



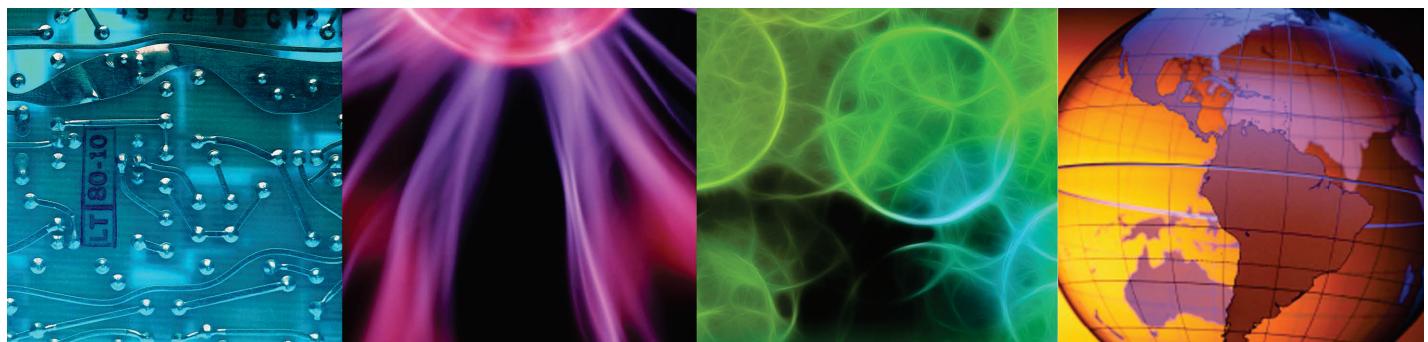
IBM Training

IBM Tivoli Netcool/OMNIbus 8.1 Installation and Configuration

Student Exercises

Course code TN025 ERC 1.0

November 2014



Cloud & Smarter Infrastructure

All files and material for this course are IBM copyright property covered by the following copyright notice.

© Copyright IBM Corp. 2014. All Rights Reserved.

US Government Users Restricted Rights: Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

IBM, the IBM logo, and ibm.com are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

The information contained in this publication is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this publication, it is provided AS IS without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this publication or any other materials. Nothing contained in this publication is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

References in this publication to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in this presentation may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth, savings or other results.



Contents

About these exercises	v
Server access	v
IP addresses	v
User IDs and passwords	v
Software	vi
Starting and stopping Security Directory Server	vi
Starting and stopping DB2	vi
1 Basic Tivoli Netcool/OMNIbus components and concepts exercises.....	1
There are no student exercises for this unit.	
2 Installation prerequisites and planning exercises	2
Exercise 1 Validating the host1 server configuration	2
Exercise 2 Validating the host2 server configuration	8
3 Netcool/OMNIbus core installation exercises	13
Exercise 1 Configuring the user environment	13
Exercise 2 Installing the core components	15
Exercise 3 Configuring initial function	18
Initial Configuration wizard	18
Verifying the initial configuration	22
Troubleshooting the ObjectServer	26
Verifying basic ObjectServer function	27
Adding a password to the root ObjectServer user	30
Configuring the Firefox plug-in	30
4 Installing Netcool/OMNIbus Web GUI exercises	32
Exercise 1 Installing Netcool/OMNIbus Web GUI	32
Installing Jazz for Service Management	32
Installing Web GUI	36
Exercise 2 Verifying access to Netcool/OMNIbus Web GUI	43
Exercise 3 Configuring Web GUI to automatically start	50
Exercise 4 Configuring LDAP as an authentication source	52
Removing the ObjectServer user repository	52
Adding the LDAP user repository	59
Configuring Dashboard Application Services Hub to allow logins if LDAP is down	70
Configuring ObjectServer synchronization	74
Configuring default users and groups	76

5 Basic ObjectServer administration exercises	83
Exercise 1 Overview of the Netcool/OMNibus directory structure	83
Exercise 2 Modifying the event record	88
Exercise 3 Creating users, and groups	94
6 Tools and automations exercises	102
Exercise 1 Creating a Web GUI tool	102
Exercise 2 Creating a custom trigger	111
7 Common integrations exercises	117
Exercise 1 Installing the Syslog probe	117
Exercise 2 Installing the SNMP probe	125
Exercise 3 Installing the Netcool Knowledge Library	132
8 ObjectServer high availability exercises	137
Exercise 1 Creating the backup ObjectServer	137
Initial Configuration wizard	140
Verifying the initial configuration	144
Configuring the Firefox plug-in	149
Exercise 2 Replicating ObjectServer configuration	150
Exercise 3 Configuring the ObjectServer gateway	152
Exercise 4 Configuring Web GUI for ObjectServer high availability	159
Exercise 5 Verifying ObjectServer high availability	161
Exercise 6 Enabling triggers for high availability	168
Exercise 7 Configuring probes for ObjectServer high availability	172
9 Process control exercises	179
Exercise 1 Configuring the Pluggable Authentication Module (PAM) on host1	179
Exercise 2 Configuring the Pluggable Authentication Module (PAM) on host2	182
Exercise 3 Configuring the process activity daemons	185
Exercise 4 Modifying the ObjectServer process activity credentials	187
Exercise 5 Working with the process activity daemon	189
Exercise 6 Verifying the auto-start configuration	194
10 Event archiving exercises	196
Exercise 1 Installing Tivoli Common Reporting	196
Exercise 2 Creating the event archive database structure	203
Exercise 3 Configuring the database gateway	207
Exercise 4 Installing the Netcool/OMNibus report package	221



About these exercises

This course shows you how to perform a complete Netcool®/OMNIbus Version 8.1 installation based on best practices.

Server access

The course uses two Linux servers that are running in a VMware virtual machine. You access these servers through the VMware console.

IP addresses

The student images each contain one Ethernet interface (eth0), configured as host-only, and has a static IP address assigned:

- **IP address:** eth0 192.168.100.160
- **Host name:** host1.tivoli.edu
- **IP address:** eth0 192.168.100.161
- **Host name:** host2.tivoli.edu

User IDs and passwords

Type	User ID	Password	Usage
Linux	root	object00	Linux super user
Linux	db2inst1	object00	Owns DB2® instance
Linux	netcool	object00	Student user ID

Unless directed in the instructions, all command-line activities are run as the Linux **netcool** user.

Software

The host1 image is configured with the following software:

- IBM® Security Directory Server V6.3.1

The host2 image is configured with the following software:

- IBM DB2® 10.5 Enterprise Server Edition

All the installation files that you require for student exercises can be found in various subdirectories in the following directory:

/software

Starting and stopping Security Directory Server

Security Directory Server is installed on the host1 image as the **root** user, and is configured to start automatically when the system starts:

`/etc/init.d/ibmslapd`

The Security Directory Server instance can be manually started or stopped on the host1 image with the **root** user ID as follows:

```
/etc/init.d/ibmslapd start  
/etc/init.d/ibmslapd stop
```

Starting and stopping DB2

DB2 is installed as the **root** user on the host2 image, and is configured to start automatically when the system starts:

`/etc/init.d/db2_tcr`

The DB2 instance can be manually started or stopped on the host2 image with the **root** user ID as follows:

```
/etc/init.d/db2_tcr start  
/etc/init.d/db2_tcr stop
```

The DB2 instance can also be manually started or stopped on the host2 image with the **db2inst1** user ID as follows:

```
db2start  
db2stop
```



1 Basic Tivoli Netcool/OMNIbus components and concepts exercises

There are no student exercises for this unit.



2 Installation prerequisites and planning exercises

The exercises in this unit validate the host configurations before installing the Netcool/OMNibus components.



Note: You are the **root** user for the steps in this unit.

Exercise 1 Validating the host1 server configuration

Netcool®/OMNibus has several system requirements that must be met before the software is installed. These requirements include such things as: server disk and memory capacity, operating system, system patches, and third-party software. The requirements vary by operating system, and are detailed in *IBM Tivoli Netcool/OMNibus: Installation and Deployment Guide*.

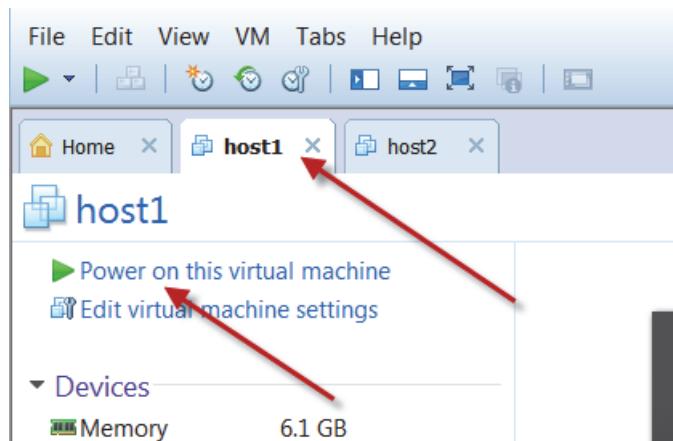
To automate the validation process, IBM provides the *prerequisite scanner*. IBM® Prerequisite Scanner is a stand-alone prerequisite checking tool that analyzes system environments before the installation or upgrade of a Tivoli® product or IBM solution.

The IBM Prerequisite Scanner verifies that the required hardware and software for Tivoli Netcool/OMNibus is present on the host computer.

Task 1 Starting the image

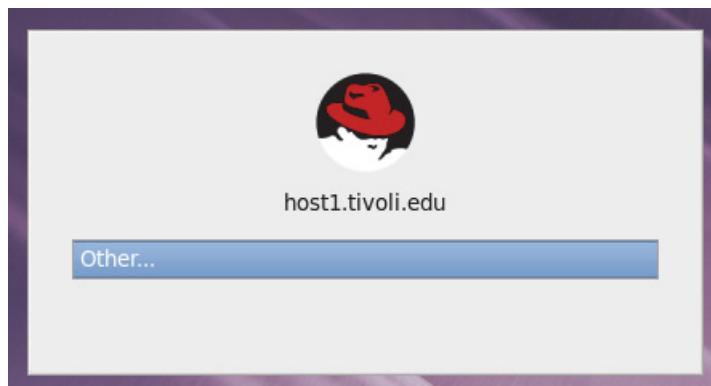
Depending on how this course is delivered, the host1 image might already be running. If the image is running, skip the steps for powering on the images. If the image is not running, use the following steps to start the image:

1. Locate the **host1** tab in the VMware console.
 - a. Click the **host1** tab to select it.
 - b. Click the line that is labeled **Power on this virtual machine**.



The image takes several minutes to initiate. The login screen opens when the image is available.

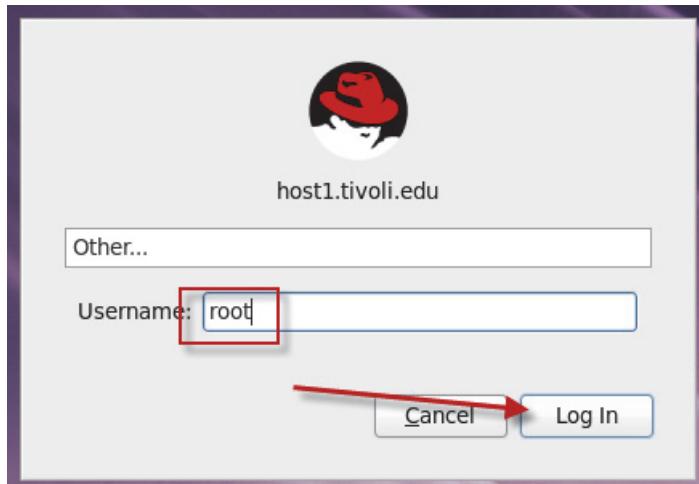
2. Log in as the *root* user:
 - a. Click **Other**.



2 Installation prerequisites and planning exercises

Exercise 1 Validating the host1 server configuration

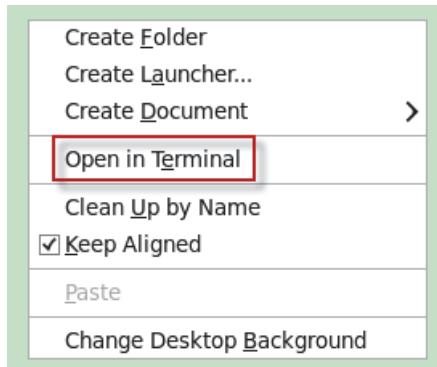
- b. Enter **root** as the user name and click **Log In**.



The password is *object00*.

The Linux console window opens.

3. Open a terminal window:
 - a. Place your cursor anywhere in the console window.
 - b. Right-click and select **Open in Terminal**.



A terminal window opens.

 Hint: Repeat the previous steps if you want to open more terminal windows.

Task 2 Installing the prerequisite scanner

The prerequisite scanner is not bundled with Netcool/OMNibus. It is distributed as a UNIX compressed file. Perform the following steps to install the prerequisite scanner:



Note: The Prerequisite Scanner is available for download from the IBM Fix Central website here:
<http://www-933.ibm.com/support/fixcentral/swg/selectFixes?parent=ibm~Tivoli&product=ibm/Tivoli/Prerequisite+Scanner&release>All&platform>All&function=all>

1. Change to the required directory:

```
cd /software/prs
```

2. Expand the compressed file:

```
tar -xvf precheck_unix_20140825.tar
```

Task 3 Running the prerequisite scanner

The Netcool/OMNibus components are categorized as follows:

NOA: Netcool/OMNibus Administrator GUI

NOC: Netcool/OMNibus complete (all core components).

