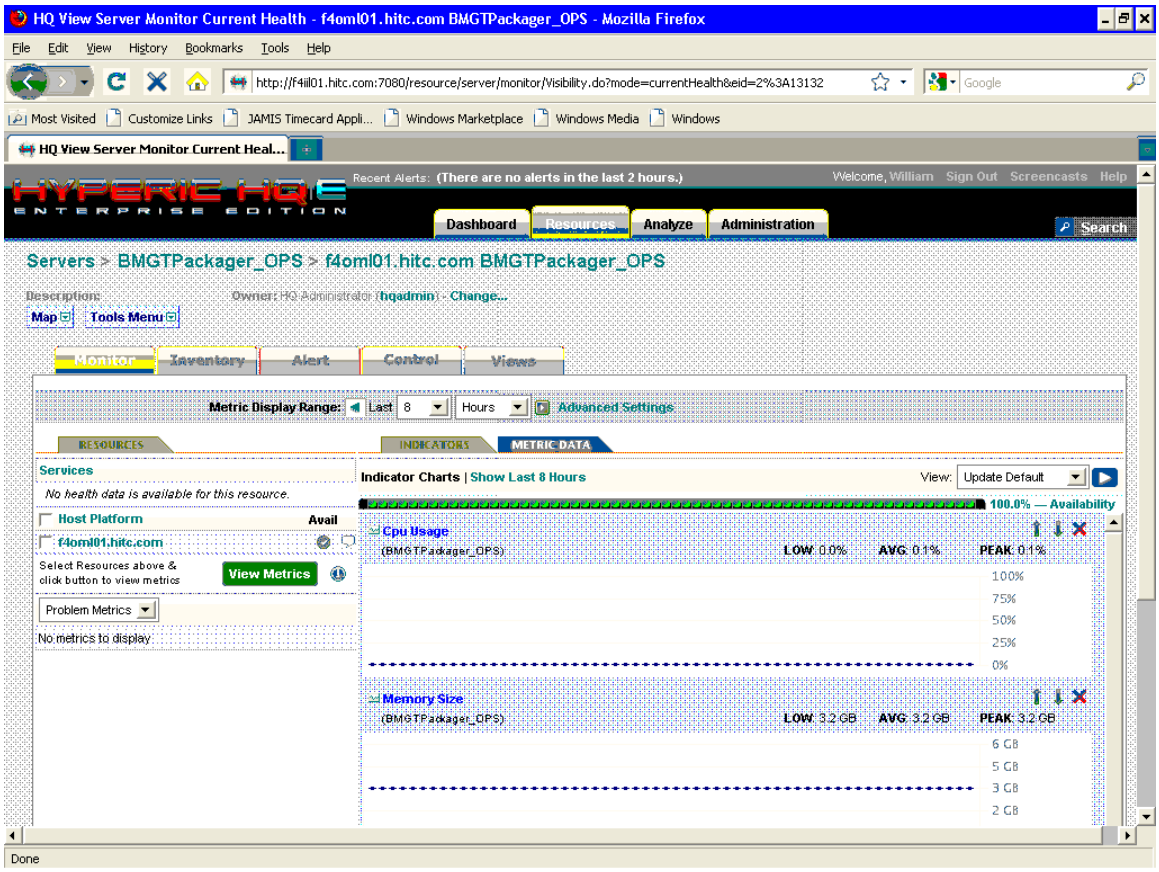

Figure 4.2.2.6.3-2. BMGT Packager Server Monitoring

4.2.2.6.4 BMGT Export Server

The BMGT Export Server monitoring is done through an xml custom plug-in. There is one plug-in file per mode. The plug-in file name is <MODE>-BMGT-ExportServer-plugin.xml. The BMGT Export Server resource will be auto-discovered by Hyperic using the PTQL query defined in the xml plug-in. The standard metrics will be collected by the SIGAR plug-in as well as the custom metric shown below.

Table 4.2.2.6.4-1. BMGT Export Server Metrics

Custom Metrics	
Metric Name	**Description**
ReexportQueueSize	The current size of the Re-export Queue

Up/down alerts are set up for the BMGT Export Server based on its availability metrics. The down alert is used to notify operator of the problem and trigger the control action that will automatically start the server; the up alert is used as a recovery alert for the down alert to mark the down alert as fixed.

Please refer to the Polling Server, section 4.2.2.4.2.3 for details on the control actions and alerts configurations. The configuration process is basically identical except the resource type, control action, script names and alert names are changed accordingly.

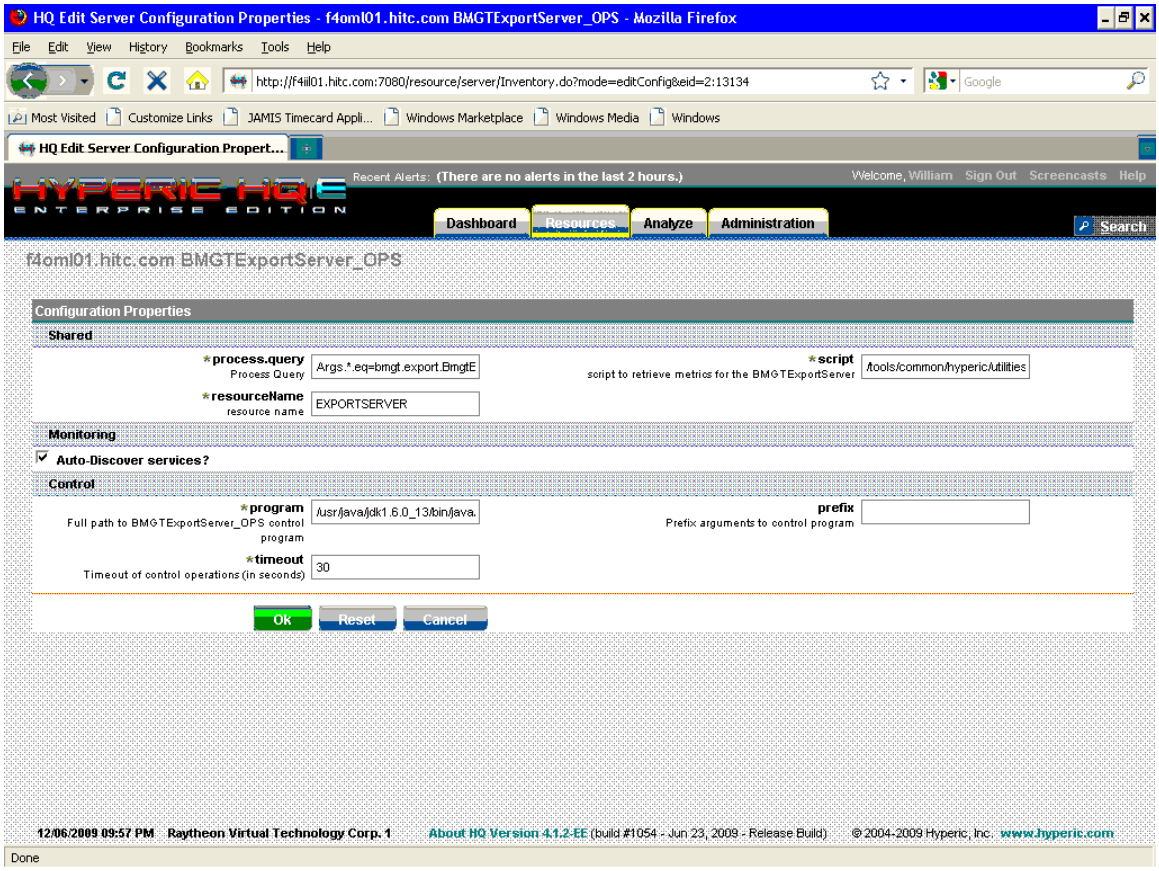

Figure 4.2.2.6.4-1. BMGT Export Server Configuration View

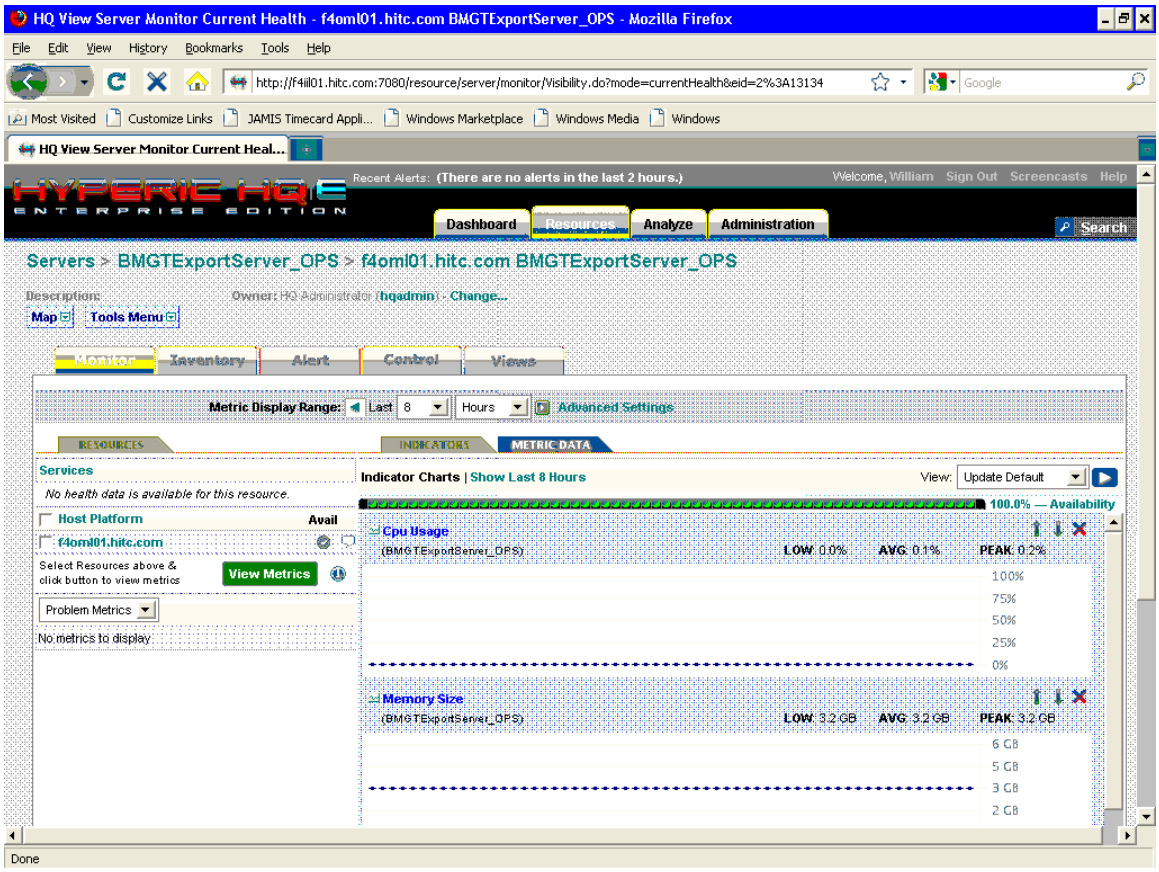

Figure 4.2.2.6.4-2. BMGT Export Server Monitoring

4.2.2.6.5 Custom Code Utilities

Custom utilities are monitored through a custom service xml plug-in. See the attached appendix for the custom utilities xml plug-in file of OPS mode.

We collect the following metrics for each custom utility:

- NumInstances
- Cpu Usage
- Memory Size
- Resident Memory Size
- LongestRunningPid
- LongestRunningTime

The custom utility plug-in is defined as a service within hyperic. Operator should add the custom utility plug-in to the host platform where the utility is installed, i.e. where it would run. This is done by clicking the “New Platform Service” menu under the “Tools Menu” of the platform detail page. See screen shot below:

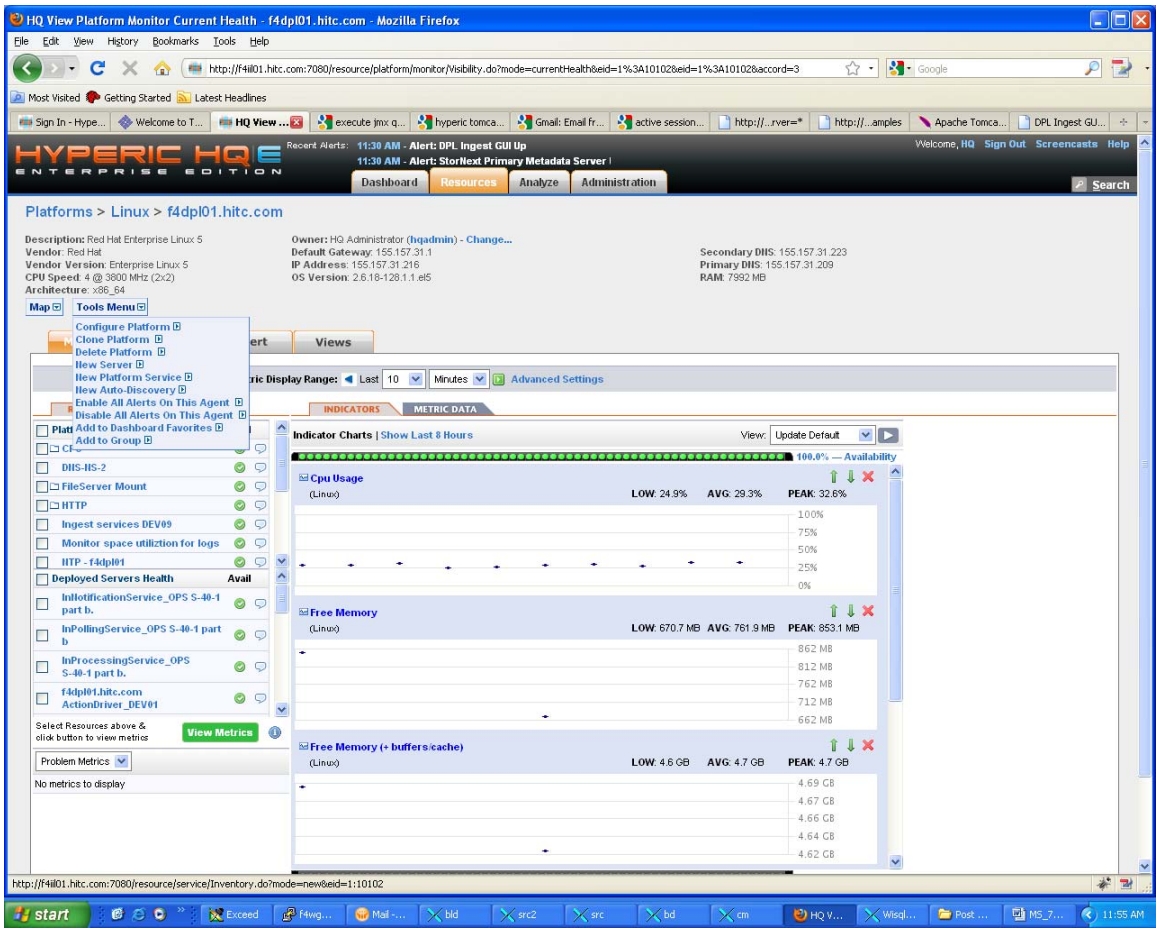

Figure 4.2.2.6.5-1. Add a New Platform Service

On the new platform service configuration page, select the custom utility service for the mode and type in the name and description, then click “OK” to create the platform service. See screen shot below:

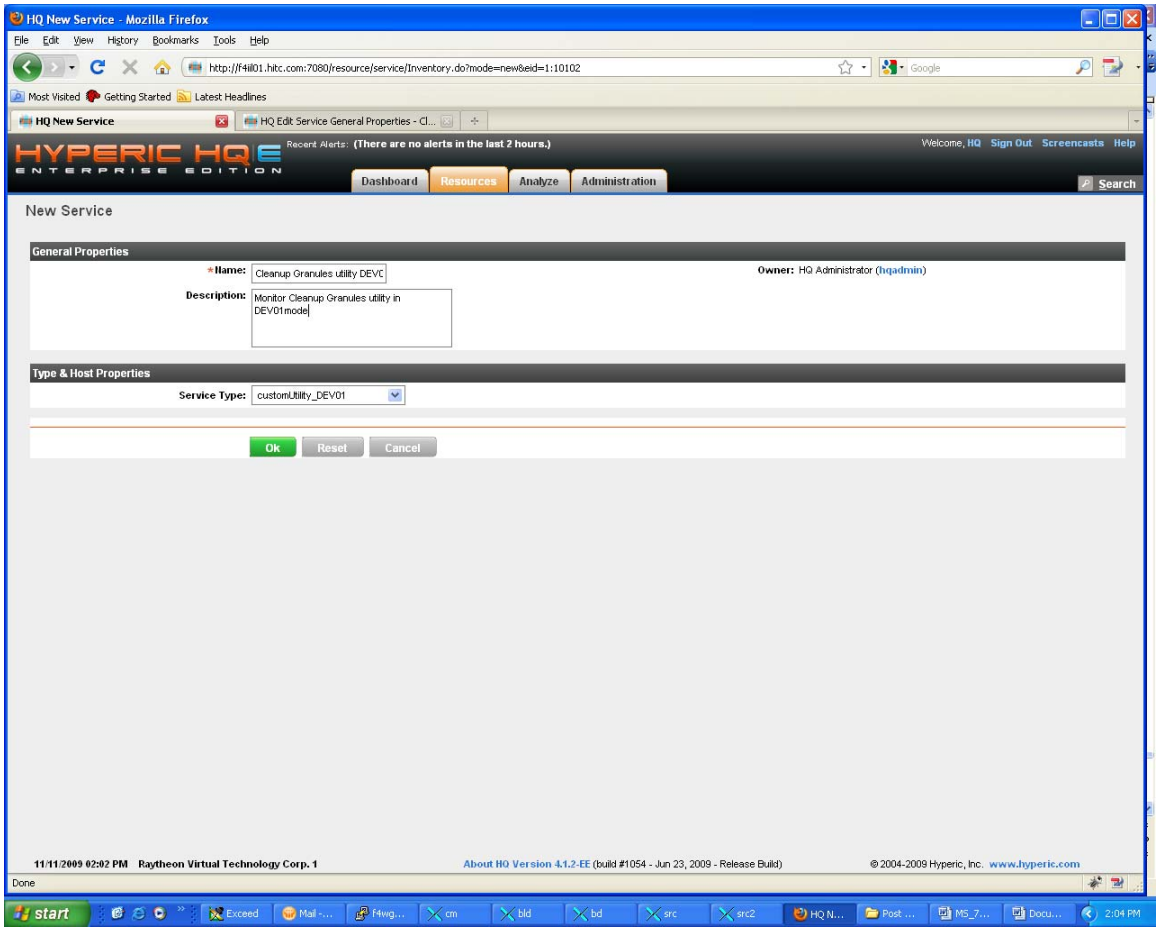

Figure 4.2.2.6.5-2. Create cleanup granules utility monitor service

Click the highlighted "Configuration Properties" link on the ensuing page (See screen shot below) to finish the configuration of the custom utility.

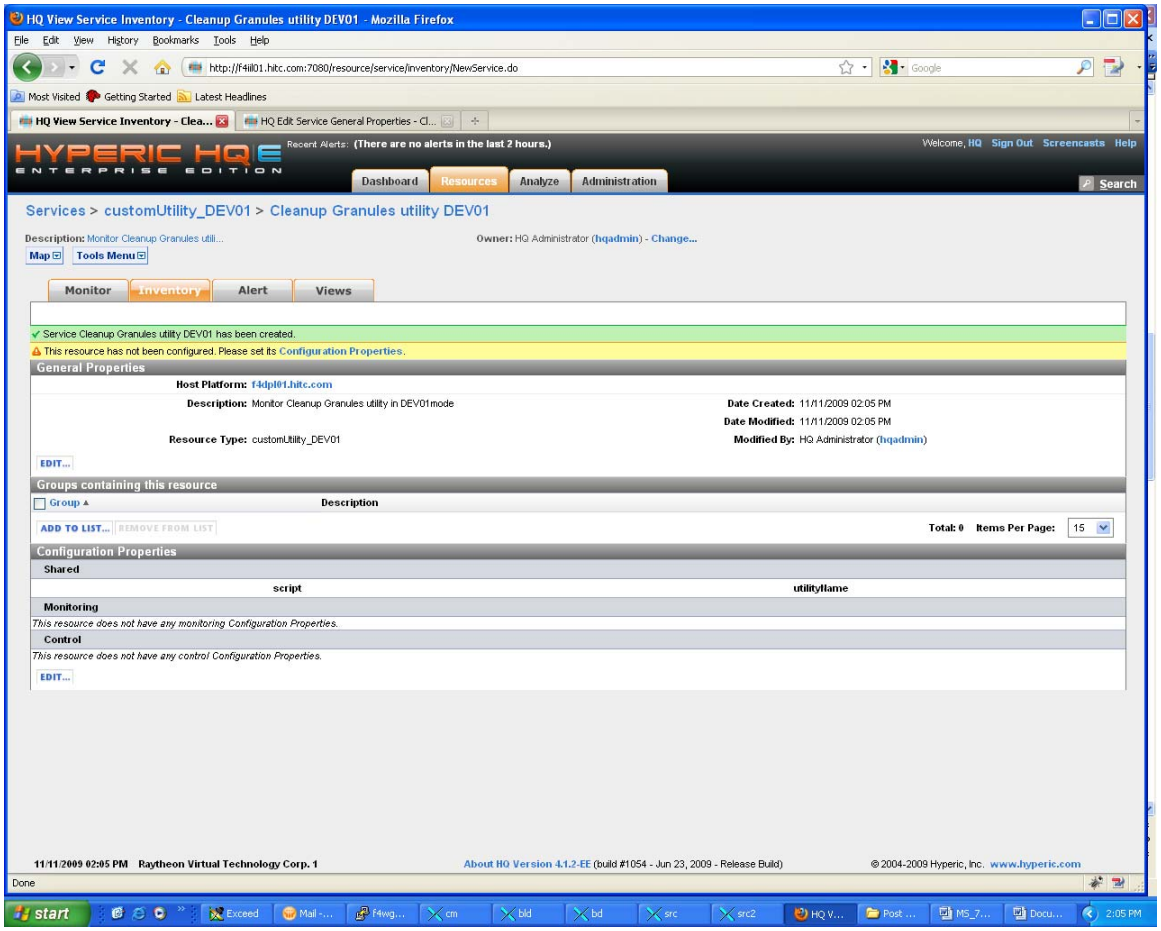

Figure 4.2.2.6.5-3. Link to Configuration Properties

On the Configuration Properties page, type in the defined utility name into the UtilityName input box, e.g. “CLEANUP_GRANULES” for the Cleanup Granules utility. Then click “OK”. See screen shot below:

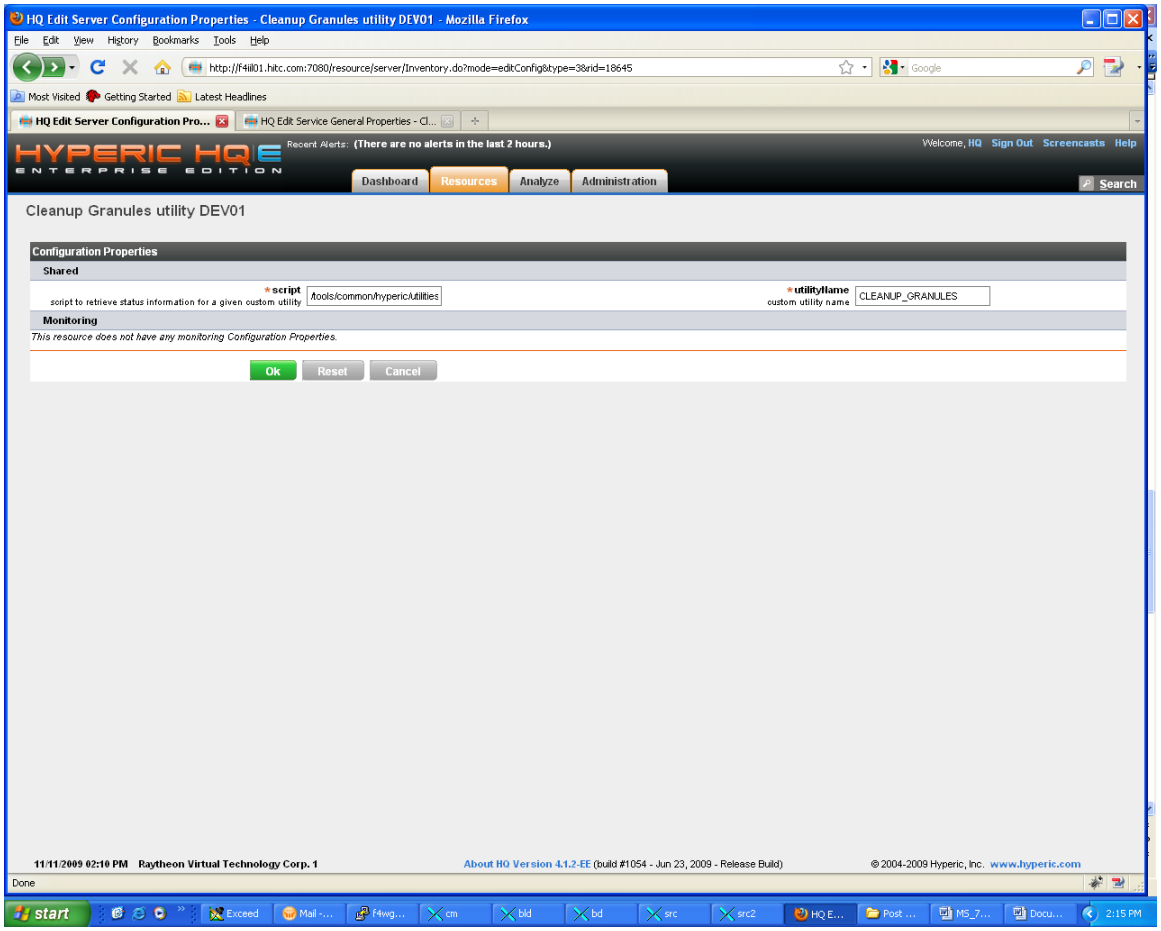

Figure 4.2.2.6.5-4. Configuration Properties Page for Cleanup Granules Utility

Below is the list of defined custom utility names:

"GRANULE_DELETE" Granule delete utility

"QA_UPDATE" QA update utility

"RESTORE_OLA" Restore OnlineArchive from tape utility

"RESTORE_TAPE" Restore tape from OnlineArchive utility

"MOVE_COLLECTION" Move collection utility

"DPCV" DataPool checksum verification utility

"ACVU" Archive checksum verification utility

"CLEANUP_GRANULES" Cleanup granules utility

"CLEANUP_FILES" Cleanup files utility

"INVENTORY_VALIDATION" Inventory validation utility

Wait for a few minutes, the metrics info will show up on the custom utility monitor page. See screen shot below:

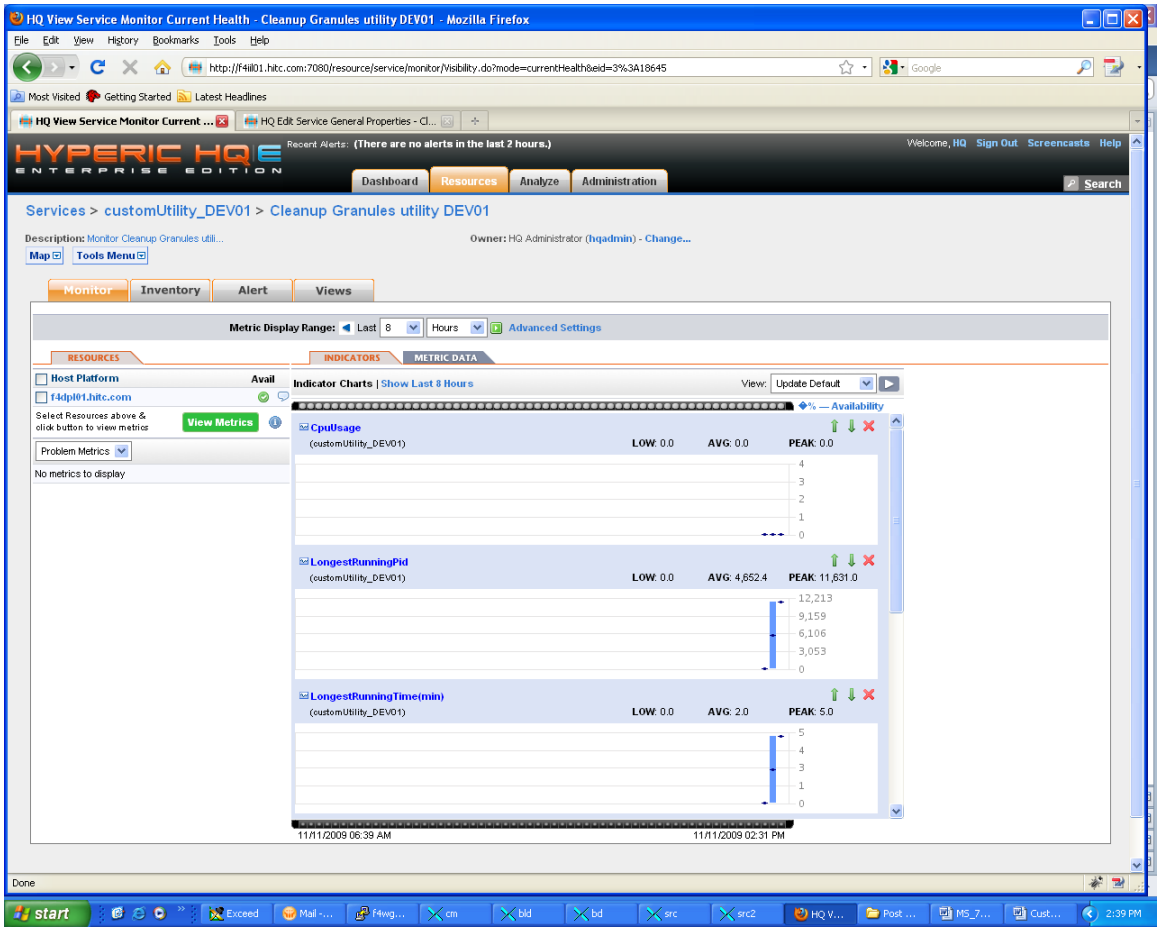

Figure 4.2.2.6.5-5. Cleanup Granules Utility Metrics

The integral part of the LongestRunningPid metric is the PID of the longest running instance of the utility. The maximum value is collected when there are multiple instances of the utility running. When there is no instance running, the metrics are set to 0. In operation environment, there should be at most one instance running for all utilities except DPCV.

4.2.2.6.6 Email GW Server

The Email GW Server monitoring is done through an xml custom plug-in. There is one plug-in file per mode. The plug-in file name is <MODE>-DPL- EcDlInEmailGWServer-plugin.xml. The Email GW Server resource will be auto-discovered by Hyperic using the PTQL query defined in the xml plug-in. The standard metrics will be collected by the SIGAR plug-in.

Up/down alerts are set up for the Email GW Server based on its availability metrics. The down alert is used to notify operator of the problem and trigger the control action that will automatically start the server; the up alert is used as a recovery alert for the down alert to mark the down alert as fixed.

Please refer to the Polling Server, section 4.2.2.4.2.3 for details on the control actions and alerts configurations. The configuration process is basically identical except the resource type, control action, script names and alert names are changed accordingly.

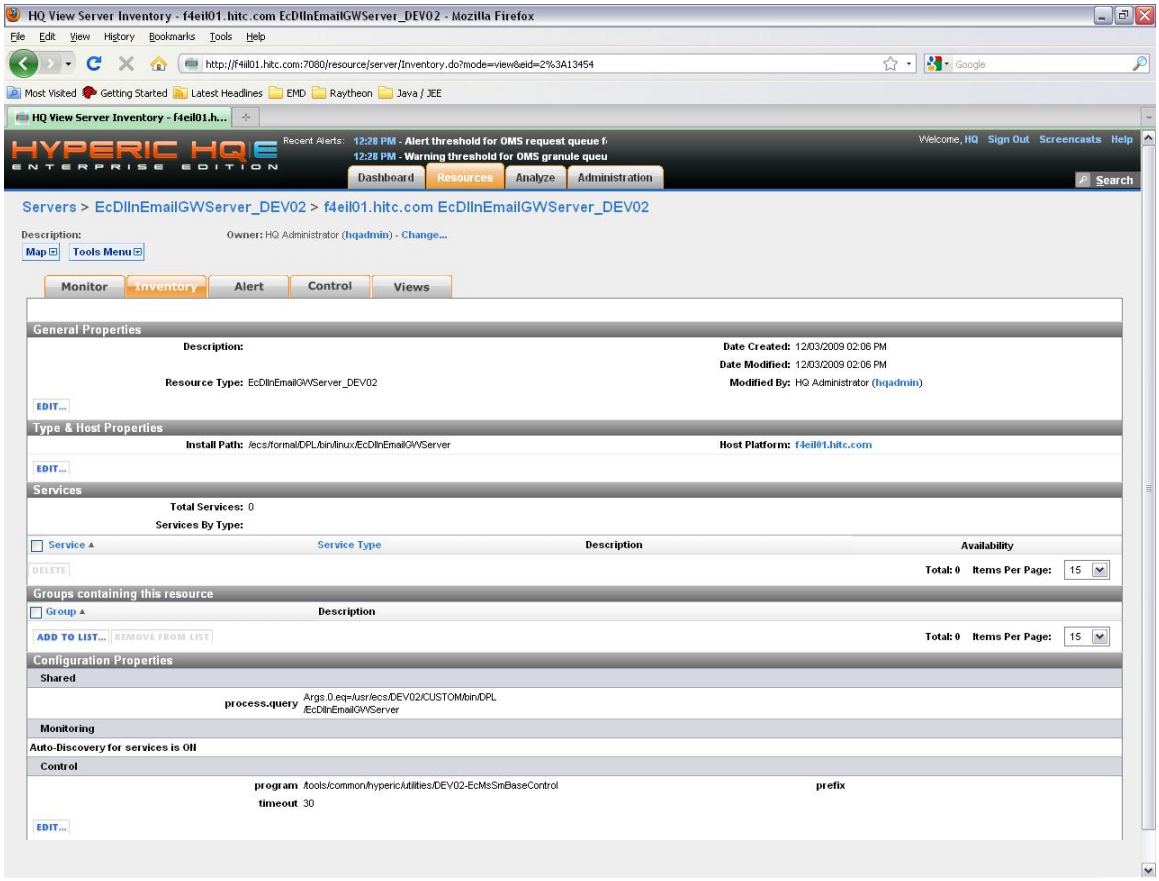

Figure 4.2.2.6.6-1. Email GW Server Configuration View

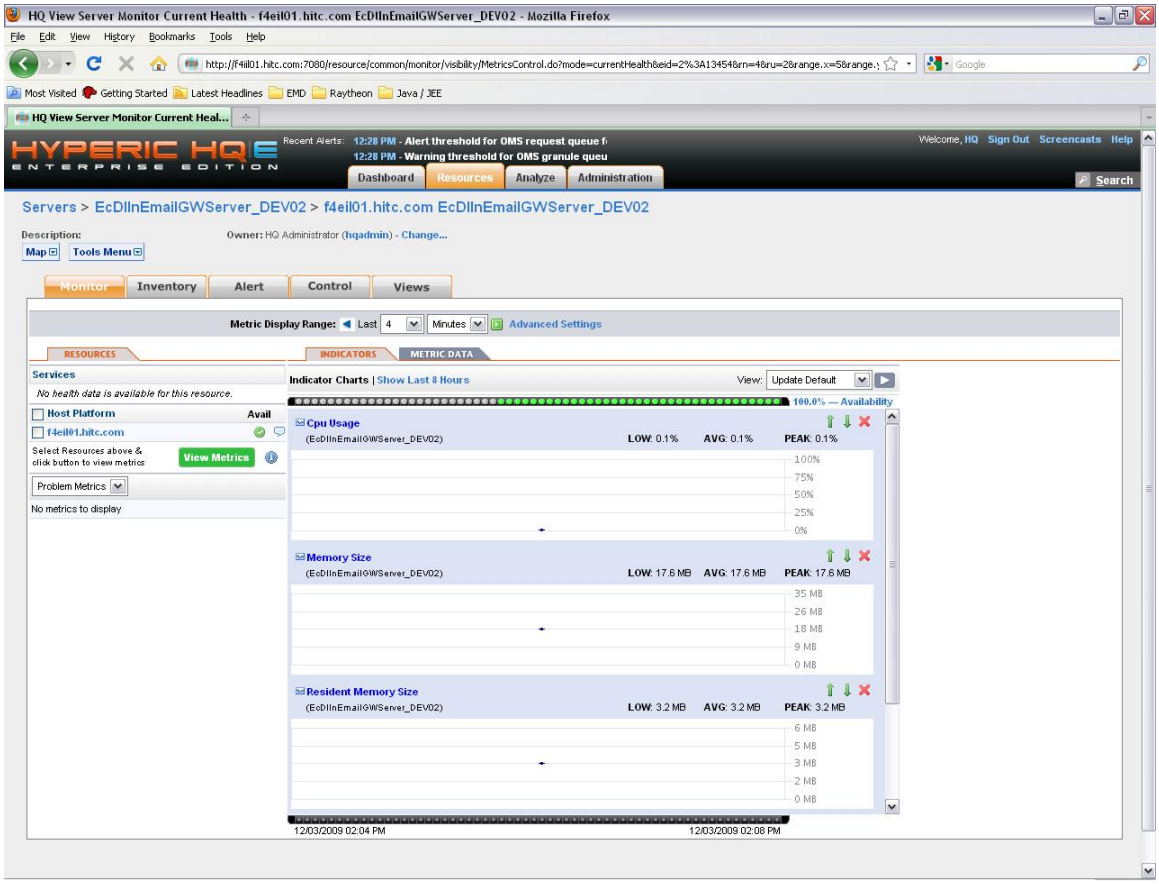

Figure 4.2.2.6.6-2. Email GW Server Monitoring

4.2.2.6.7 Action Driver Server

The Action Driver Server monitoring is done through an xml custom plug-in. There is one plug-in file per mode. The plug-in file name is <MODE>-DPL-ActionDriver-plugin.xml.

The metrics that will be collected for the Action Driver Server includes the standard Hyperic metrics *(see section 4.2.2.2.13, Standard Plugin Metrics)* and custom metrics (see below).

Table 4.2.2.6.7-1. Action Driver Server Metrics (1 of 2)

Custom Metrics	
Metric Name	**Description**
DPL Insert Queue Size	The total number of granules waiting to be inserted.
Number of Granules in 'Pending' State	The total number of granules in the PENDING state.