NOD: Netcool/OMNibus desktop (native client)

NOP: Netcool/OMNibus probes (and gateways)

NOS: Netcool/OMNibus server (ObjectServers)

NOW: Netcool/OMNibus Web GUI

NPA: Netcool/OMNibus Process Agent components

All core components are installed on the host1 server in this course. Therefore, the NOC category is used. The Web GUI component is installed on host2.



Important: In a production environment, the components are typically distributed across multiple servers.

1. Change to the required directory:

```
cd /software/prs
```

2. Run the scanner:

```
./prereq_checker.sh "NOC 08010000" detail
```

IBM Prerequisite Scanner

2 Installation prerequisites and planning exercises

Exercise 1 Validating the host1 server configuration

```
Version: 1.2.0.13
Build : 20140825
OS name: Linux
User name: root

Machine Information
Machine name: host1.tivoli.edu

.
.
.

os.package gtk2.x86_64          PASS      gtk2-2.18.9-12.el6.x86_64
gtk2-2.18.9-4.el6.x86_64+
os.package libjpeg.x86_64         FAIL      Unavailable
libjpeg-6b-46.el6.x86_64+

Aggregated Properties for Scanned Products:
Property          Result    Found          Expected
=====
/                PASS     23552.00MB      910MB
Memory           PASS     4.98GB        4.00GB

Overall result: FAIL (NOC 08010000: FAIL)

Detailed results are also available in /software/prs/result.txt
```

The scanner presents its detailed output. Verify that all checks are flagged as PASS. This output verifies that the host system meets all of the requirements to install Netcool/OMNIbus core, desktop, and server components.



Important: The scan on host1 fails due to a missing package. The missing file is not an issue, as another package replaces the missing package on this image.

Task 4 Verifying the user environment

The software is installed as the netcool user. The netcool user belongs to the ncoadmin group. To facilitate the workshop, the netcool user and the ncoadmin group are already created.

1. Examine the *ncoadmin* group:

```
more /etc/group | grep ncoadmin
```

```
ncoadmin:x:501:
```

The *ncoadmin* group is a requirement of Netcool/OMNIbus Process Activity. The group ID number (GID) is not important. Only the name *ncoadmin* is important.

2. Examine the *netcool* user:

```
more /etc/passwd | grep netcool
```

```
netcool:x:500:501::/home/netcool:/bin/bash
```

The *netcool* user does not possess any special authority or privileges. The only unique characteristic is that the user is a member of the *ncoadmin* group.

Task 5 Configuring the operating system

You must make a number of operating system changes that the various components that are installed for this workshop require.

1. Create the installation directory.

```
cd /opt/IBM  
mkdir tivoli
```

2. Set the ownership to the *netcool* user.

The Netcool/OMNIbus software is installed as the non-root user *netcool*.

```
chown -R netcool:ncoadmin tivoli
```

The directory exists, is empty, and owned by the *netcool* user.

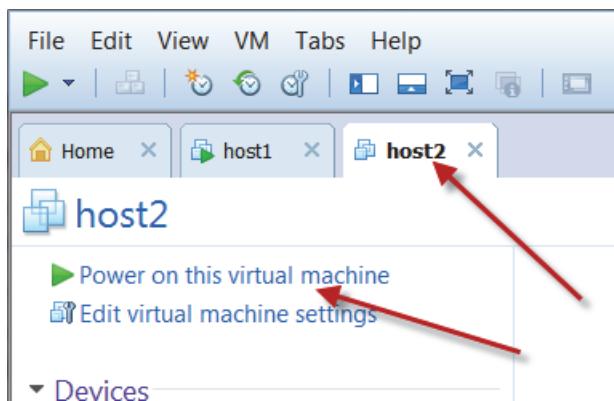
3. Log out of host1 as the *root* user.

The system requirements for the host1 image are verified and the initial configuration is complete.

Exercise 2 Validating the host2 server configuration

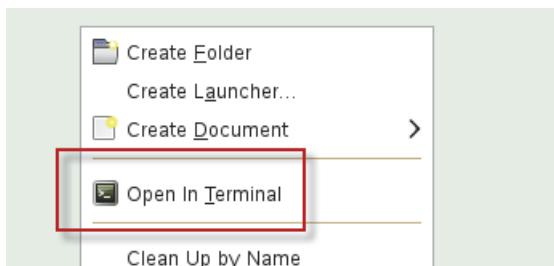
Task 1 Starting the image

1. Locate the **host2** tab in the VMware console.
 - a. Click the **host2** tab to select it.
 - b. Click the line that is labeled **Power on this virtual machine**.

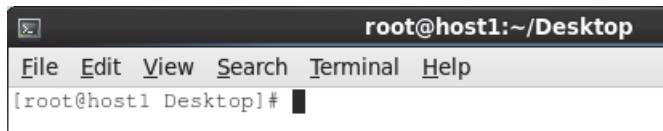


The image takes several minutes to initiate. The login screen appears when the image is available.

2. Log in as the **root** user.
The password is *object00*.
The Linux console window opens.
3. Open a terminal window:
 - a. Place your cursor anywhere in the console window.
 - b. Right-click and select **Open in Terminal**.



A terminal window opens.



Hint: Repeat the previous steps if you want to open more terminal windows.

Task 2 Installing the prerequisite scanner

1. Change to the required directory:

```
cd /software/prs
```

2. Expand the compressed file:

```
tar -xvf precheck_unix_20140825.tar
```

Task 3 Running the prerequisite scanner

Some core components are installed on the host2 server in this course in addition to the Web GUI and Tivoli Common Reporting components. Therefore, the NOC category, NOW category, and TCR category are used.



Important: In a production environment, the components are typically distributed across multiple servers.

1. Change to the required directory:

```
cd /software/prs
```

2. Run the scanner to check the core requirements:

```
./prereq_checker.sh "NOC 08010000" detail
```

```
IBM Prerequisite Scanner
Version: 1.2.0.13
Build : 20140825
OS name: Linux
User name: root
```

```
Machine Information
Machine name: host2.tivoli.edu
```

```
.
```

```
.
```

```
.
```

2 Installation prerequisites and planning exercises

Exercise 2 Validating the host2 server configuration

```
os.package.gtk2.x86_64          PASS      gtk2-2.18.9-12.el6.x86_64
gtk2-2.18.9-4.el6.x86_64+
os.package.libjpeg.x86_64        FAIL      Unavailable
libjpeg-6b-46.el6.x86_64+
```

Aggregated Properties for Scanned Products:

Property	Result	Found	Expected
=====	=====	=====	=====
/	PASS	23552.00MB	910MB
Memory	PASS	4.98GB	4.00GB

Overall result: FAIL (NOC 08010000: FAIL)

Detailed results are also available in /software/prs/result.txt

The scanner presents its detailed output. Verify that all checks are flagged as PASS. This output verifies that the host system meets all of the requirements to install Netcool/OMNIbus core, desktop, and server components.



Important: The NOC scan on host2 fails due to a missing package. The missing file is not an issue, as another package replaces the missing package on this image.

3. Run the scanner to check the Web GUI requirements:

```
./prereq_checker.sh "NOW 08010000" detail
```

IBM Prerequisite Scanner

Version: 1.2.0.13

Build : 20140825

OS name: Linux

User name: root

Machine Information

Machine name: host2.tivoli.edu

Serial number: VMware-56 4d 54 68 f7 21 f0 f8-41 23 f6 73 15 ef 36 55

Scenario: Prerequisite Scan

NOW - Tivoli Netcool/OMNIbus Web GUI [version 08010000] :

Property	Result	Found	Expected
=====	=====	=====	=====
os.RAMSize	PASS	5.7GB	2GB
intel.cpu	FAIL	2.20GHz	2.4GHz
os.space.opt_root	PASS	23552MB	
[dir:root=/opt/IBM] 700MB			
os.space.opt_nonroot	PASS	NOT_REQ_CHECK_ID	
[dir:non_root=USERHOME/IBM] 700MB			
os.space.imshared	PASS	23552MB	
[dir:root=/opt;non_root=USERHOME] 100MB			

Aggregated Properties for Scanned Products:

Property	Result	Found	Expected
	=====	=====	=====
/	PASS	23552.00MB	800MB

Overall result: FAIL (NOW 08010000: FAIL)

The scanner presents its detailed output. Verify that all checks are flagged as PASS. This output verifies that the host system meets all of the requirements to install Netcool/OMNIbus Web GUI components.



Important: The NOW scan on host2 fails due to CPU speed. The CPU speed is not an issue for the class exercises.

- Run the scanner to check the Tivoli Common Reporting requirements:

```
./prereq_checker.sh "TCR 03010000" detail
```

```
IBM Prerequisite Scanner
Version: 1.2.0.13
Build : 20140825
OS name: Linux
User name: root
```

```
Machine Information
Machine name: host2.tivoli.edu
Serial number: VMware-56 4d 54 68 f7 21 f0 f8-41 23 f6 73 15 ef 36 55
```

Scenario: Prerequisite Scan

TCR - Tivoli Common Reporting [version 03010000]:

Property	Result	Found	Expected
	=====	=====	=====
OS Version	PASS	Red Hat Enterprise Linux Server rel... AIX V6.1 AIX V7.1	
Red Hat Enterprise Linux Server release 5.6+			Red Hat

Enterprise Linux Server release 6.*

SuSE Linux

Enterprise Server 10 (*)

SuSE Linux

Enterprise Server 11 (*)

os.architecture

64-bit

64-bit

.

.

.

Aggregated Properties for Scanned Products:

Property	Result	Found	Expected
	=====	=====	=====
/	PASS	20.00GB	5.68GB

Overall result: PASS (TCR 03010000: PASS)

The scanner presents its detailed output. Verify that all checks are flagged as PASS. This output verifies that the host system meets all of the requirements to install Tivoli Common Reporting components.

Task 4 Verifying the user environment

The software is installed as the netcool user. The netcool user belongs to the ncoadmin group. To facilitate the workshop, the netcool user, and the ncoadmin group are already created.

1. Examine the *ncoadmin* group:

```
more /etc/group | grep ncoadmin
```

```
ncoadmin:x:501:
```

The *ncoadmin* group is a requirement of Netcool/OMNIbus Process Activity. The group ID number (GID) is not important. Only the name *ncoadmin* is important.

2. Examine the *netcool* user:

```
more /etc/passwd | grep netcool
```

```
netcool:x:500:501::/home/netcool:/bin/bash
```

The *netcool* user does not possess any special authority or privileges. The only unique characteristic is that the user is a member of the *ncoadmin* group.

Task 5 Configuring the operating system

You must make a number of operating system changes that the various components that are installed for this workshop require.

1. Create the installation directories.

```
cd /opt/IBM  
mkdir netcool  
mkdir tivoli
```

2. Set the ownership to the *netcool* user.

The Netcool/OMNIbus software is installed as the non-root user netcool.

```
chown -R netcool:ncoadmin netcool
```

```
chown -R netcool:ncoadmin tivoli
```

The directories exists, are empty, and owned by the *netcool* user.

3. Log out of *host2* as the *root* user.

The system requirements for the host2 image are verified and the initial configuration is complete. T



3 Netcool/OMNibus core installation exercises

In this unit, you learn how to install the Netcool/OMNibus core component, create an ObjectServer, and validate the installation.

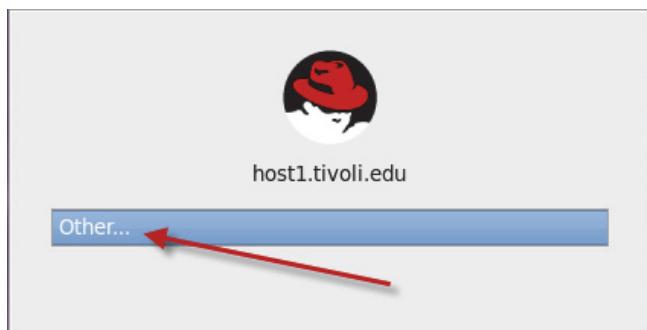


Note: You are the **netcool** user for the steps in this unit.

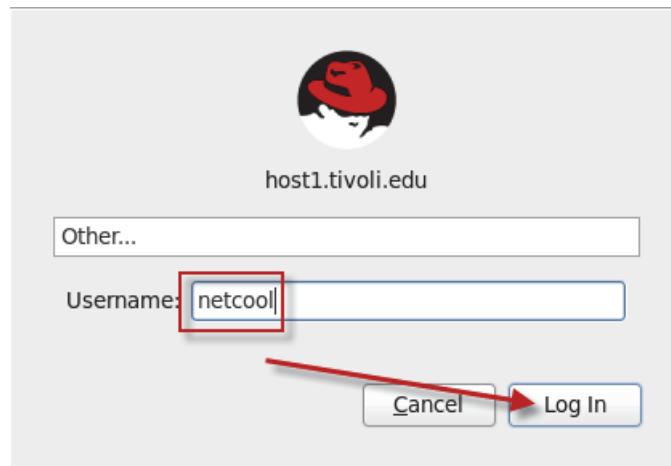
Exercise 1 Configuring the user environment

You logged out of the **host1** image at the end of the previous exercise. The login screen is available.

1. Log in as the **netcool** user:
 - a. Click **Other**.



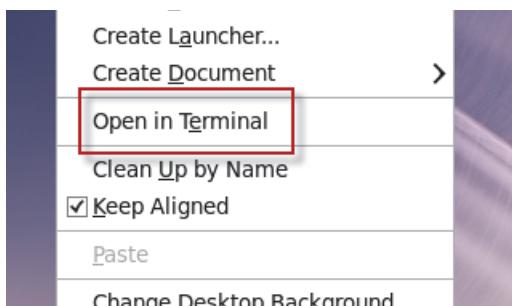
- b. Enter *netcool* as the user.



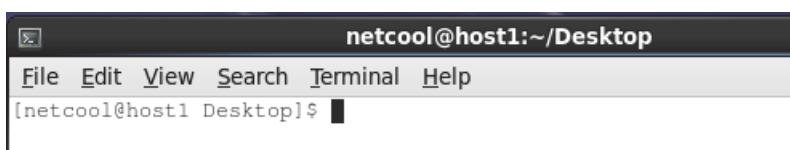
The password is **object00**.

The Linux console window opens.

2. Open a terminal window:
 - a. Place your cursor anywhere in the console window.
 - b. Right-click and select **Open in Terminal**.



A terminal window opens.



Hint: Repeat the previous steps if you want to open more terminal windows.

3. Configure environment variables:

```
cd /workshop/netcool
```

```
cat .bashrc >> /home/netcool/.bashrc
```

```
source /home/netcool/.bashrc
```

4. Verify environment variables:

```
env | grep IBM
```

```
PATH=/opt/IBM/tivoli/netcool/bin:/opt/IBM/tivoli/netcool/omnibus/bin:/opt/
IBM/tivoli/netcool/omnibus/probes:/usr/lib64/qt-3.3/bin:/usr/local/bin:/us
r/bin:/bin:/usr/local/sbin:/usr/sbin:/sbin:/home/netcool/bin
NCHOME=/opt/IBM/tivoli/netcool
OMNIHOME=/opt/IBM/tivoli/netcool/omnibus
```

Exercise 2 Installing the core components

In this exercise, you install the Netcool®/OMNIbus core components. You are installing all of Netcool/OMNIbus core on a single server, which is not typically done in a production environment.

You have various options for installing Netcool/OMNIbus core components. The following steps demonstrate how to use the Netcool/OMNIbus installation script to install IBM® Installation Manager and Netcool/OMNIbus with the graphical option.

1. Expand the core installation file as follows:

```
cd /software/omnibus
```

```
unzip OMNIbus-v8.1-Core.linux64.zip
```

2. Install the core software:

```
./install_gui.sh
```

a. Verify that the Netcool/OMNIbus package is selected for installation.

b. Scroll to the bottom of the view and click **Next**.

Install Packages

Select packages to install:

Installation Packages	Status	Vendor
IBM® Installation Manager	Will be installed	IBM
Version 1.7.2	Will be installed	IBM
IBM Tivoli Netcool/OMNIbus	Will be installed	IBM
Version 8.1.0.0	Will be installed	IBM

- c. Accept the license agreement and click **Next**.

I accept the terms in the license agreements

I do not accept the terms in the license agreements

- d. Leave the default locations for Installation Manager, and click **Next**.
 - e. Leave the default installation directory, and click **Next**.
 - f. Leave all of the features selected, and click **Next**.
 - g. Leave the option for **Data migration** cleared, and click **Next**.



Hint: The option is used when you upgrade from a previous version of Netcool/OMNIbus.

- h. Review the installation summary and click **Install**.

Install Packages

Review the summary information.

Install Licenses Location Features **Summary**

Target Location

Package Group Name: IBM Tivoli Netcool OMNIbus
 Installation Directory: /opt/IBM/tivoli/netcool
 Shared Resources Directory: /home/netcool/IBM/IBMIMShared

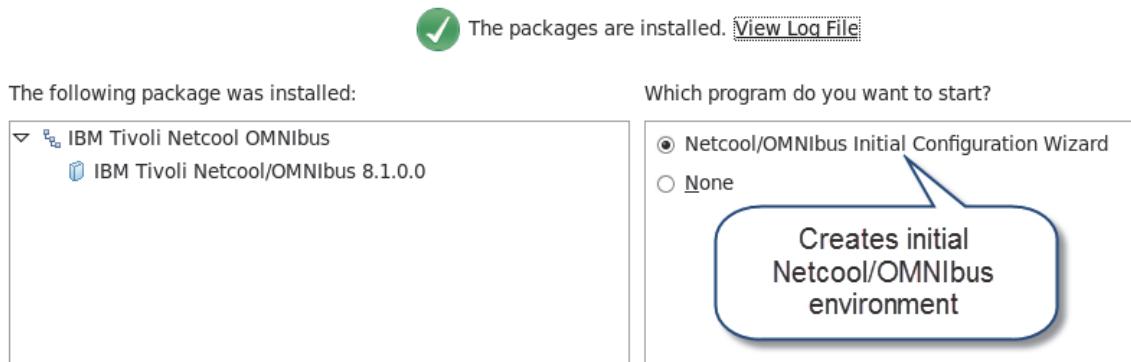
Packages

Packages	
	IBM® Installation Manager 1.7.2
	IBM Tivoli Netcool/OMNIbus 8.1.0.0
▷	Administrator components
▷	Operator components
▷	Server components
▷	Process control components
▷	Probe and gateway components
▷	Extensions



Hint: An installation on most servers runs approximately 10 minutes.

- i. Leave the option selected to run the configuration wizard and click **Finish**.



Verify that the installation is successful.

- j. Click **Done** to exit the installation wizard.
k. Click **Exit** to close the launchpad utility.

Exercise 3 Configuring initial function

The next step in the installation process is to define, create, and start an ObjectServer. The standard multitier architecture configuration is the prebuilt, best practice Netcool/OMNibus configuration that comes with the product, which can deploy one-tier, two-tier, or three-tier systems. The aggregation layer represents the core of any Netcool/OMNibus deployment. It can stand alone as a *single-tier* system. Therefore, the starting point for any Netcool/OMNibus installation is an aggregation ObjectServer. To minimize the customization that is required, you can use the multitier ObjectServer naming convention. The usual name for the primary aggregation ObjectServer is AGG_P.

Initial Configuration wizard

At the conclusion of the installation process, the installation wizard started automatically.



Hint: You start the configuration wizard manually with the following command:

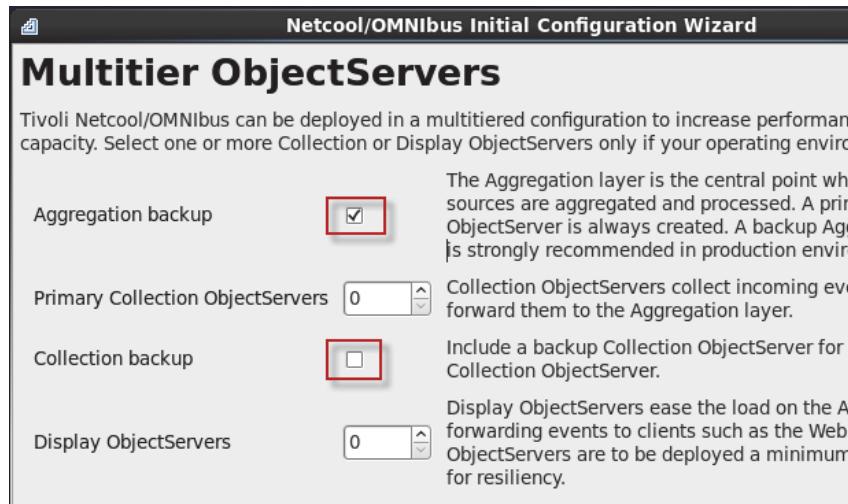
```
/opt/IBM/tivoli/netcool/omnibus/bin/nco_icw
```

1. Complete the configuration with the wizard as follows:
 - a. Scroll to the bottom of the view and click **Next**.



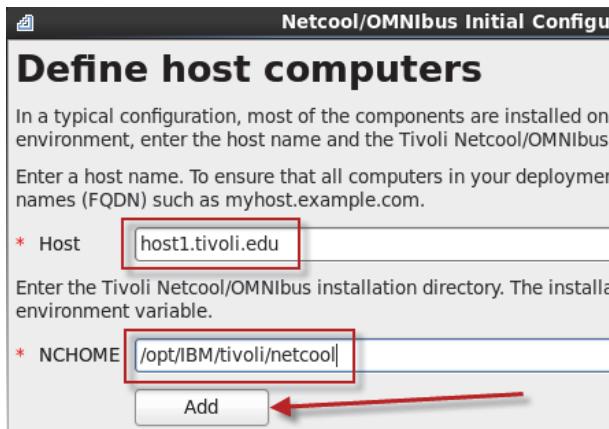
- b. Leave the option selected to create a new configuration and click **Next**.
- c. Leave the check for **Aggregation backup** and clear the check for **Collection backup**.

d. Click **Next**.



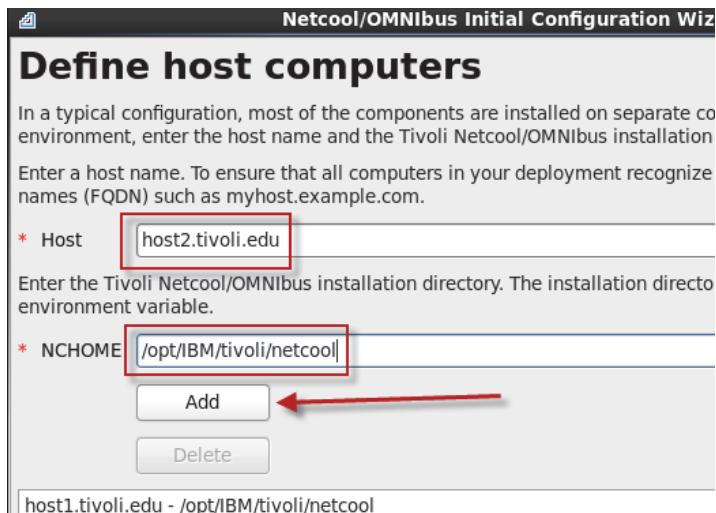
e. Enter **host1.tivoli.edu** and **/opt/IBM/tivoli/netcool**.

f. Click **Add**.



g. Enter **host2.tivoli.edu** and **/opt/IBM/tivoli/netcool**.

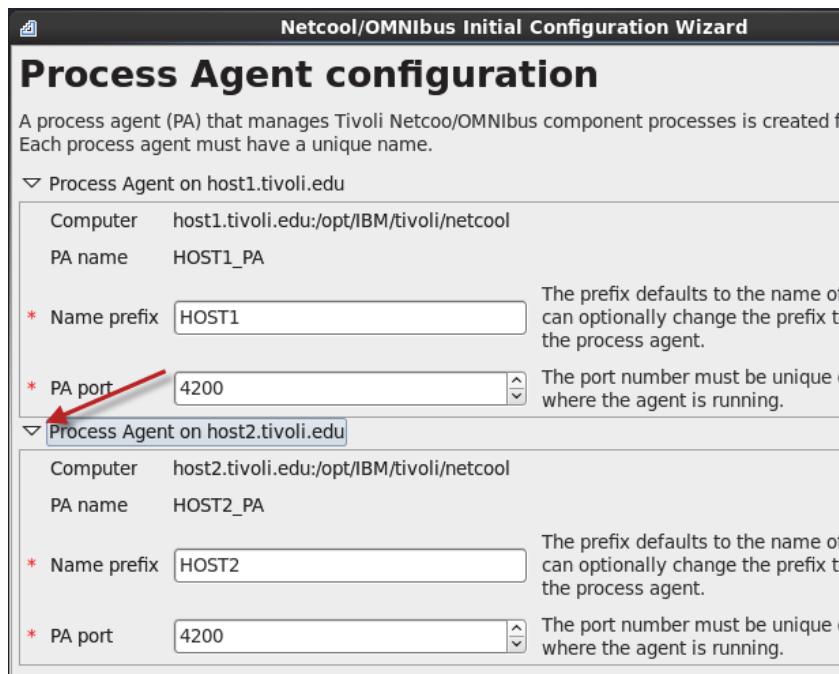
h. Click **Add**.



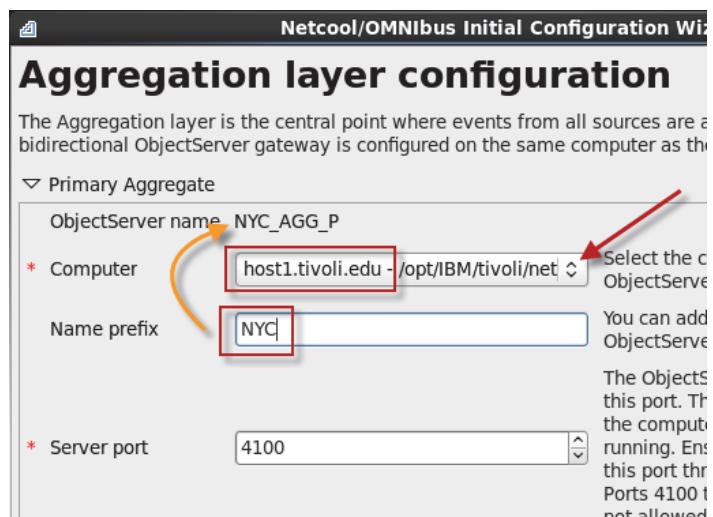
- i. Verify that both entries look like this example and click **Next**.



- j. Verify that the settings for Process Agents look like this example and click **Next**.