Table 4.2.2.6.7-1. Action Driver Server Metrics (2 of 2)

Custom Metrics	
Metric Name	**Description**
Number of Granules in 'Validated' State	The total number of granules in the VALIDATED state.
Number of Granules in 'Copied' State	The total number of granules in the COPIED state.
Number of Granules in 'Checksummed' State	The total number of granules in the CHECKSUMMED state.
Number of Granules in 'Extracted' State	The total number of granules in the EXTRACTED state.
Number of Failed Inserts	The total number of failed DPL Inserts.
Number of Recent Failed Inserts	The total number of recently failed DPL inserts. The total is a count of failed inserts from the current time to a calculated end time. This end time is dependent on a configurable interval.
Number of Completed Inserts	The total number of completed DPL inserts.
Number of Recent Completed Inserts	The total number of recently completed DPL inserts. The total is a count of completed inserts from the current time to a calculated end time. This end time is dependent on a configurable interval.
Number of Available DPL File System	The numbers of available DPL file systems.
Number of Unavailable DPL File System	The numbers of unavailable DPL file systems.
Number of Active Checksum Service	The numbers of active checksum service host.
Number of Disabled Checksum Service	The numbers of disabled checksum service host.
Number of Suspended Checksum Service	The numbers of suspended checksum service host.
Number of Active Copy Service	The numbers of active copy service host.
Number of Disabled Copy Service	The numbers of disabled copy service host.
Number of Suspended Copy Service	The numbers of suspended copy service host.

The Action Driver Server resource will be auto-discovered by Hyperic using the PTQL query defined in the xml plug-in. The standard metrics will be collected by the SIGAR plug-in and the custom metrics will be collected through the SQL plug-in by querying the DPL database.

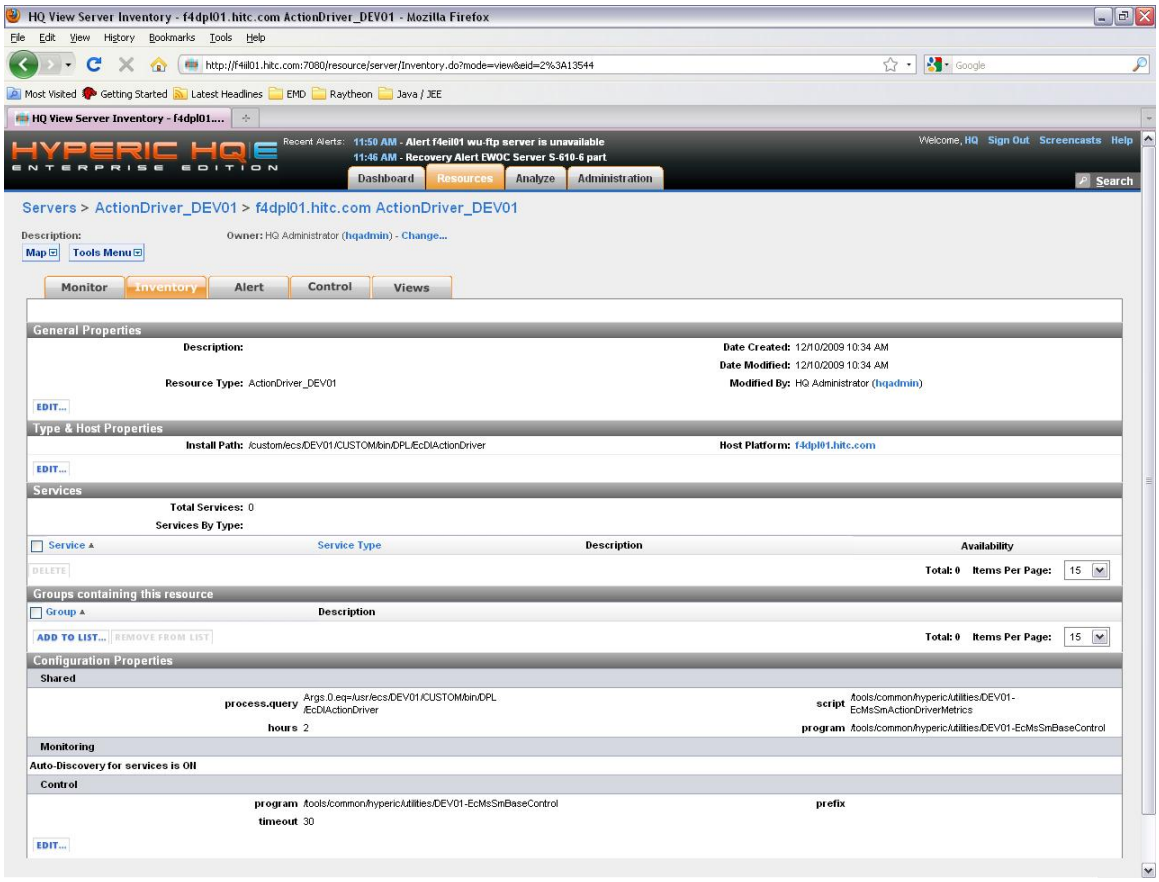

Figure 4.2.2.6.7-1. Action Driver Server Configuration View

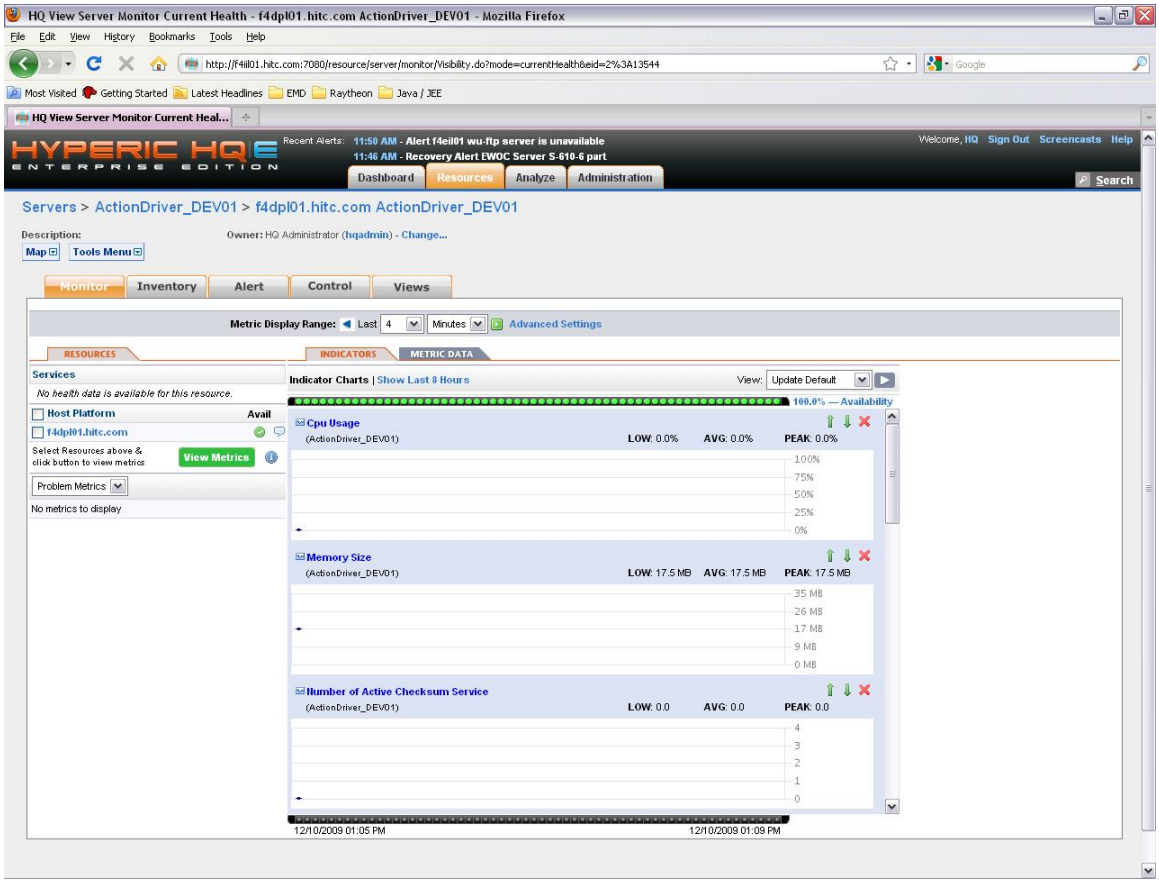

Figure 4.2.2.6.7-2. Action Driver Server Monitoring

The data collected for the 'Number of Recent Failed Inserts' and the 'Number of Recent Completed Inserts' metrics are configurable. To configure this value go to the Action Driver server detail page and click on the 'Inventory' (see Figure 233). In the configuration properties section, click the edit button and modify the hours field. This field is used to calculate the end time (current time minus the configured hours). Using this end time, the agent will get a count of failed and completed inserts from the current time to the end date.

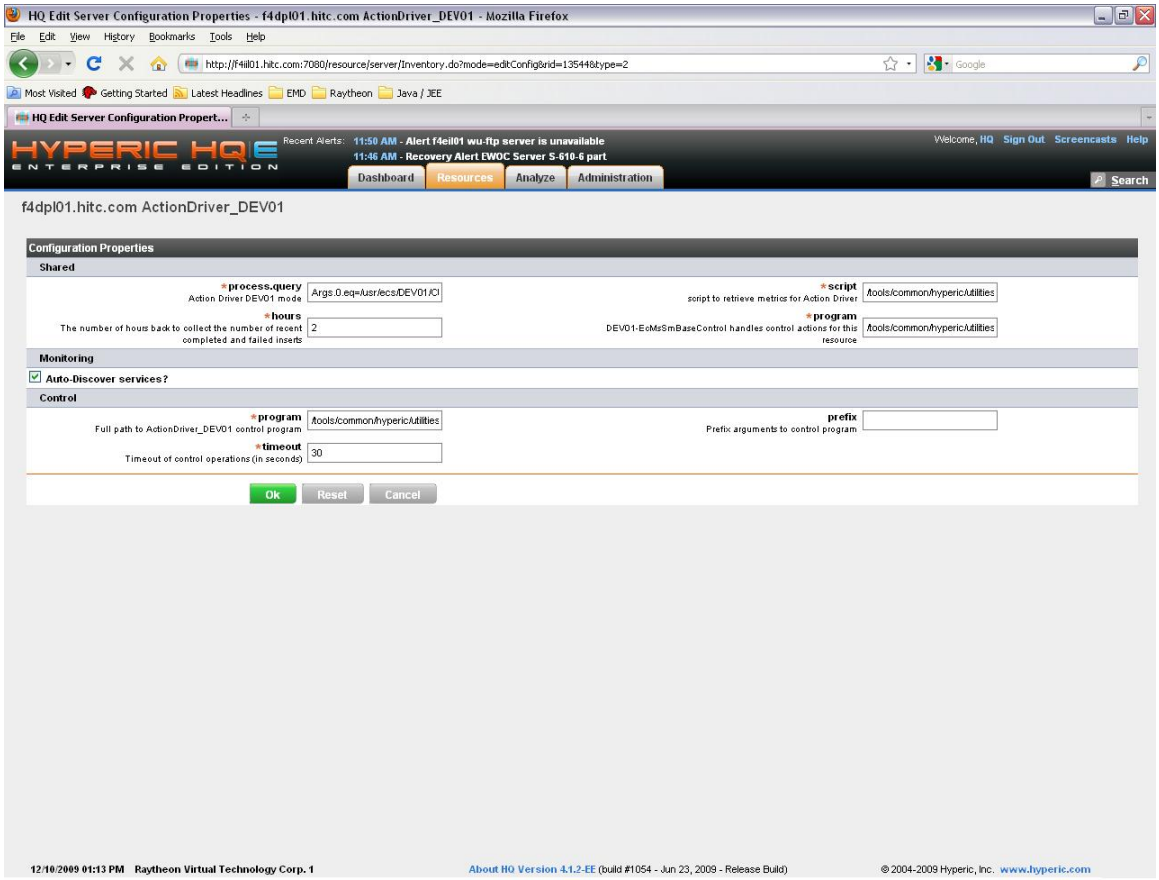

Figure 4.2.2.6.7-3. Action Driver Configure Metric Interval

Up/down alerts are set up for the Action Driver Server based on its availability metrics. The down alert is used to notify operator of the problem and trigger the control action that will automatically start the server; the up alert is used as a recovery alert for the down alert to mark the down alert as fixed.

Please refer to the Polling Server, section 4.2.2.2.13 for details on the control actions and alerts configurations. The configuration process is basically identical except the resource type, control action, script names and alert names are changed accordingly.

4.2.2.6.8 Checksum Verification Server

The Checksum Verification Server monitoring is done through an xml custom plug-in. There is one plug-in file per mode. The plug-in file name is <MODE>-DPL-ChecksumServer-plugin.xml. The Checksum Verification Server resource will be auto-discovered by Hyperic using the PTQL query defined in the xml plug-in. The standard metrics will be collected by the SIGAR plug-in.

Up/down alerts are set up for the Checksum Verification Server based on its availability metrics. The down alert is used to notify operator of the problem and trigger the control action that will

automatically start the server; the up alert is used as a recovery alert for the down alert to mark the down alert as fixed.

Please refer to the Polling Server, section 4.2.2.4.2.3 for details on the control actions and alerts configurations. The configuration process is basically identical except the resource type, control action, script names and alert names are changed accordingly.