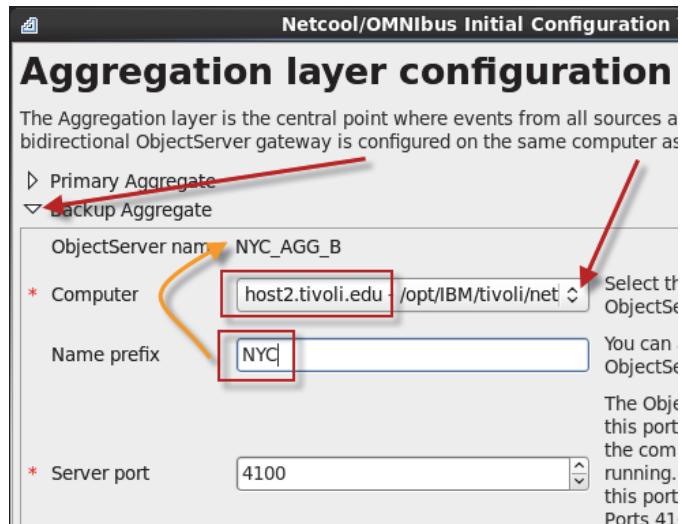


- k. Select **host1.tivoli.edu** for Computer and enter **NYC** for the Name prefix.



The primary ObjectServer name is set to AGG_P and cannot be changed. You can enter text in the **Name prefix** field, and that text adds a prefix to AGG_P.

- I. Expand the option for **Backup Aggregate**.
- m. Select **host2.tivoli.edu** for the computer and enter **NYC** for the name prefix.



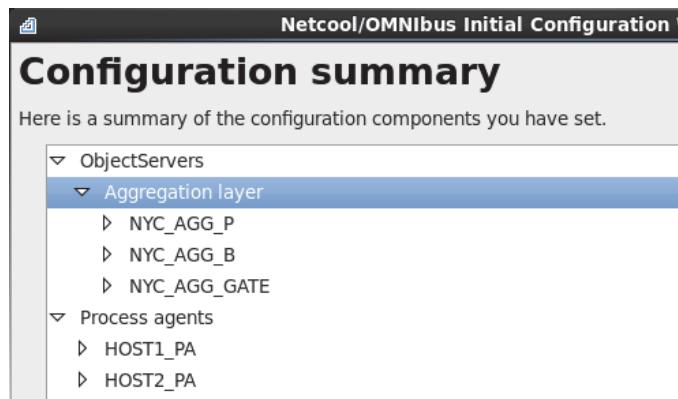
The backup ObjectServer name is set to AGG_B and cannot be changed. You can enter text in the **Name prefix** field, and that text adds a prefix to AGG_B.

- n. Scroll down in the view, enter **NYC** for the gateway prefix, and click **Next**.

Gateway name	NYC_AGG_GATE
Gateway prefix	<input type="text" value="NYC"/>
* Gateway port	4300

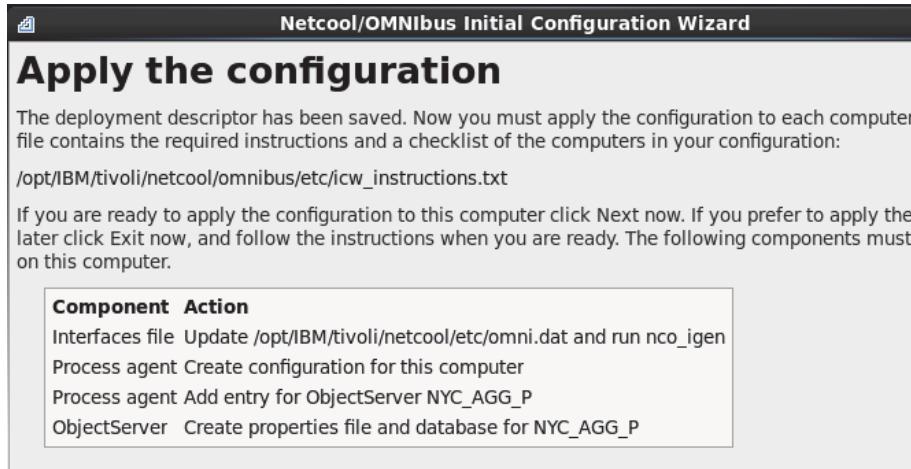
The gateway name is set to AGG_GATE and cannot be changed. You can enter text in the **Gateway prefix** field, and that text adds a prefix to AGG_GATE.

- o. Review the configuration summary and click **Next**.

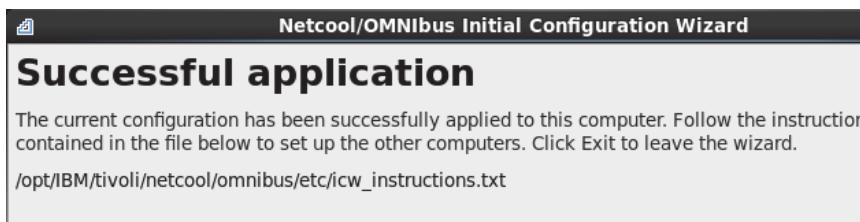


- p. Click **Next**.

- q. Click **Next** to apply the configuration to **host1**.



- r. Verify that the configuration is successfully applied and click **Exit**.



The configuration is applied to the host1 system. You configure host2 in a subsequent exercise.

Verifying the initial configuration

The initial configuration wizard performs a number of steps that you can perform manually.

1. Examine the communications file as follows:

```
cd /opt/IBM/tivoli/netcool/etc
more omni.dat
```

```
[netcool@host1 etc]$ more omni.dat
[NYC_AGG_P]
{
    Primary: host1.tivoli.edu 4100
}
[NYC_AGG_B]
{
    Primary: host2.tivoli.edu 4100
}
[AGG_V]
{
    Primary: host1.tivoli.edu 4100
    Backup: host2.tivoli.edu 4100
}
[NYC_AGG_GATE]
{
    Primary: host2.tivoli.edu 4300
}
[HOST1_PA]
{
    Primary: host1.tivoli.edu 4200
}
[HOST2_PA]
{
    Primary: host2.tivoli.edu 4200
}
```

primary ObjectServer

backup ObjectServer

virtual ObjectServer

synchronizer gateway

host1 Process Agent

host2 Process Agent

The omni.dat file contains the entries for the primary ObjectServer (NYC_AGG_P), the backup ObjectServer (NYC_AGG_B), and the *virtual* ObjectServer (AGG_V). The file also contains the entry for the synchronizer gateway (NYC_AGG_GATE), and the two process agents (HOST1_PA and HOST2_PA).

2. Examine the interfaces file as follows:

```
more interfaces.linux2x86

#
# interfaces file generated by nco_igen on Tue Jun 10 17:18:47 2014
#
#
# DO NOT AMEND THIS FILE !!!!!!!
# MAKE CHANGES IN THE $NCHOME/etc/omni.dat FILE AND
# RE-GENERATE THIS FILE WITH NCO_IGEN
#
#
# NYC_AGG_P => host1.tivoli.edu 4100
NYC_AGG_P
    master tcp sun-ether host1.tivoli.edu 4100
    query tcp sun-ether host1.tivoli.edu 4100

# NYC_AGG_B => host2.tivoli.edu 4100
NYC_AGG_B
    master tcp sun-ether host2.tivoli.edu 4100
    query tcp sun-ether host2.tivoli.edu 4100

# AGG_V => host1.tivoli.edu 4100
AGG_V
    master tcp sun-ether host1.tivoli.edu 4100
    query tcp sun-ether host1.tivoli.edu 4100
    master tcp sun-ether host2.tivoli.edu 4100
    query tcp sun-ether host2.tivoli.edu 4100

# NYC_AGG_GATE => host2.tivoli.edu 4300
NYC_AGG_GATE
    master tcp sun-ether host2.tivoli.edu 4300
    query tcp sun-ether host2.tivoli.edu 4300

# HOST1_PA => host1.tivoli.edu 4200
HOST1_PA
    master tcp sun-ether host1.tivoli.edu 4200
    query tcp sun-ether host1.tivoli.edu 4200

# HOST2_PA => host2.tivoli.edu 4200
```

```

HOST2_PA
  master tcp sun-ether host2.tivoli.edu 4200
  query tcp sun-ether host2.tivoli.edu 4200

```

```

#
# EOF
#

```

The interfaces file is created from the **omni.dat** file by running the nco_igen utility. The configuration wizard runs the utility.

3. Verify that the process agent is configured as follows:

- a. Examine the process agent configuration file.

```

cd /opt/IBM/tivoli/netcool/omnibus/etc
more nco_pa.conf

```

```

#NCO_PA3
#
# Process Agent Daemon Configuration File 1.1
#
# List of processes
#
nco_process 'MasterObjectServer'
{
    Command '$OMNIHOME/bin/nco_objserv -name NYC_AGG_P -pa HOST1_PA' run as 0
    Host ='host1.tivoli.edu'
    Managed = True
    RestartMsg ='${NAME} running as ${EUID} has been restored on
    ${HOST}.'
    AlertMsg ='${NAME} running as ${EUID} has died on ${HOST}.
    '

    RetryCount = 0
    ProcessType = PaPA_AWARE
}

#
# List of Services
#
# NOTE: To ensure that the service is started automatically, change the
#       "ServiceStart" attribute to "Auto".
#
nco_service 'Core'
{
    ServiceType = Master
    ServiceStart = Auto
    process 'MasterObjectServer' NONE
}

```

```
}

#
# This service should be used to store processs that you want to temporarily
# disable. Do not change the ServiceType or ServiceStart settings of this
# process.
#
nco_service 'InactiveProcesses'
{
    ServiceType=Non-Master
    ServiceStart=Non-Auto
}

#
# ROUTING TABLE
#
# 'user'      - (optional) only required for secure mode PAD on target host
#                 'user' must be member of UNIX group 'ncoadmin'
# 'password'   - (optional) only required for secure mode PAD on target
host
#                 use nco_pa_crypt to encrypt.

nco_routing
{
    host 'host1.tivoli.edu' 'HOST1_PA'
    host 'host2.tivoli.edu' 'HOST2_PA'
}
```

The wizard creates the process agent configuration file. The wizard assumes that the processes under the control of the process agent are run as the **root** user. Most users want to limit the processes that run as **root**. In the next step, you modify the configuration file to run the ObjectServer as the **netcool** user.

- b. Determine the UID value of the **netcool** user.

```
more /etc/passwd | grep netcool
```

```
netcool:x:500:501::/home/netcool:/bin/bash
```

In this example, the UID for the *netcool* user is 500.

- c. Modify the configuration file:

```
cd /opt/IBM/tivoli/netcool/omnibus/etc
gedit nco_pa.conf
```

- i. Locate the following line:

```
Command '$OMNIHOME/bin/nco_objserv -name NYC_AGG_P -pa HOST1_PA' run as 0
```

- ii. Change run as 0 to run as 500.

```
Command '$OMNIHOME/bin/nco_objserv -name NYC_AGG_P -pa HOST1_PA' run as
500
```

- d. Save the changes and exit gedit.

4. Verify that the primary ObjectServer is created as follows:

- a. Examine the ObjectServer property file:

```
cd /opt/IBM/tivoli/netcool/omnibus/etc
more NYC_AGG_P.props
```

- b. Examine the ObjectServer database directory:

```
cd /opt/IBM/tivoli/netcool/omnibus/db/NYC_AGG_P
ls -1
```

```
master_store.tab
table_store.tab
```

5. Start the process agent:

```
nco_pad -name HOST1_PA
```



Hint: The directory is not required because the PATH environment variable contains this path:
/opt/IBM/tivoli/netcool/omnibus/bin

6. Verify that the ObjectServer is running:

```
nco_ping NYC_AGG_P
```

NCO_PING: Server available.

Troubleshooting the ObjectServer

There are typically a few reasons why a *new* ObjectServer fails to start.



Important: The use of the Initial Configuration Wizard usually eliminates all of the typical configuration issues.

1. There are issues with the interfaces file.

Issues with the interface file generally fall into one or more of the following areas:

- a. ObjectServer name is not correct.

The name must match exactly the name that is used when the ObjectServer is created:

```
nco_dbinit -server XXXX.
```

- b. The ObjectServer host is not correct.

The value for host can be a short name (host1) or fully qualified name (host1.tivoli.edu). In either case, the name must resolve to a valid IP address. The name resolution uses whatever is the local name resolution technique, such as hosts file, DNS, or other.

- c. The ObjectServer port is not valid.

This message is shown only if there is a port conflict, two components that attempt to use the same port number on the same server.

- d. Changes not applied.

When using the nco_xigen utility, you must click **Apply** to save the changes. A common mistake is to open the utility, add an ObjectServer entry, and then try to start the ObjectServer. The modified entry shows in the utility view, but it is not implemented until you click **Apply**.

2. The ObjectServer is not created correctly with nco_dbinit.

The most common mistake here is either misspelling the ObjectServer name or failing to specify the name in *uppercase* characters. However, in either case, the nco_dbinit utility seems to run correctly, but the ObjectServer fails to start.

3. The ObjectServer name is not correct on the nco_objserv command.

- a. The wrong command-line parameter specified.

A common mistake is to use the wrong command-line parameter as follows:

```
nco_objserv -server XXXX &
```

The correct syntax is:

```
nco_objserv -name XXXX &
```

- b. The ObjectServer name is not entered correctly.

Again, the ObjectServer must be all capital letters, and the name must match exactly what is shown in the interfaces file and what you used with the nco_dbinit utility.

The log file that is created by the ObjectServer is found here:

```
$OMNIHOME/log/XXXX.log
```

XXXX is the name of the ObjectServer. The ObjectServer is called NYC_AGG_P for the class exercise.

Verifying basic ObjectServer function

The Simnet probe allows for the automatic generation of incidents to simulate network events. The probe provides a convenient mechanism for verifying basic ObjectServer functions.



Important: The Simnet probe is bundled with Netcool/OMNibus. You must install all other probes individually.

1. Start the probe, and send events to the NYC_AGG_P ObjectServer as follows:

```
nco_p_simnet -server NYC_AGG_P &
```



Hint: The directory is not required because the PATH environment variable contains this path:
/opt/IBM/tivoli/netcool/omnibus/probes

2. Examine the simulated events.

- a. Start the native event list:

```
nco_event &
```

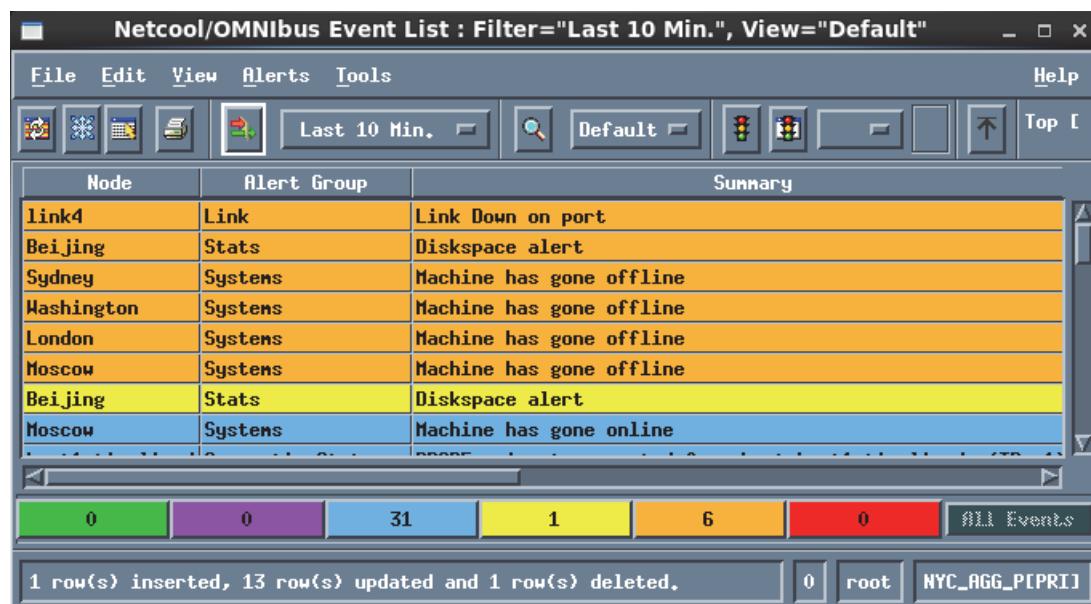
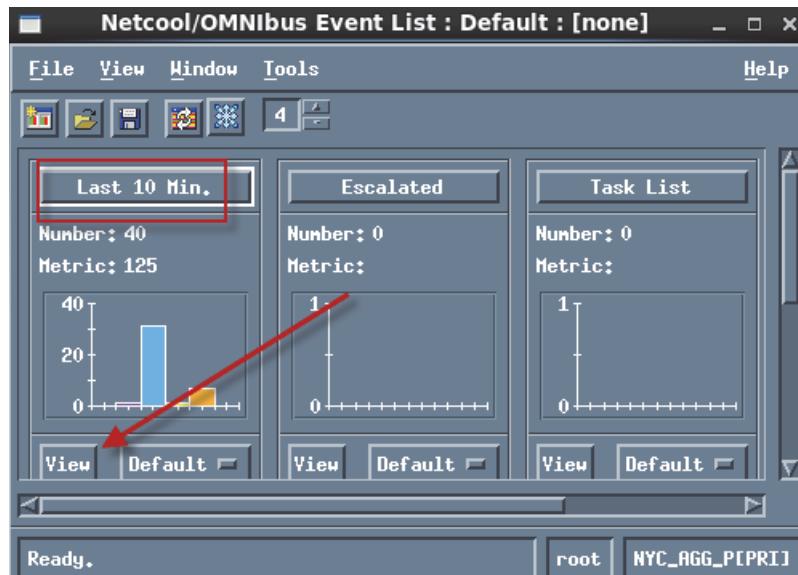
- b. Connect to the NYC_AGG_P ObjectServer as the *root* user, with no password.



- c. Click OK.

The **Event List** window opens.

- d. Locate the box that is labeled **Last 10 Mins.**, and click **View**:



The Sub-Event List view opens. The Simnet probe generates the events in this view. These steps verify that the ObjectServer is active, the Simnet probe is able to connect, and the ObjectServer generates events that are based on data that is provided by the probe.

- e. Click **File > Close** to close the Sub-Event List window.
f. Click **File > Exit** to close the Event List window.
g. Click **Yes** to abandon the changes.

Adding a password to the root ObjectServer user

When an ObjectServer is created, the **root** user is defined with no password. The following steps use a command-line utility to add a password to that user.

1. Add a password to the **NYC_AGG_P root** user as follows.

- a. Connect to the ObjectServer with the `nco_sql` utility:

```
nco_sql -server NYC_AGG_P -user root -password ''
```



Important: The value for password in the command that is shown is *two single quotation marks* (''). This syntax indicates a *blank* password.

- b. Enter the following commands that are shown in bold text:

```
1> alter user 'root' set password 'object00';
2> go
(0 rows affected)
1> quit
```

The password for the **root** user is now **object00** on the **NYC_AGG_P** ObjectServer.

- c. Verify that the password is correct:

```
nco_sql -server NYC_AGG_P -user root -password 'object00'

1> quit
```

The prompt characters (1>) indicate that the utility is able to connect to the ObjectServer with the revised password. Enter **quit** to exit the utility.

Configuring the Firefox plug-in

Several components within Web GUI require a special Java plug-in. The plug-in is bundles with Netcool/OMNIbus core libraries. You configure the plug-in to Firefox as the **root** user.

1. Configure Firefox plug-in as follows:

- a. Change to the **root** user:

```
su -
Password: object00
```

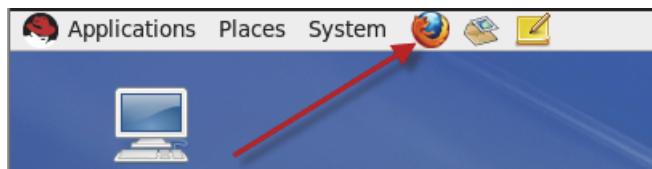
- b. Create the **plugins** directory:

```
cd /usr/lib64/firefox
mkdir plugins
```

- c. Define a symbolic link to the plug-in:

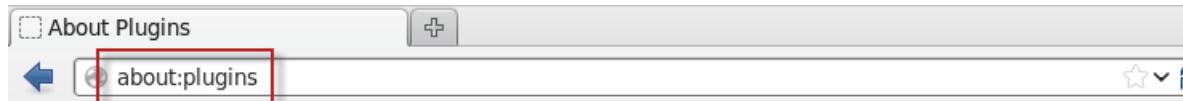
```
cd plugins
ln -s /opt/IBM/tivoli/netcool/platform/linux2x86/jre64_1.7.0/jre/lib/amd64/libnpjp2.so
```

2. Verify that the plug-in is available to Firefox.
 - a. Open a Firefox browser.



- b. Enter the following text in the URL field:

`about:plugins`



Enabled plugins

Find more information about browser plugins at mozilla.org.
Find updates for installed plugins at mozilla.com/plugincheck.
Help for installing plugins is available from plugindoc.mozdev.org.

IBM Developer Kit for Linux,Java,1.7.0

File: libnpjp2.so
Version: 10.45.2
Next Generation Java Plug-in 10.45.2 for Mozilla browsers

- c. Close the Firefox browser.
3. Exit the **root** user back to the **netcool** user.

`exit`

The following list is a summary of the accomplishments from this unit:

- Install Netcool/OMNIbus core component.
- Create and run the primary ObjectServer.
- The root user has a valid password in the ObjectServer.
- Verify basic ObjectServer function.



4 Installing Netcool/OMNIbus Web GUI exercises

In this unit, you install the Netcool/OMNIbus Web GUI component, and validate the installation.



Note: You are the **netcool** user for the steps in this unit.

Exercise 1 Installing Netcool/OMNIbus Web GUI

Netcool®/OMNIbus Web GUI is an application that runs inside Dashboard Application Services Hub. Dashboard Application Services Hub is also known as Visualization Services within Jazz™ for Service Management. You must install Jazz for Service Management before you install Web GUI.

Installing Jazz for Service Management

1. Log in to the **host2** server as user **netcool** with password **object00**.
2. Open a Terminal window.
3. Configure environment variables:

```
cd /workshop/netcool
```

```
cat .bashrc >> /home/netcool/.bashrc
```

```
source /home/netcool/.bashrc
```

4. Verify environment variables:

```
env | grep IBM
```

```
PATH=/opt/IBM/tivoli/netcool/bin:/opt/IBM/tivoli/netcool/omnibus/bin:/opt/IBM/tivoli/netcool/omnibus/probes:/usr/lib64/qt-3.3/bin:/usr/local/bin:/usr/bin:/bin:/usr/local/sbin:/usr/sbin:/sbin:/home/netcool/bin  
NCHOME=/opt/IBM/tivoli/netcool  
OMNIHOME=/opt/IBM/tivoli/netcool/omnibus
```



Hint: The environment variables are not required for the installation.

5. Create a directory to hold the installation files:

```
mkdir /tmp/jazz_install
```

6. Expand the Jazz installation file into the target directory:

```
cd /tmp/jazz_install  
unzip /software/jazz/jazz_for_sm_1.1.0.3_for_linux.zip
```

7. Expand the WebSphere® installation file into the target directory:

```
unzip /software/was/WAS_V8.5.0.1_FOR_JAZZSM_LINUX_ML.zip
```

8. Install the Jazz software:

```
./launchpad.sh
```

a. Select **Custom**.

Jazz for Service Management 1.1.0.3

- ▶ Welcome
- ▷ Full
- ▷ Custom
- ▷ Tools
- ▷ Exit

Welcome to Jazz for Service Management 1.1.0.3

Jazz for Service Management provides the following integration services: Administration Tivoli Common Reporting, Registry Services, and Security Services. Use the launchpad installation.

For links to the latest release and support information, expand **Release Information**.

Click one of the following links to guide you through installing Jazz for Service Management.

- [Full](#)

Use full installation for evaluation or development purposes. After you provide information about your environment, the installation wizard guides you through the steps to install Jazz for Service Management integration services, IBM DB2, and IBM WebSphere Application Server.

- [Custom](#)

b. Select **Next**.

- c. Leave the existing location field blank and click **Next**.

Specify Jazz for Service Management Home Location

Existing environment

Existing Jazz for Service Management home location:

- d. Observe the package repository locations and click **Next**.

Specify Source Locations

Jazz for Service Management 1.1 or later package repository:

▼ DB2, WebSphere Application Server, and Tivoli Common Reporting source locations

IBM DB2 10.1 or later, Enterprise Server Edition installation image:

IBM WebSphere Application Server 8.5 or later package repository:

IBM Tivoli Common Reporting 3.1 or later installation image:



Note: Ignore the warning messages about DB2® and Tivoli® Common Reporting. These messages are resolved in a subsequent exercise.

- e. Change the options to **not install** Administration Services, Registry Services, and Security Services.
- f. Leave the options to install Visualization Services (DASH) and IBM® WebSphere Application Server. Click **Next**.

Select Operations

For each component, select the operation to run. Select **None** if you do not want to run an operation for a specific component.

Component	Current	Target	Operation
Administration Services	-	-	<input type="button" value="None"/>
Service provider	1.1.0.3	1.1.0.3	<input type="button" value="None"/>
User Interface	1.1.0.3	1.1.0.3	<input type="button" value="None"/>
Registry Services	-	1.1.0.3	<input type="button" value="None"/>
Reporting Services	-	1.1.0.3	<input type="button" value="None"/>
Security Services	-	1.1.0.3	<input type="button" value="None"/>
Visualization Services	3.1.0.3	3.1.0.3	<input type="button" value="Install"/>
IBM DB2 Enterprise Server Edition	9.7.0.6	-	<input type="button" value="None"/>
IBM WebSphere Application Server	-	8.5.0.1	<input type="button" value="Install"/>

- g. Scroll down and accept the license agreement. Click **Next**.
- h. Accept the default locations and click **Next**.
- i. Enter **object00** for the administrator password and click **Next**.

Specify Credentials

WebSphere Application Server Profile

Administrator name:	smadmin
Administrator password:	*****
Confirm administrator password:	*****

- j. Verify the installation summary and click **Run**.

Run Tasks

The following tasks will be run in a sequence:

Task
Check task dependencies
Check system prerequisites
Install IBM WebSphere Application Server 8.5.0.1
Install Jazz for Service Management extension for IBM WebSphere 8.5 1.1.0.2
Install IBM Dashboard Application Services Hub 3.1.0.3

Click **Run** to continue.



Note: The installation process runs for approximately 50 minutes.

- k. Verify the completion status.

Review Results

Task	Status
Check task dependencies	Complete
Check system prerequisites	Warning
Install IBM WebSphere Application Server 8.5.0.1	Complete
Install Jazz for Service Management extension for IBM WebSphere 8.5 1.1.0.2	Complete
Install IBM Dashboard Application Services Hub 3.1.0.3	Complete

A callout bubble labeled "Prerequisite Scanner" points to the "Warning" status for the "Check system prerequisites" task.