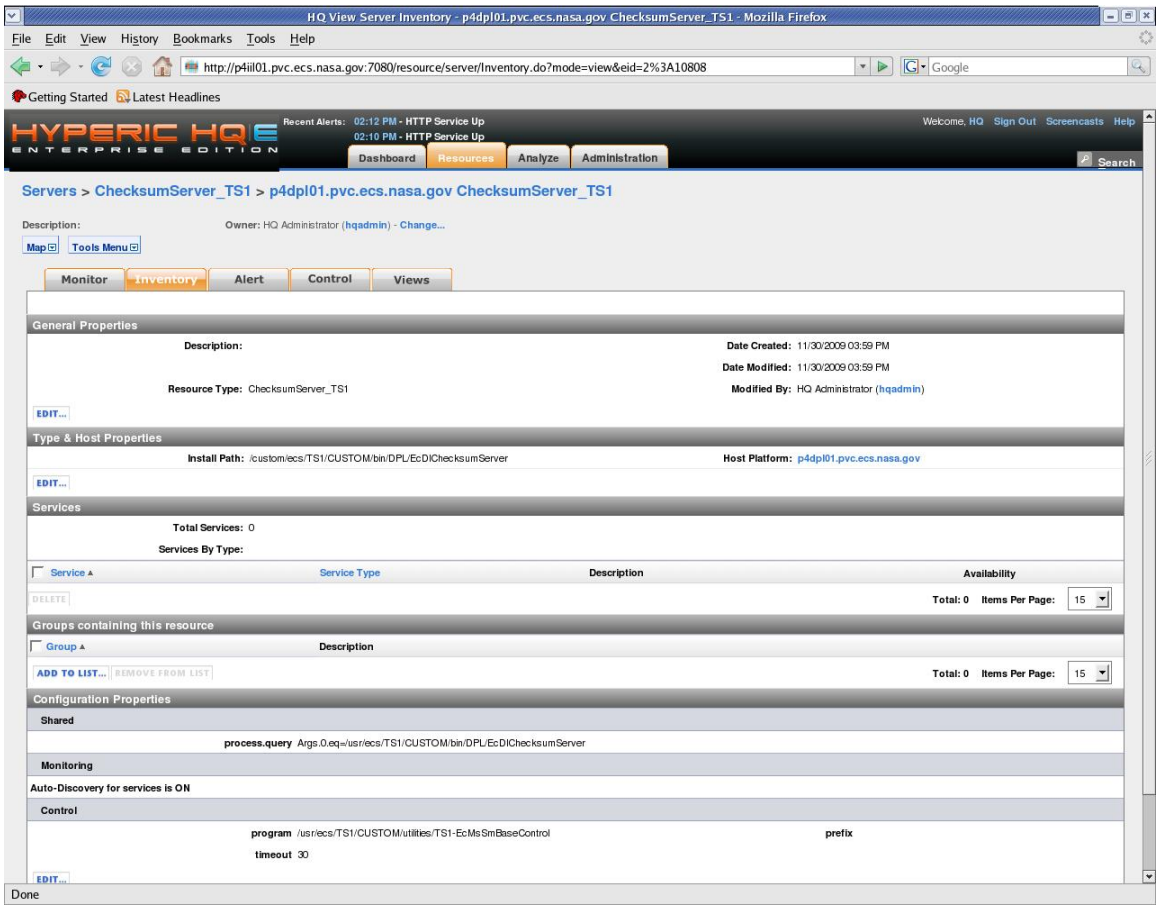

Figure 4.2.2.6.8-1. Checksum Verification Server Configuration View

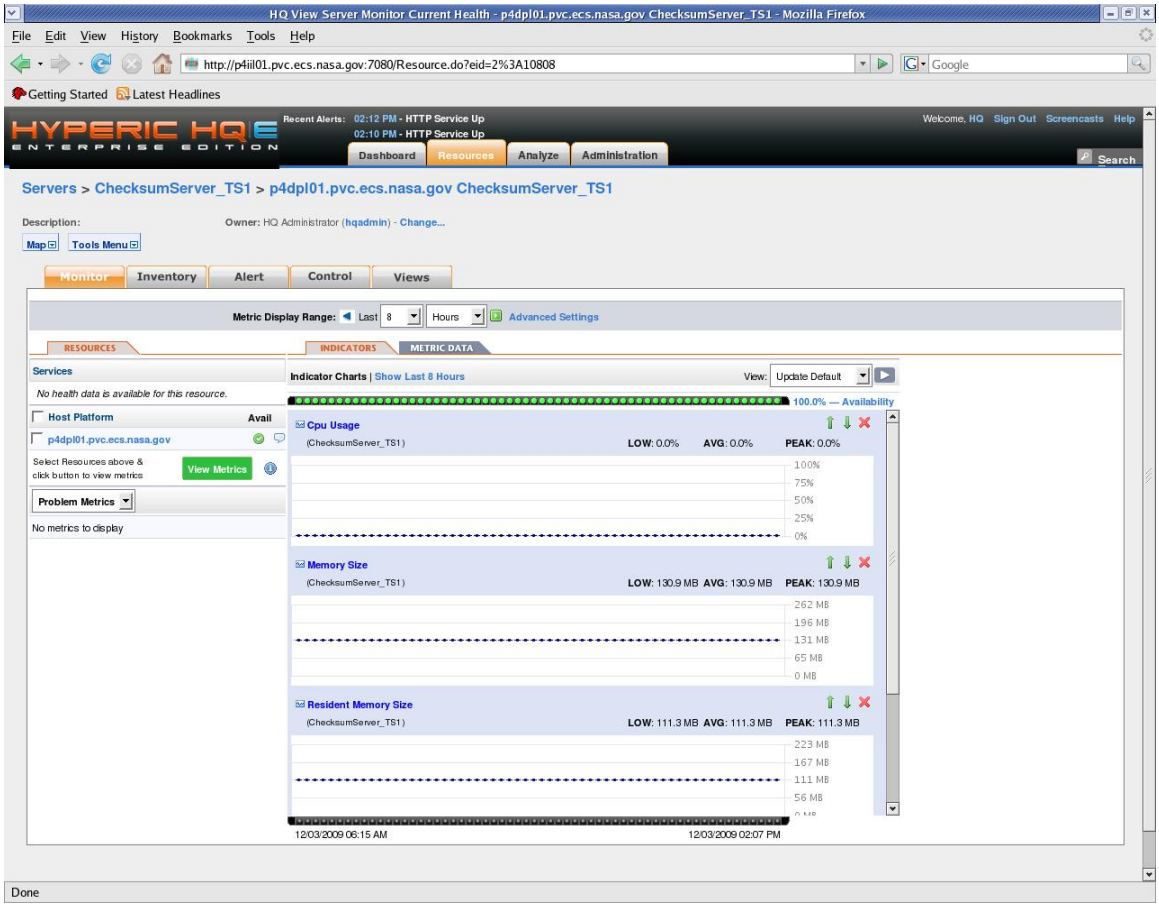

Figure 4.2.2.6.8-2. Checksum Verification Server Monitoring

4.2.2.6.9 Subscription Servers

Monitoring Spatial Subscription Server Drivers work by configuring the designated server on the xxoml01 platform. The 4 different servers are the Action Driver, Event Subscription Driver, Delete Request Driver, and the Recovery Driver. The Action, Subscribed Event, and Delete Request Drivers have their respective metrics monitoring the amount of work each has left to do. All 4 servers have the following metrics: Availability, Cpu Usage, Memory Size, Number of Processes, and Resident Memory Size. To configure an SSS Driver, perform the following steps.

1. Choose the xxoml01 platform to create the SSS Driver.
2. Drill down to the platform using the Resources->Browse on the Hyperic GUI. The screen should look like the following.

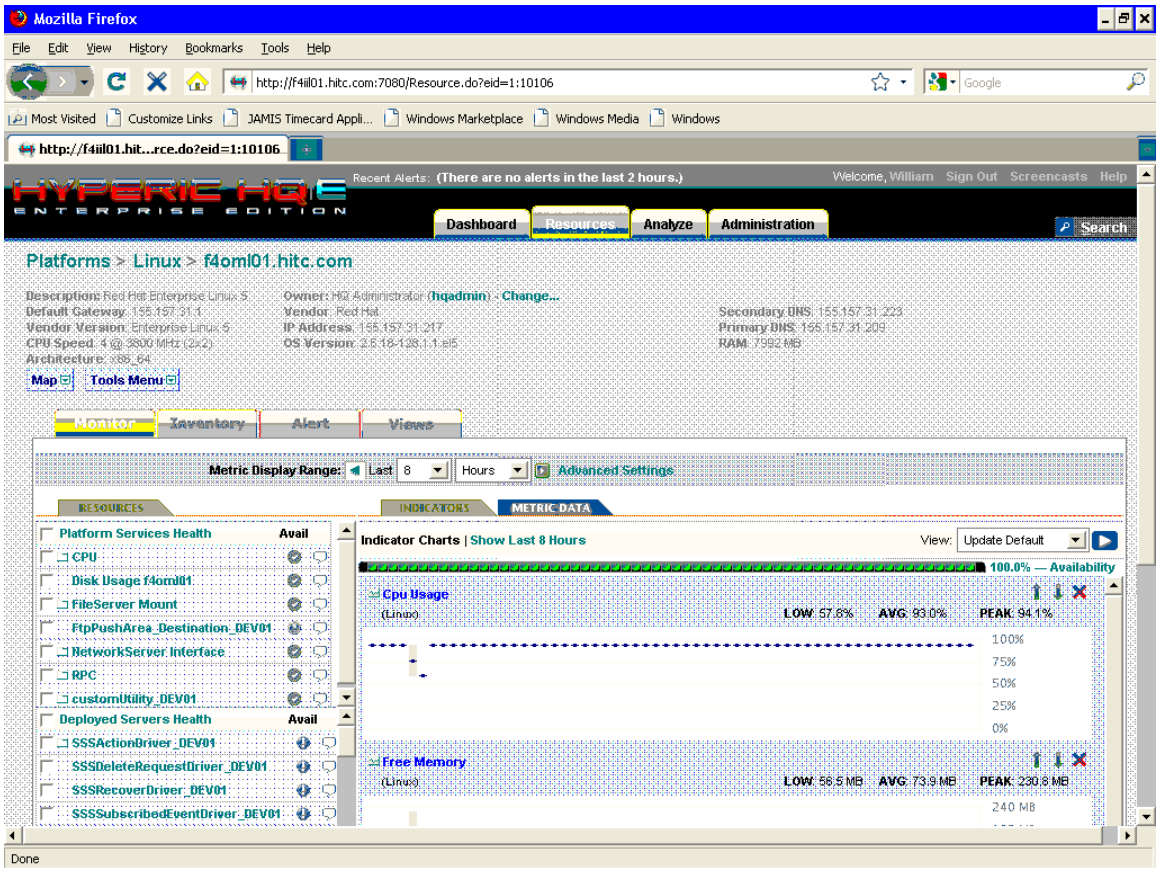

Figure 4.2.2.6.9-1. Host Resource Page

3. From this screen choose from the Tools Menu (upper left) the New Server link.

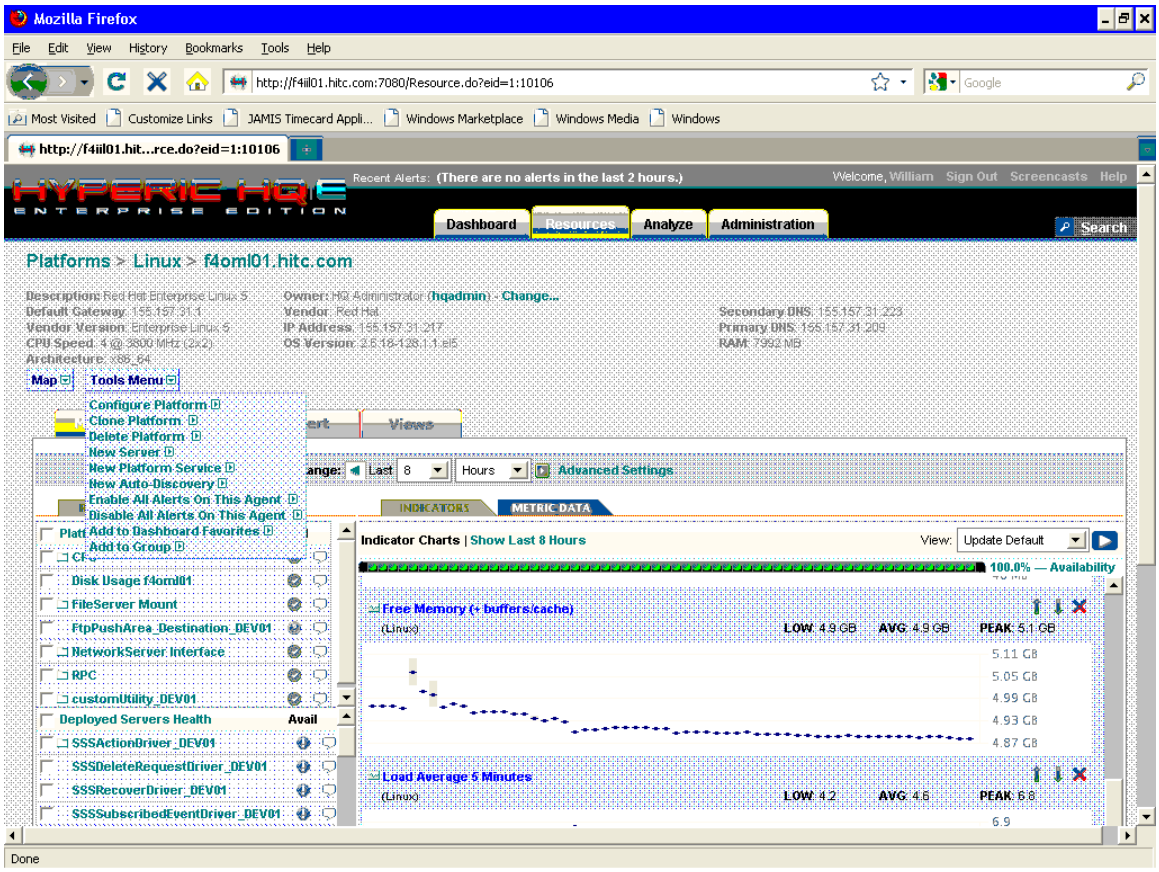

Figure 4.2.2.6.9-2. Add New Platform Server

4. The following screen appears. Fill in the following

   name: Choose a name such as SSS Action Driver DEV01

   description: Choose a description such as "SSS Action Driver for DEV01 mode"

   server type: From the drop down menu choose the SSS Driver for the mode you are working in ie: SSSActionDriver_DEV01

   install path: For the install path just use the base directory "/" without the quotes.

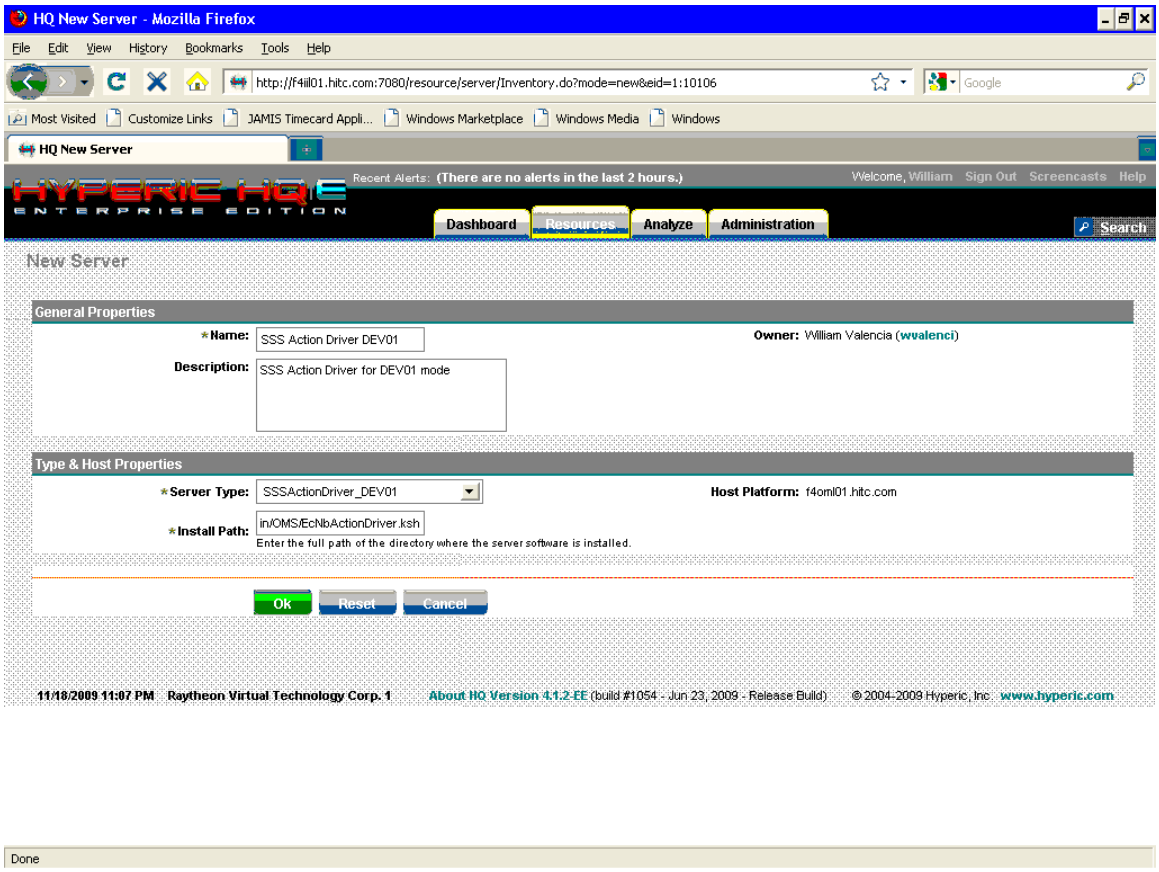

Figure 4.2.2.6.9-3. Configuration Screen

5. Click “OK” button.
6. The next screen allows for the configuration of the SSS Driver. Hit the Edit button in the Configuration Properties section or click on Configuration Properties on the yellow pane

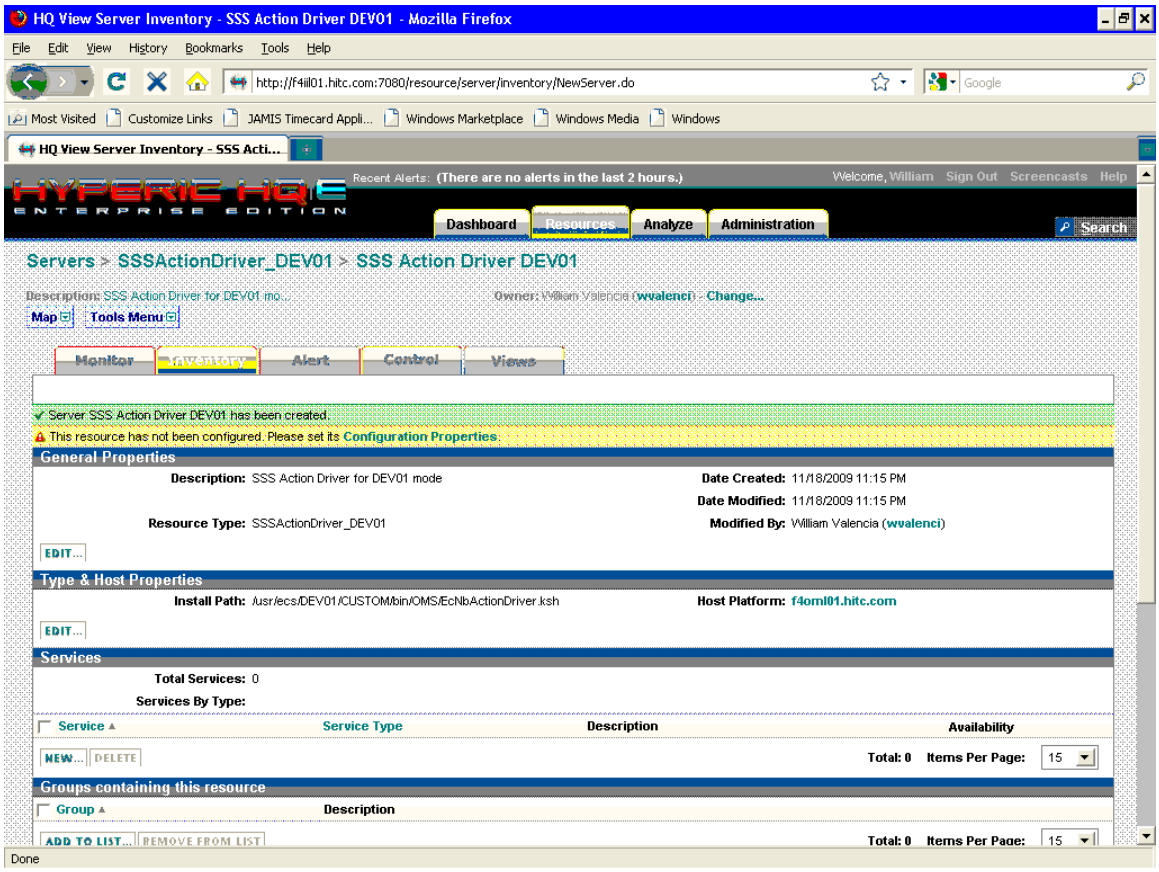

Figure 4.2.2.6.9-4. SSS Action Driver Inventory Page

7. You should not need to fill in anything here.

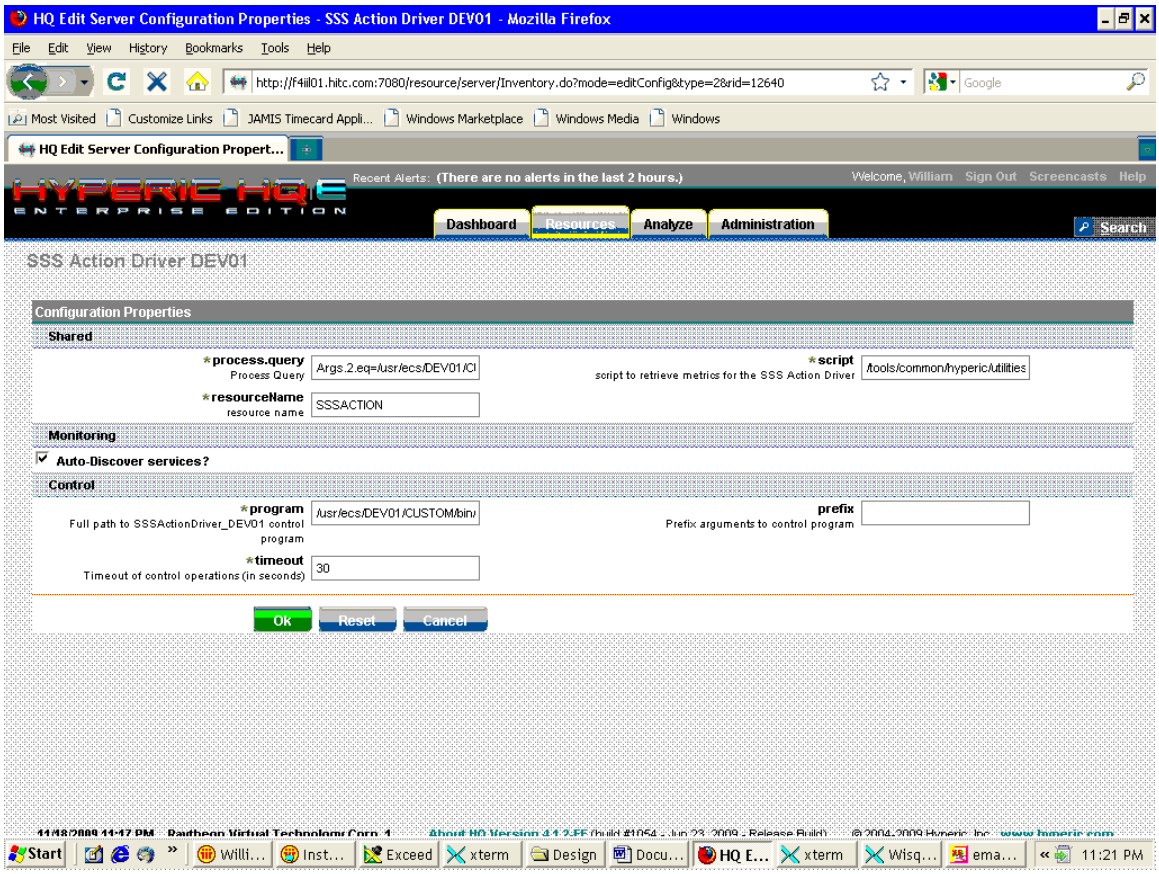

Figure 4.2.2.6.9-5. ssh Configuration Page

8. Click “OK” button.
9. After a few minutes, the SSS Driver Server monitor page should appear similar to the following page.