The IBM Prerequisite Scanner is bundled with the Jazz for Service Management installation.
The scanner runs automatically during the installation.

- I. Click View Details and examine the results of the scan.

```
intel.cpu FAIL 2.20GHz 2.4GHz
```

The scan fails because the CPU speed is not considered fast enough. This failure is not considered unrecoverable, so the overall result is a warning.

- m. Click **Exit** to close the launchpad.

9. Remove the installation files to conserve disk space.

```
cd /tmp  
/bin/rm -R jazz_install
```



Hint: Use **/bin/rm** to circumvent the UNIX alias for the **rm** command that prompts for verification to remove files.

```
cd /software  
/bin/rm -R jazz  
/bin/rm -R was
```

Installing Web GUI

The ObjectServer must be running when Web GUI is installed. The Web GUI installation process runs a postinstallation utility that adds users and groups to the ObjectServer. It is possible to install Web GUI and then run the postinstallation utility manually later. In this case, the ObjectServer does not need to be running during the installation.



Important: The Web GUI component is installed with IBM Installation Manager. The installation of Jazz for Service Management installed a copy of IBM Installation Manager. However, that version is not compatible for installing Web GUI. IBM Installation Manager v1.7.2 is required for Web GUI, and is bundled with the Web GUI software. In the following steps, you install IBM Installation Manager v1.7.2 and Web GUI v8.1.

1. Verify that the NYC_AGG_P ObjectServer is running on **host1**.

2. Expand the installation file:

```
cd /software/webgui  
unzip OMNIbus-v8.1-WebGUI.linux64.zip
```

3. Install the Web GUI software:

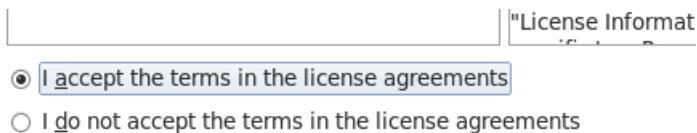
- a. Start installation launchpad

```
cd /software/webgui/im.linux.x86_64  
./userinst
```

- b. Review the packages and click **Next**.

Installation Packages	Status	Vendor
IBM® Installation Manager Version 1.7.2	Will be installed	IBM
IBM Tivoli Netcool/OMNibus Web GUI Version 8.1.0.0	Will be installed	IBM

- c. Accept the license agreement and click **Next**.



- d. Leave the option selected to create a new package group and click **Next**.

Package Group Name	Installation Directory
IBM Netcool	/opt/IBM/netcool

Package Group Name: IBM Netcool
Installation Directory: /opt/IBM/netcool
Architecture Selection: 32-bit 64-bit

parent directory can change if necessary



Hint: The installation directory can be changed if you want.

- e. Accept all of the defaults and click **Next**.



Note: If you use SmartCloud Analytics - Log Analysis, you select the box to install the integration. You can also configure the integration manually after Web GUI is installed.

- f. Enter **object00** as the password and click **Next**.

Installation Directory Details

WebSphere Application Server	/opt/IBM/WebSphere/AppServer
Jazz for Service Management UI	/opt/IBM/JazzSM/ui

Profile Details

Server name	server1
User name	smadmin
Password	*****

A callout bubble points to the 'server1' field with the text: 'This is the WebSphere internal server name. It is not the UNIX host name. Do not change this'.



Important: The value **server1** is the name of the WebSphere® internal server. It is not the UNIX host name.

The installer verifies that the user name and password provide access to Dashboard Application Services Hub.

- g. Review the installation summary and click **Install**.

Install Packages
Review the summary information.

Install Licenses Location Features Summary

Target Location

Package Group Name: IBM Netcool
Installation Directory: /opt/IBM/netcool
Shared Resources Directory: /opt/IBM/IMShared

Packages

Packages

- IBM® Installation Manager 1.7.2
- IBM Tivoli Netcool/OMNibus Web GUI 8.1.0.0
 - Install base features

Note: The installation process runs approximately 15 minutes.

- h. Verify that the installation is successful and click **View Log File**.



The list of available log files opens. The most recent file is selected.

- i. Click the last icon on the right to open the selected log file.

Current session installation log

Severity	Description
	Log File: /home/netcool/var/ibm/InstallationManager/logs/2014

A red arrow points to the three icons at the top right of the log file window: a magnifying glass, a refresh symbol, and a double arrow.

- j. Scroll to the bottom of the log file and examine the elapsed installation time.

48:44.44	Elapsed time 09:47.39 for: Performing "post-install configure" phase
48:44.44	Performing "complete" phase
48:44.44	Elapsed time 00:00 00 for: Performing "complete" phase
48:45.53	Elapsed time 10:17.96 for: Installing com.ibm.tivoli.netcool.omnibus.webgu selectors: ws=gtk, os=linux, arch=x86_64, nl=en

The green color on each line indicates that the step is successful. If there are any issues during the installation that are considered failures, then Installation Manager automatically rolls back any changes that are made during the current installation.

- k. Close the log file view.
l. Close the window with the list of log files.

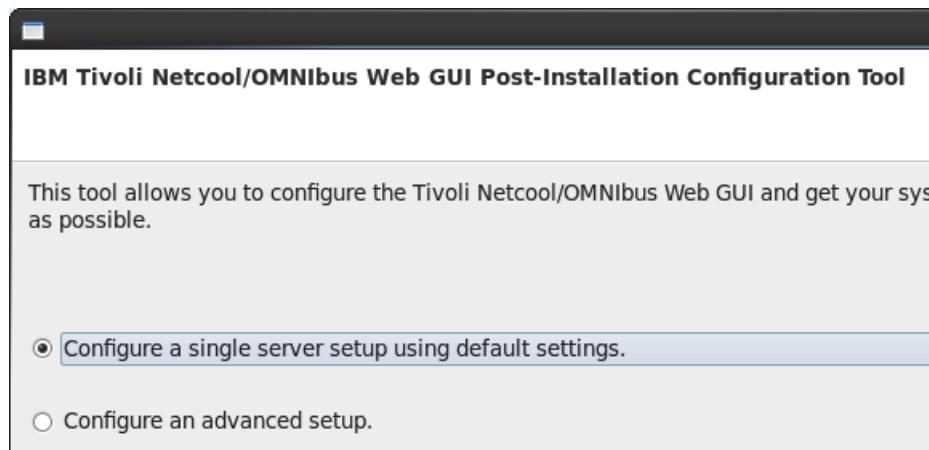
- m. Leave the option set to configure Web GUI and click **Finish**.



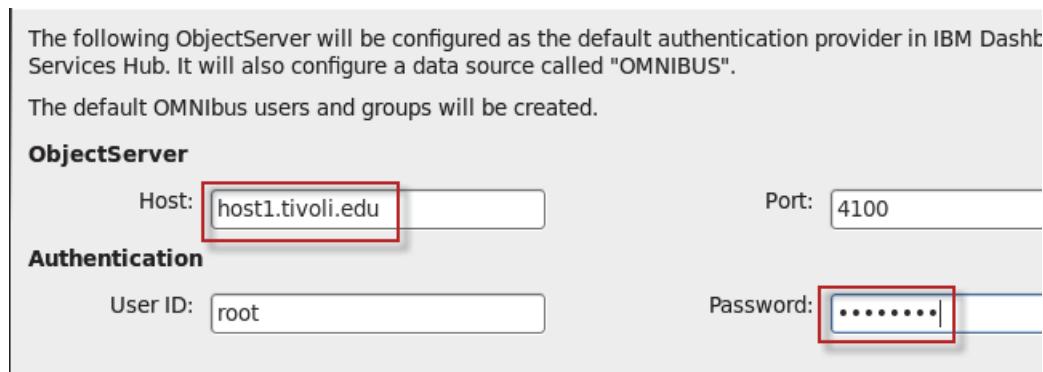
Web GUI postinstallation configuration

The installation process starts the Web GUI Post-Installation Configuration Tool.

1. Leave the default and click **Next**.



2. Change the Host to **host1.tivoli.edu**, enter **object00** for the password, and click **Next**.



3. Review the summary and click **Next**.

IBM Tivoli Netcool/OMNIBus Web GUI Post-Installation Configuration Tool

You are about to configure the following as the default authentication provider in IBM Dashboard Application Services Hub:

- **ObjectServer Repository**
- host1.tivoli.edu : 4100 (Primary Server)
- root (User)

You have opted to create the following users and groups:

Default Users (2)

- ncoadmin
- ncouser

Default Groups (2)

- Netcool_OMNIBus_Admin
- Netcool_OMNIBus_User

The system will also configure a data source called **OMNIBUS**

Click Next to continue.

4. Verify that the steps are complete and click **Next**.

IBM Tivoli Netcool/OMNIBus Web GUI Post-Installation Configuration Tool

Configuring ObjectServer user repository in IBM Dashboard Application Services Hub...



Configuring data source for Tivoli Netcool/OMNIBus Web GUI...



Restarting IBM Websphere Application server...



Creating default users and groups...



5. Review the configuration results and click **Finish**.

IBM Tivoli Netcool/OMNibus Web GUI Post-Installation Configuration Tool

You have successfully configured IBM Tivoli Netcool/OMNibus Web GUI.

- **ObjectServer Repository**
 - host1.tivoli.edu : 4100 (Primary Server)
 - root (User)

Following users and groups have been created:

Default Users (2)

- ncoadmin
- ncouser

Default Groups (2)

- Netcool_OMNIbus_Admin
- Netcool_OMNIbus_User

The system has also configured a data source called **OMNIBUS**

Click Finish to exit.

6. Remove the installation files to conserve disk space.

```
cd /software
/bin/rm -R webgui
```

The Configuration Tool can be run manually as follows:

```
cd /opt/IBM/netcool/omnibus_webgui/configtool/linux.gtk.x86_64
./ncwConfigUI -WASUserID smadmin -WASPassword object00
```

7. Restart Dashboard Application Services Hub.

- a. Stop the server.

```
cd /opt/IBM/JazzSM/profile/bin
./stopServer.sh server1 -username smadmin -password object00
ADMU0116I: Tool information is being logged in file
        /opt/IBM/JazzSM/profile/logs/server1/stopServer.log
ADMU0128I: Starting tool with the JazzSMPProfile profile
ADMU3100I: Reading configuration for server: server1
ADMU3201I: Server stop request issued. Waiting for stop status.
ADMU4000I: Server server1 stop completed.
```

 **Note:** You must provide the administrator user names and password to stop the server.

- b. Start the server.

```
./startServer.sh server1
ADMU0116I: Tool information is being logged in file
/opt/IBM/JazzSM/profile/logs/server1/startServer.log
ADMU0128I: Starting tool with the JazzSMProfile profile
ADMU3100I: Reading configuration for server: server1
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process id is 9137
```



Note: The user name and password are not required to start the server.

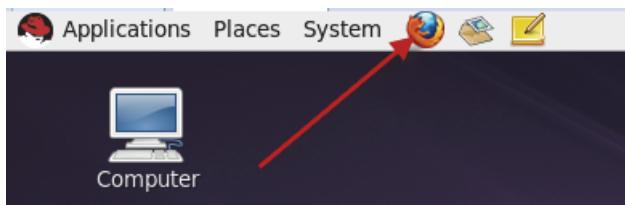
Exercise 2 Verifying access to Netcool/OMNIbus Web GUI

The installation process starts the Dashboard Application Services Hub and Web GUI components.



Important: You perform the Web GUI verification steps on the host1 image. You use this image because you previously configured the Firefox browser on the host1 image with the correct Java plug-in. The plug-in is included in the Netcool/OMNIbus core libraries, but not in the Web GUI libraries. In a production environment, you can retrieve a copy of the plug-in file and copy it to some location in the Web GUI libraries. In a subsequent unit, you install Netcool/OMNIbus core libraries on the host2 image. Then you configure the Firefox browser on host2.

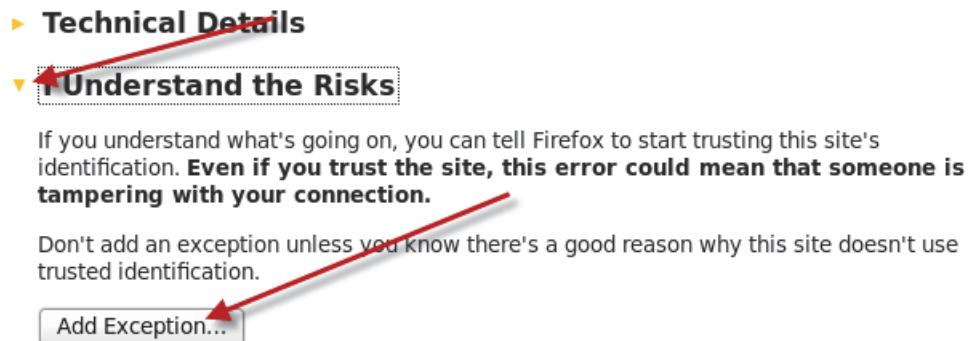
1. Switch to the **host1** image.
2. Verify access to the Web GUI.
 - a. Open a Firefox browser.



- b. Enter the following URL:

<http://host2.tivoli.edu:16310/ibm/console>

- c. Accept the security certificate:



- i. Expand **I Understand the Risks**.

- ii. Click **Add Exception**.



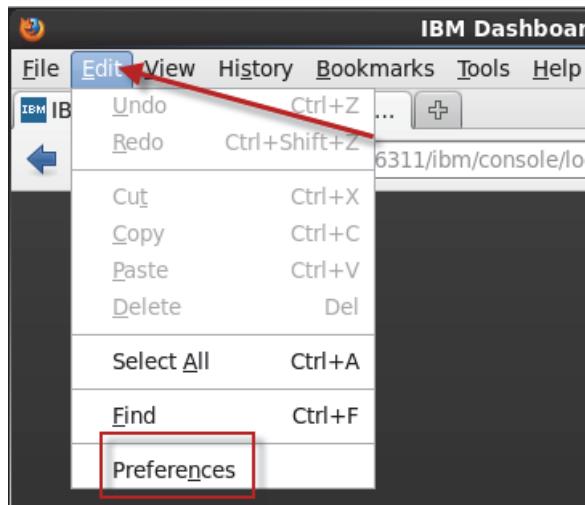
- iii. Click **Confirm Security Exception**.

The URL automatically redirects to:

<https://host2:16311/ibm/console/logon.jsp>

- iv. Set **Dashboard Application Services Hub** as the browser home page.

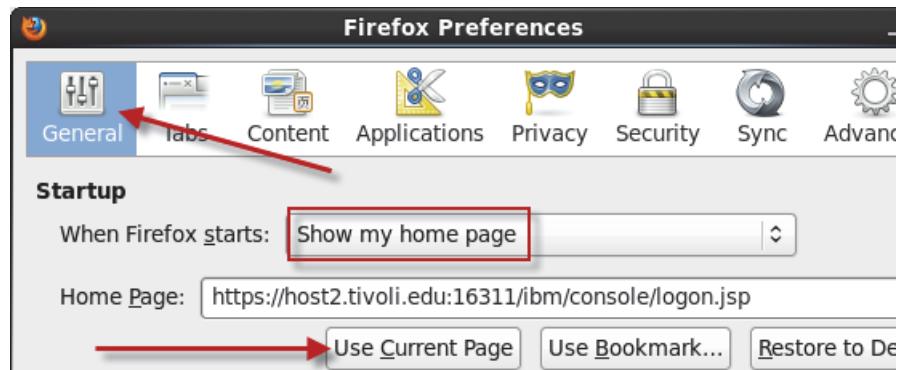
- d. Select **Edit > Preferences**.



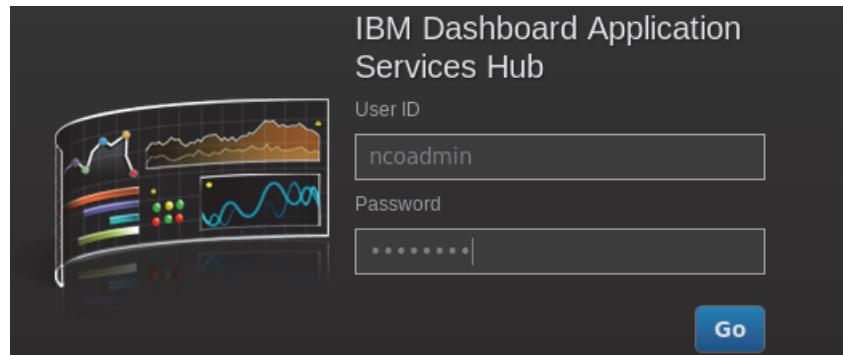
- e. Click the **General** tab.

- f. Click **Use Current Page** to set the home page.

- g. Click **Close**.

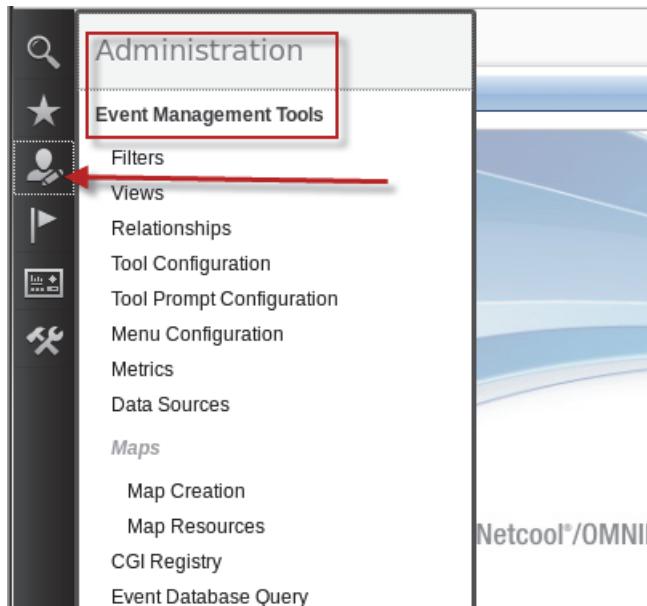


- h. Log in as user **ncoadmin** with password **object00**.



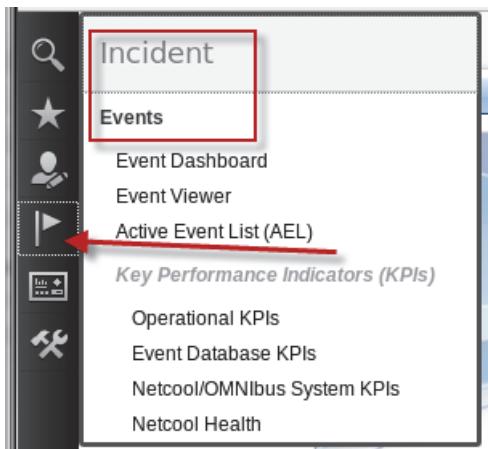
The postinstallation configuration tool created the **ncoadmin** user. The **ncoadmin** user is configured with Netcool administrative authority in addition to normal user capabilities.

- i. Click the indicated icon and examine the Event Management Tools that are available to the **ncoadmin** user.



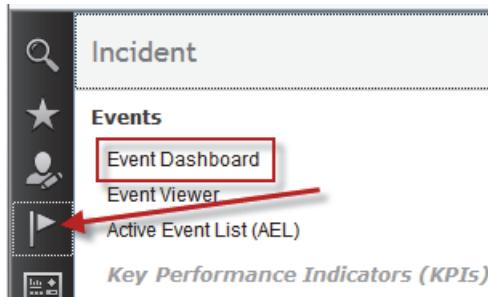
Important: Access to Event Management Tools requires the **ncw_admin** role.

- j. Click the indicated icon and examine the Events features that are available to the **ncoadmin** user.



Important: Access to Events requires the *ncw_user* role.

- k. Click **Event Dashboard**.

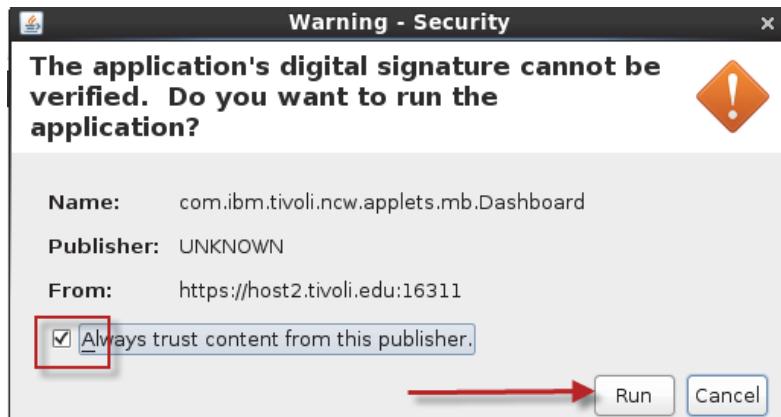


Note: The Event Dashboard requires the Java plug-in.

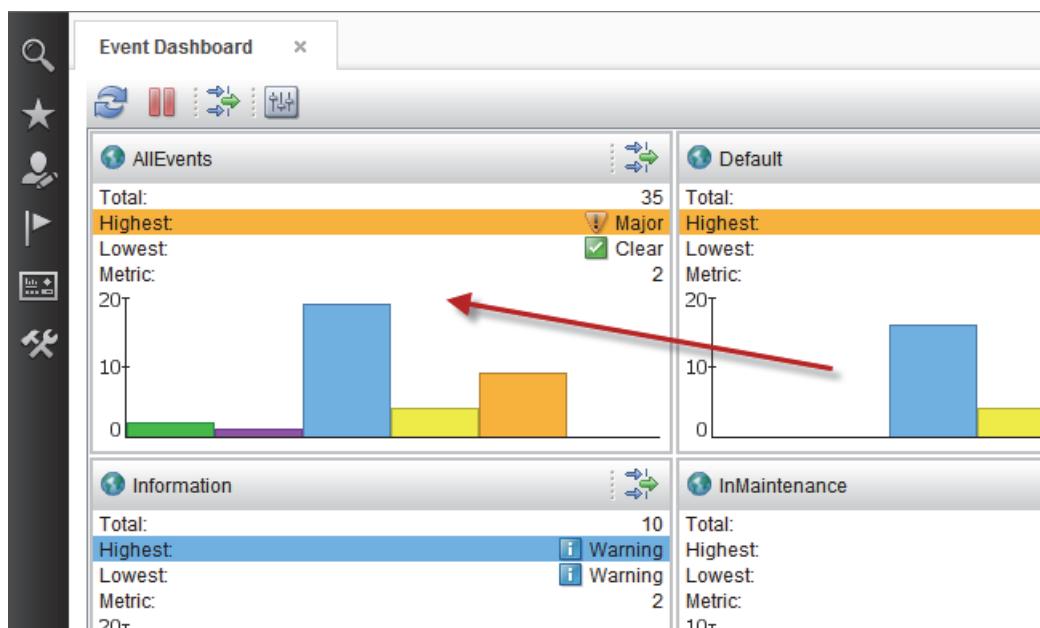
- l. Select the option to always trust the content and click **Yes**.



- m. Select the option to always trust the content and click **Run**.



- n. Click anywhere in the box that is labeled **AllEvents**.

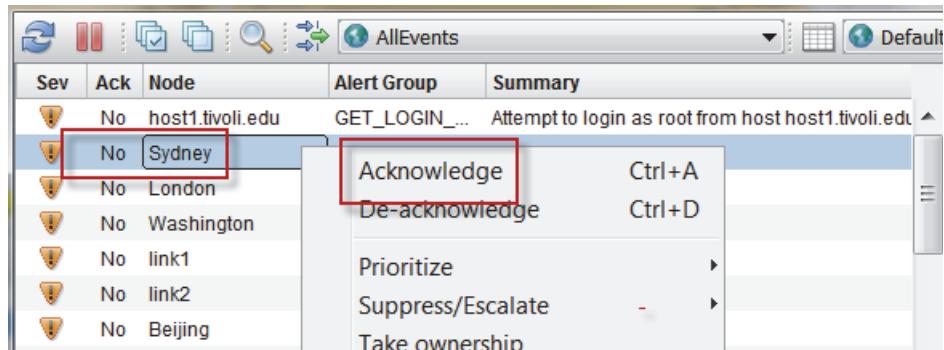


The Active Event List opens in a new window.



Note: The Active Event List requires the Java plug-in.

- o. Click any event to select it, right-click, and select **Acknowledge**.

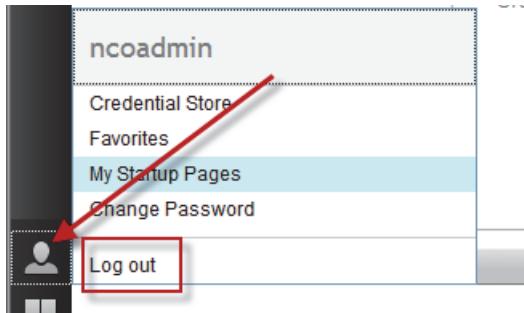


The tool runs and the event is updated.

Sev	Ack	Node	Alert Group	Summary
!	No	host1.tivoli.edu	GET_LOGIN...	Attempt to login as root from t
!	Yes	Sydney	Systems	Machine has gone offline
!	No	London	Systems	Machine has gone offline
!	No	Washington	Systems	Machine has gone offline
!	No	link1	Link	Link Down on port

This action verifies access to Dashboard Application Services Hub and the Web GUI component. It also verifies that the Java plug-in functions correctly, and that the **ncoadmin** user has the correct authority to update event records in the ObjectServer.

- Close the **Active Event List** window.
- Log out as the **ncoadmin** user.



- Verify access with the **smadmin** user.
 - Log in as the **smadmin** user with password **object00**.
 - Click the icon to open **Console Settings** and select **WebSphere Administrative Console**.

Console Settings

General

- Catalogs
- Connections
- Console Preference Profiles
- Pages
- Widgets
- Views
- WebSphere Administrative Console**
- Console Integrations
- Export Wizard

Roles

- Group Roles
- Roles
- User Roles**

Version 8

Licensed M
licensors: I
registered
or both. Ja
and/or its :

- c. Click **Launch WebSphere administrative console**.

WebSphere Administrative Console

The WebSphere administrative console provides access to security and management functions for your WebSphere application server.
Logging out from the WebSphere administrative console will also log you out of the application server.

[Launch WebSphere administrative console](#)

Accept any security requirements.

- d. Expand **Users and Groups** and click **Manage Groups**.



- e. Observe the entries that are highlighted in the screen capture.