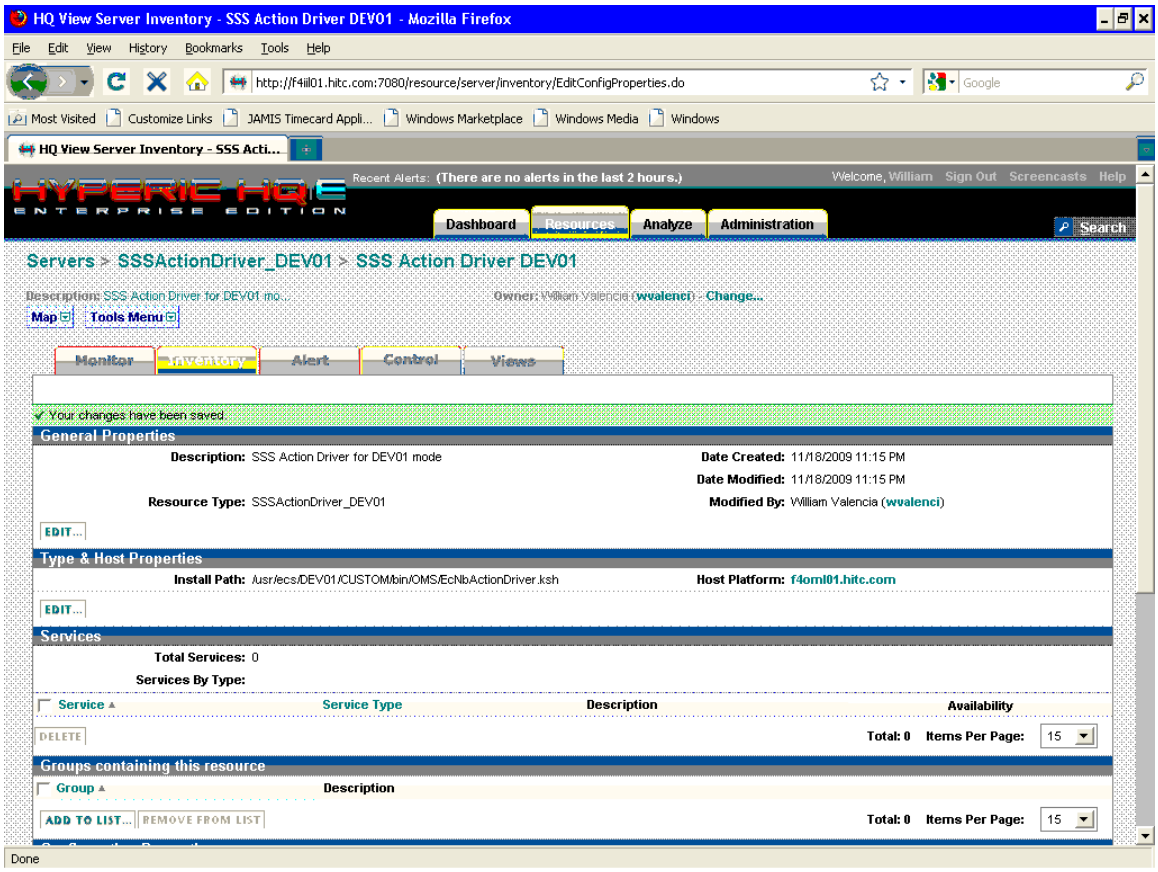

Figure 4.2.2.6.9-6. SSS Driver Configured Monitor Page

10. Repeat steps 1-9 for the rest of the SSS Drivers. You would need to run through this 4 times, Once for each of the following SSS Drivers SSSAction Driver, SSS Subscribed Event Driver, SSS Delete Request Driver, SSS Recover Driver.

11. If you want to configure the default number of drivers that start for the SSS start scripts, you can modify each of the driver's respective .CFG files located under /usr/ecs/<MODE>/CUSTOM/cfg directory on the x4oml01 box. The configuration parameter is called DEFAULT_NUM_DRIVERS.

4.2.2.7 Server Control

4.2.2.7.1 Overview

Server Control is designed for the user to be able to start or stop certain modes, servers, or resources. It can be done in two ways, namely through the GUI or through the command line. The sections below briefly describe how that can be achieved.

4.2.2.7.2 Server Control Command Line Tool

The Server Control Command Line utility will provide the user the ability to start and stop custom code servers via web service calls to custom services running on the Hyperic server. The result of these calls will be to disable or enable the alert definition which is mapped to the availability metric associated with the server being stopped or started. The associated servers start or stop script will then be executed by Hyperic via a Hyperic agent resulting in the server being stopped or started.

The Server Control Command Line utility provides a menu driven ability to start or stop servers by mode, by mode and host, or by individual server. The utility also allows for these options to be passed in via the command line.

4.2.2.7.2.1 Server Control Command Line Tool Usage

Starting a mode via the menu:

./HypericControlCommandLine

Action:

1)Start Mode

2)Stop Mode

3)Start Servers by Mode/Host

4)Stop Servers by Mode/Host

5)Start Server

6)Stop Server

Enter:1

Action:<Start Mode> Mode:

1)OPS

2)TS1

Enter the mode:1

Processed servers running in mode OPS

Starting a mode via command line options:

./HypericControlCommandLine –ACTION [START_MODE,STOP_MODE] –MODE <mode>

Starting of servers on a given host running in a given mode via the menu:

./HypericControlCommandLine

Action:

1)Start Mode

2)Stop Mode

3)Start Servers by Mode/Host

4)Stop Servers by Mode/Host

5)Start Server

6)Stop Server

Enter:3

Action:<Start Servers by Mode/Host> Mode:

1)OPS

2)TS1

Enter the mode:1

Action:<Start Servers by Mode/Host> Mode:<OPS> Host:

1)f4oml01.hitc.com

2)f4eil01.hitc.com

Enter the Host:1

Processed servers running in mode OPS on host f4oml01.hitc.com

Starting of servers on a given host running in a given mode via command line options:

./HypericControlCommandLine –ACTION [START_HOST,STOP_HOST] –MODE <mode> -HOST <host>

Starting of an individual server via the menu:

Action:

1)Start Mode

2)Stop Mode

3)Start Servers by Mode/Host

4)Stop Servers by Mode/Host

5)Start Server

6)Stop Server

Enter:5

Action:<Start Server> Mode:

1)OPS

2)TS1

Enter the mode:1

Action:<Start Server> Mode:<OPS> Host:

1)f4oml01.hitc.com

2)f4eil01.hitc.com

Enter the Host:2

Action:<Start Server> Mode:<OPS> Host:<f4eil01.hitc.com> Server:

1)EcDlPollingServer

2)EcDlQuickServer

Enter the Server:1

Processed server EcDlPollingServer

Starting of a server via command line options:

./HypericControlCommandLine –ACTION [START_SERVER,STOP_SERVER] –MODE <mode> -HOST <host> -SERVER <server>

4.2.2.7.3 Server Control GUI

In addition to the Server Control Command Line utility a user is able to disable and enable custom code servers through the Hyperic GUI on a new page labeled “Server Center”. The Server Center link is located under the Analyze tab.

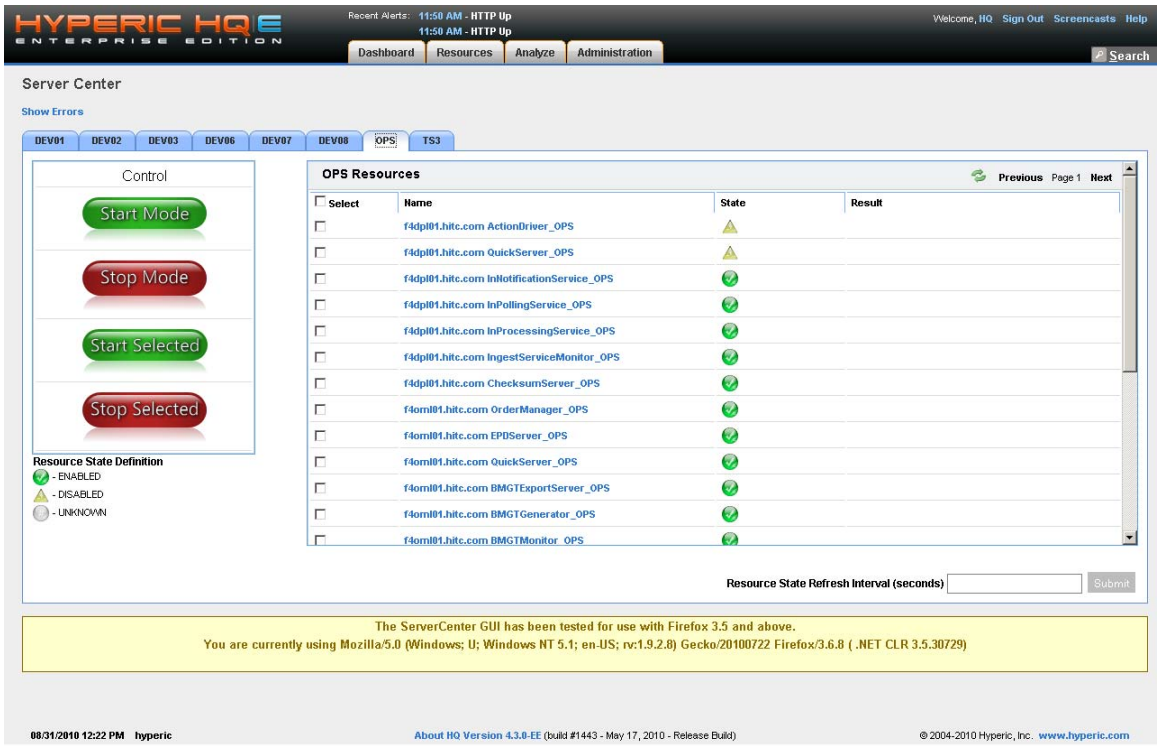

Figure 4.2.2.7.3-1. Server Center Main Screen

The server center page is divided based upon mode tabs. When a particular mode is selected the page will only show custom code servers pertaining to the selected mode. An entire mode can be brought down by simply clicking on the “Stop Mode” button and the mode that you are currently viewing will be stopped intelligently (each server will be stopped in the order specified by a start and stop precedence number). Similarly, a mode can be started by selecting the “Start Mode” button. An individual resource that is part of that mode can be started or stopped – once the selected resources are selected on the right side, one can either start or stop them by using the “Start Selected” or the “Stop Selected” buttons on the left side.

Resource State

The state of a resource is dependent upon the its availability alerts. ***The state of a resource does not denote the resource’s availability***. There are three recognized states for a resource – Enabled, Disabled, and Warning.

Table 4.2.2.7.3-1. Server Center State Icon Description

State Icon	State Definition	Description
	Enabled	**At least one of the alerts pertaining to the resource that has a condition for availability is active**
	Disabled	**All of the alerts pertaining to the resource that has condition for availability is inactive**
	Unknown	**There are no alerts for the specified that contain availability as a condition thus the state of the resource cannot be determined**

Start/Stop Resource Status Feedback

When a resource is selected to be started, a control action for the resource is scheduled and deployed to the agent on the host on which the server is running on.

Before the control action is passed to agent, a pop-up dialog is shown to confirm the operation; Server Center lets the user choose whether to proceed the command or not.

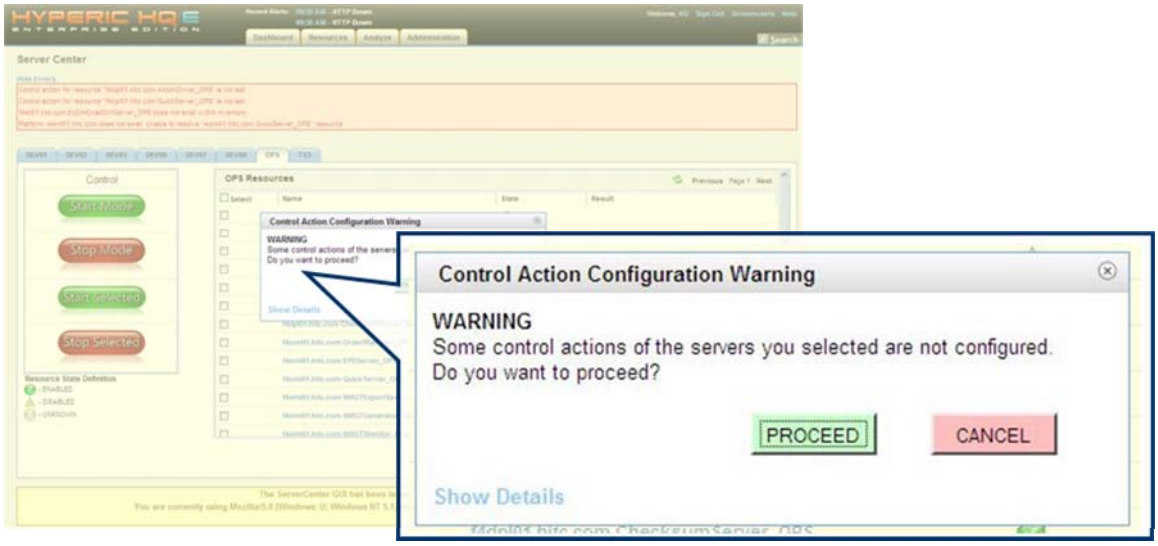

Figure 4.2.2.7.3-1. Pop-up Dialog

By clicking on “Show Details”, more information will be provided:

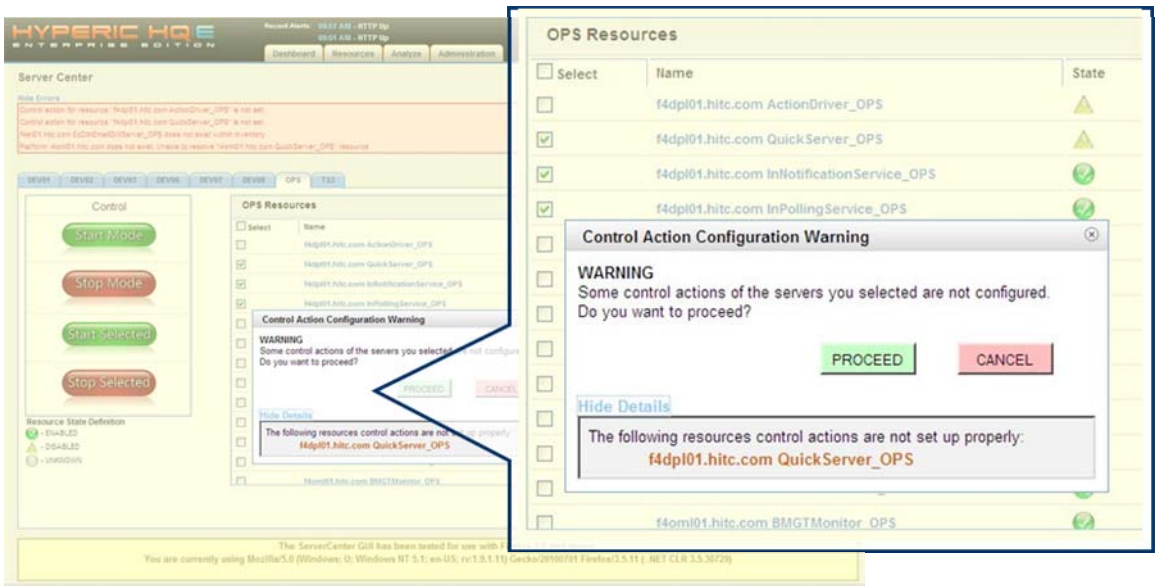

Figure 4.2.2.7.3-2. Show Detail Pane

At the same time, Server Center will again make sure that all the resources Start/Stop control actions have been set up in the configuration file. Also, Server Center checks all the agents availabilities before any control actions are executed by the user.

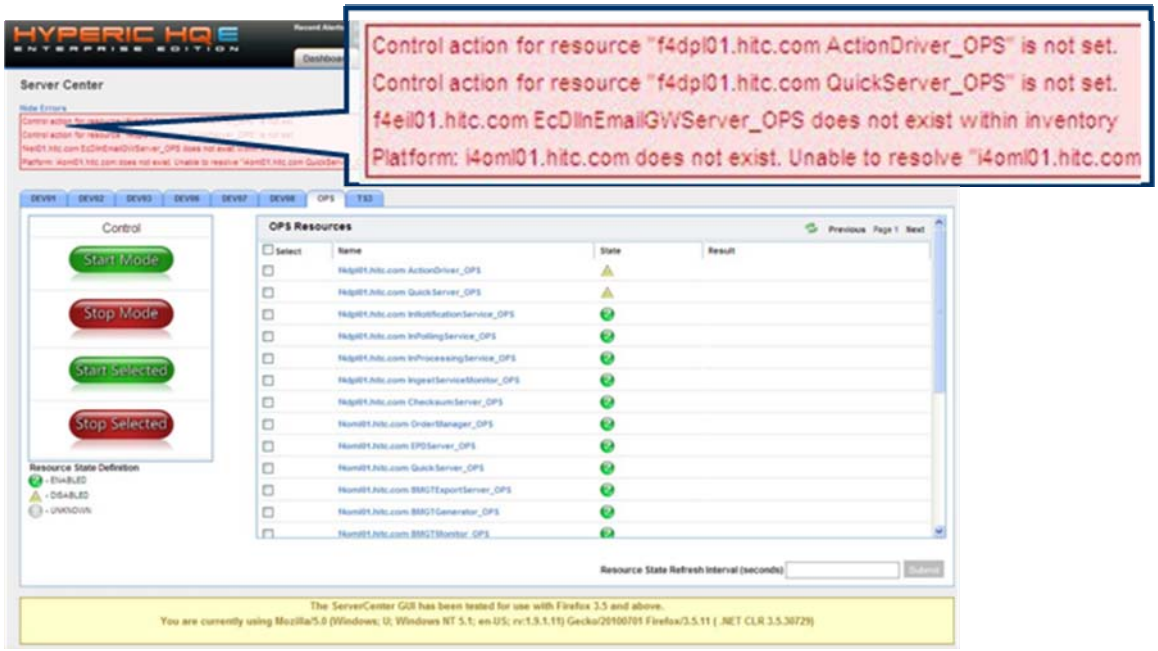

Figure 4.2.2.7.3-3. Red Error Pane

The agent then executes the control action which will invoke a start or stop script. The agent will send the Hyperic server a status report on the issued control action. This status is relayed to the GUI in the “result” column of the resources table. A green loading bar will appear for resources that have been issued a start command and the process has not yet completed. Similarly, a red loading bar will appear for resources that have been issued a stop command the process has not yet completed. Upon completion of any action the status of the control action will be printed in the result column pertaining to the resource.

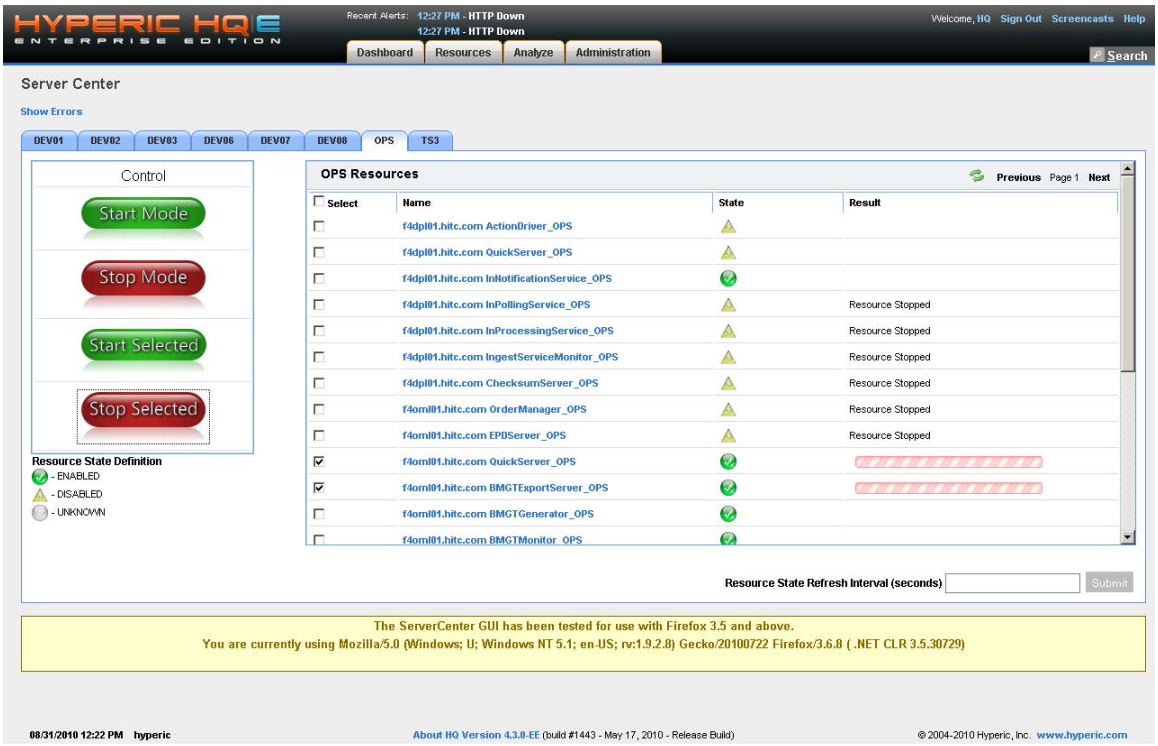

Figure 4.2.2.7.3-4. Server Control Status Feedback Example

4.2.2.7.4 Server Control Configuration

The Server Control Command line utility and GUI will have a configuration file that has the following parameters, HOST, PORT, USER, ENCRYPTED_PWD along with an xml file that contains information about a DAACs holding. The delivered xml file will be preconfigured for each DAAC. DAACs may add to this xml if they want to control their DUE’s.

Example Configuration File:

<?xml version="1.0" encoding="UTF-8"?>

<HypericControl xsi:noNamespaceSchemaLocation="HypericControl2.xsd" xmlns:xsi="h

ttp://www.w3.org/2001/XMLSchema-instance">

```
<DAAC Name="PVC">
    <MODE Enabled="true" Name="OPS">
        <Host Name="f4oml01.hitc.com">
            <Server Enabled="true" Name="EcOmOrderManager">
                <StartScript>EcOmOrderManagerStart</StartScript>
                <StartNumber>0</StartNumber>
                <StopScript>EcOmOrderManagerStop</StopScript>
                <StopNumber>0</StopNumber>
            </Server>
            <Server Enabled="true" Name="EcDlQuickServer">
                <StartScript>EcDlQuickServerStart</StartScript>
                <StartNumber>0</StartNumber>
                <StopScript>EcDlQuickServerStop</StopScript>
                <StopNumber>0</StopNumber>
            </Server>
        </Host>
        <Host Name="f4eil01.hitc.com">
            <Server Enabled="true" Name="EcDlInPollingService">
                <StartScript>EcDlInPollingServiceStart</StartScript>
                <StartNumber>0</StartNumber>
                <StopScript>EcDlInPollingServiceStop</StopScript>
                <StopNumber>0</StopNumber>
            </Server>
            <Server Enabled="true" Name="EcDlQuickServer">
                <StartScript> EcDlQuickServerStart</StartScript>
                <StartNumber>0</StartNumber>
                <StopScript> EcDlQuickServerStop</StopScript>
                <StopNumber>0</StopNumber>
```

```
            </Server>
        </Host>
    </MODE>
  </DAAC>
</HypericControl>
```

Note: The Enabled attribute is meant to give the ability to disable the controlling of the resource or mode while preserving the information for possible later use. The StartNumber and StopNumber are used to determine the order of starting and stopping of servers. Having StartNumber/StopNumber values being the same will result in those servers being logically started/stopped together.

4.2.2.8 Configuration Verification

4.2.2.8.1 Overview

Configuration verification checks the Hyperic inventory configuration against a baseline and reports any discrepancies. The configuration verification is filtered by mode. The baseline to validate the specified mode configuration is defined through the xml file. DAACs are able to customize the xml file to define requirements for resource monitoring, alerts, and control actions.

XML Filename: <Mode>_<Configuration Category>_Config_Verification.xml

XML File Location (x4iil01 host): /usr/ecs/OPS/COTS/hyperic/cfg

4.2.2.8.2 Configuration Verification Page

The page displays configuration categories (Ex: Monitoring Defaults, COTS, CustomServers, DataProviders, OmDestinations, DUEs) filtered by mode. There is an option to display all configured resources or just those with configuration issues. Configuration verification is conducted against the specified mode baseline (defined in the customizable xml file).

The configuration category status displays the overall health of a configuration category. A green check symbol indicates the category has been fully verified and there are no existing issues. A red “x” symbol indicates that issues exist within the configuration category. A user is able to drill-down on each category to determine the exact issues that may result in a failed verification or to view the members of the category. An example of the drill-down capability is shown in figure 256.

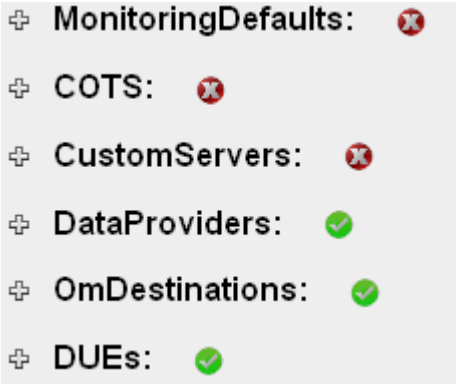

Figure 4.2.2.8.2-1. Configuration Category Status

Figure 4.2.2.8.2-2. Configuration Category Drill Down

Verify Mode Configuration

Step 1: Under the Analyze tab select “EEB Configuration Verification”

Figure 4.2.2.8.2-3. Verify Mode

Step 2: Select the mode to verify and choose the "Verify Configuration" option

4.2.2.8.3 Create Default Availability Alerts Command Line Utility

The Create default availability utility creates default availability alerts. This utility uses the information in the Configuration Verification XML files to create type based avalability alerts for all resources mentioned in those XML files. This should help in configuring hyperic and save some time manually configuring availability alerts for each individual resource. One limitation is that control actions cannot be configured this way so they would need to be entered manually via the GUI.

The usage for this utility is as follows:

EcMsSmCreateDefaultAlerts.pl [-delete] [-skipbackup]

Parameters:

-delete - optional parameter to delete previously created Availability alerts

-skipbackup - optional parameter to skip creating a backfup file

The -delete flag is optional and will delete previously created resource based availability alerts for the resources mentioned in the Configuration Verification XML files. The –skipbackup flag is also optional and will skip creating an XML backup of all previous alerts.

The utility needs a user created configuration file located at and called ~/.hq/client.properties. This file will have the following info

host=localhost

port=7443

secure=true

user=<hyperic GUI admin ie. hqadmin>

password=<password>

This utility is located under “/tools/common/hyperic/hqapi/hqapi1-3.1/bin”. The utility should be run by the “hyperic” user on the x4iil01 box.

4.2.2.3 Outputs

None

4.2.2.4 Event and Error Messages

None

4.2.2.5 Reports

None

This page intentionally left blank.