Select	Group name	Description	Unique Name
<input type="checkbox"/>	Administrator	Admin Group	cn=Administrator,o=netcoolObjectServerRepository
<input type="checkbox"/>	Gateway	Permissions required for a gateway user	cn=Gateway,o=netcoolObjectServerRepository
<input type="checkbox"/>	ISQL	Read only ISQL access	cn=ISQL,o=netcoolObjectServerRepository
<input type="checkbox"/>	ISQLWrite	Write ISQL access	cn=ISQLWrite,o=netcoolObjectServerRepository
<input type="checkbox"/>	Netcool_OMNIbus Admin		cn=Netcool_OMNIbus_Admin,o=netcoolObjectServerRepository
<input type="checkbox"/>	Netcool_OMNIbus User		cn=Netcool_OMNIbus_User,o=netcoolObjectServerRepository
<input type="checkbox"/>	Normal	Normal Group	cn=Normal,o=netcoolObjectServerRepository
<input type="checkbox"/>	Probe	Permissions required for a probe user	cn=Probe,o=netcoolObjectServerRepository
<input type="checkbox"/>	Public	Public Group	cn=Public,o=netcoolObjectServerRepository
<input type="checkbox"/>	System	System Group	cn=System,o=netcoolObjectServerRepository

Page 1 of 1

Total: 10

The postinstallation configuration tool created the two groups `Netcool_OMNIbus_Admin` and `Netcool_OMNIbus_User`.

- f. Observe the values for Unique Name.

The values under the column Unique Name contain the text:

`o=netcoolObjectServerRepository`

This text indicates that the group names are defined in the ObjectServer. All of the groups that are shown here are defined in the `NYC_AGG_P` ObjectServer.

- g. Click **Manage Users**.



- h. Observe the entry for **smadmin**.

Select	User ID	First name	Last name	E-mail	Unique Name
<input type="checkbox"/>	ncoadmin	ncoadmin	tivoli		uid=ncoadmin,o=netcoolObjectServerRepository
<input type="checkbox"/>	ncouser	ncouser	tivoli		uid=ncouser,o=netcoolObjectServerRepository
<input type="checkbox"/>	nobody		Nobody		uid=nobody,o=netcoolObjectServerRepository
<input type="checkbox"/>	root	Root	User		uid=root,o=netcoolObjectServerRepository
<input type="checkbox"/>	smadmin	smadmin	smadmin		uid=smadmin,o=defaultWIMFileBasedRealm

The **smadmin** user is created in Dashboard Application Services Hub by the installer.

Notice the value under the column **Unique Name**:

`o=defaultWIMFileBasedRealm`

This setting indicates that the user definition and password for this user are stored in a file.

This file is internal to Dashboard Application Services Hub. It does not use the ObjectServer.

The other users and their passwords are defined in the ObjectServer. The postinstallation configuration tool created the two users: **ncoadmin** and **ncouser**.

- i. Log out of administrative console.
 - j. Log out of Dashboard Application Services Hub.
 - k. Close the browser.
4. Switch to the **host2** image.

Exercise 3 Configuring Web GUI to automatically start

The following steps demonstrate how to create a script that is used to start Dashboard Application Services Hub when the server is restarted.



Note: The start script is provided with the class image. It is not included with the Web GUI software.

1. Configure Dashboard Application Services Hub to start automatically when the system starts.

- a. Change to the **root** user.

```
su -
```

```
Password: object00
```

- b. Copy the supplied start script to the correct location.

```
cd /workshop/etc/init.d
```

```
cp jazz /etc/init.d
```

- c. Change the file permissions to allow execute.

```
cd /etc/init.d
```

```
chmod +x jazz
```

- d. Create the logical links to enable auto-start.

```
chkconfig jazz on
```

2. Verify auto-start.

- a. Restart the server.

```
init 6
```

Wait for the server to initialize.

- b. Log in as **netcool** with password **object00**.

- c. Open a Firefox browser and connect to Dashboard Application Services Hub.

```
http://host2.tivoli.edu:16310/ibm/console
```



The login screen verifies that Dashboard Application Services Hub started automatically.

3. Configure Firefox to use this URL as the home page.

4. Close the browser.

You can log in as any valid user. However, the Firefox plug-in is not configured correctly yet. The configuration of the plug-in on host2 happens in a subsequent unit.

Exercise 4 Configuring LDAP as an authentication source

The following steps demonstrate how to modify the existing configuration so that Dashboard Application Services Hub uses LDAP as an authentication source. Some of the following steps are performed on the host1 image and some steps are performed on the host2 image. Make sure that you perform the steps on the correct image.



Important: In a production environment, you might decide not to use an LDAP repository. If that is the case, you do not perform any of the following steps.

Removing the ObjectServer user repository

Perform the following steps on the **host2** image.

The configuration for the Virtual Member Manager component is defined in an XML file. Save a copy of this file before you modify the existing configuration.

1. Save a copy of the VMM configuration file:

```
cd /opt/IBM/JazzSM/profile/config/cells/JazzSMNode01Cell/wim/config  
cp wimconfig.xml /home/netcool
```



Important: If any of the following configuration steps fail, you can recover the original configuration by copying the saved file back to the original location, and restarting Dashboard Application Services Hub.

2. Connect to WebSphere Administrative Console as follows:

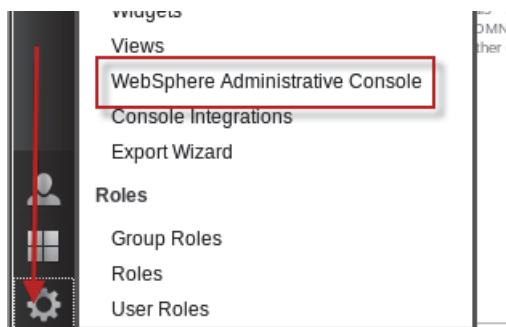
- a. Open a Firefox browser and connect to Dashboard Application Services Hub.

<http://host2.tivoli.edu:16310.ibm/console>

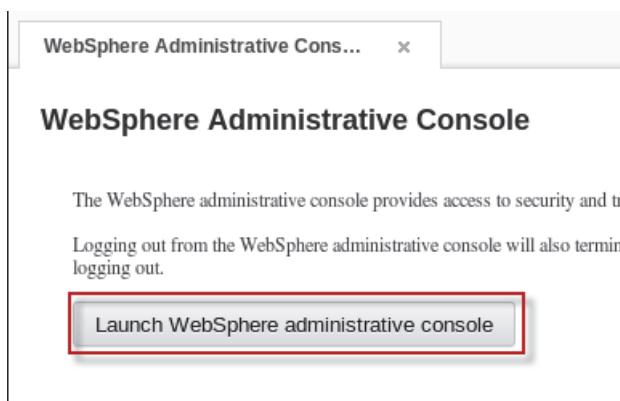


Hint: If you did not change the Firefox home page previously, change it now.

- b. Log in as the **smadmin** user with password **object00**.
- c. Click the icon and select **WebSphere Administrative Console**.

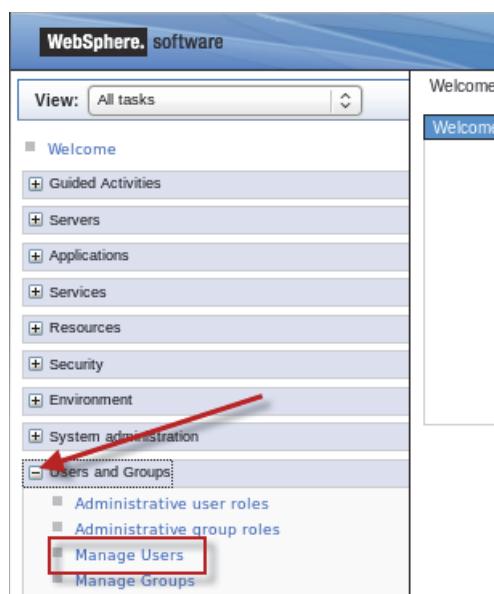


- d. Click **Launch WebSphere administrative console**.



- e. Accept all security messages. The administrative console opens in a new Firefox tab.
3. Remove the default users and groups.
Two users and two groups were created when Web GUI was installed. Remove those entries before changing the user repositories. You add them again in a subsequent step.

- a. Expand **Users and Groups** and click **Manage Users**.



- b. Click the boxes to select the two users and click **Delete**.

5 users matched the search criteria.					
Select	User ID	First name	Last name	E-mail	Unique Name
<input checked="" type="checkbox"/>	ncoadmin	ncoadmin	tivoli		uid=ncoadmin,o=netcoolObjectSe
<input checked="" type="checkbox"/>	ncouser	ncouser	tivoli		uid=ncouser,o=netcoolObjectSe
<input type="checkbox"/>	nobody		Nobody		uid=nobody,o=netcoolObjectSe
<input type="checkbox"/>	root	Root	User		uid=root,o=netcoolObjectServerRe
<input type="checkbox"/>	smadmin	smadmin	smadmin		uid=smadmin,o=defaultWIMFileBa

- c. Click **Delete**.



Important: Do not remove any of the other users.



The **ncoadmin** and **ncouser** IDs are deleted from the ObjectServer.

- d. Click **Manage Groups**.

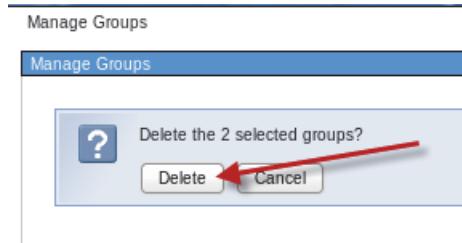
- e. Click the boxes to select the two groups and click **Delete**.

11 groups matched the search criteria.		
Select	Group name	Description
<input type="checkbox"/>	Administrator	Admin Group
<input type="checkbox"/>	Gateway	Permissions required for a gateway
<input type="checkbox"/>	ISQL	Read only ISQL access
<input type="checkbox"/>	ISQLWrite	Write ISQL access
<input checked="" type="checkbox"/>	Netcool_OMNIbus_Admin	
<input checked="" type="checkbox"/>	Netcool_OMNIbus_User	

- f. Click **Delete**.



Important: Do not remove any of the other groups.



The Netcool_OMNIbus_Admin and Netcool_OMNIbus_User groups are deleted.

4. Removing the ObjectServer definition.

- a. Expand **Security** and click **Global Security**.



- b. Scroll down on the page to the User account repository section and click **Configure**.

User account repository

Realm name: defaultWIMFileBasedRealm

Current realm definition: Federated repositories

Available realm definitions: Federated repositories

Configure... Set as current

- c. Scroll down on the page to the *Repositories in the realm*, select the check box for the ObjectServer entry, and click **Remove**.

Repositories in the realm:			
Add repositories (LDAP, custom, etc)...		Use built-in repository	Remove
Select	Base Entry	Repository Identifier	Repository Type
You can administer the following resources:			
<input type="checkbox"/>	o=defaultWIMFileBasedRealm	InternalFileRepository	File
<input checked="" type="checkbox"/>	o=netcoolObjectServerRepository	NetcoolObjectServer	Custom
Total 2			

- d. Click **Save**.

Global security

Messages

Changes have been made to your local configuration. You can:

- Save directly to the master configuration.
- Review changes before saving or discarding.

The server may need to be restarted for these changes to take effect.

- e. Scroll down on the page to the *Related Items* section and click **Manage repositories**.

Repositories in the realm:

Add repositories (LDAP, custom, etc)...				Use built-in repository	Remove
Select	Base Entry	Repository Identifier	Repository Type		
You can administer the following resources:					
<input type="checkbox"/>	o=defaultWIMFileBasedRealm	InternalFileRepository	File		
Total 1					

Related Items

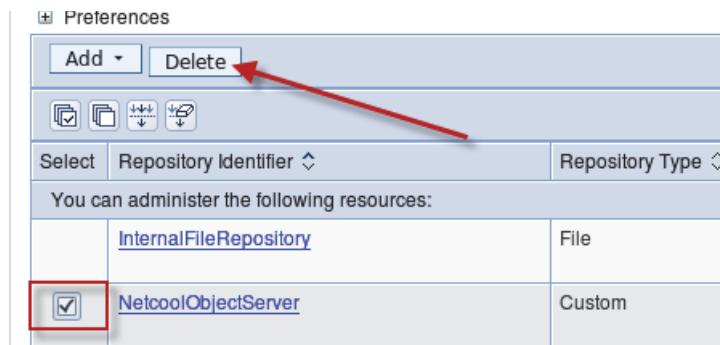
Additional Properties

- Property extension repository
- Entry mapping repository

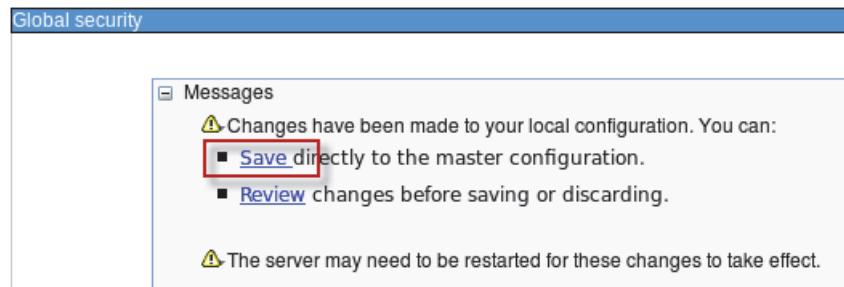
Manage repositories

Trusted authentication realms - inbound

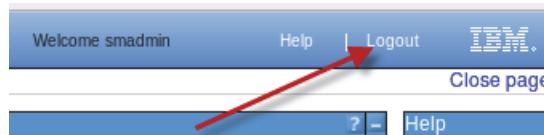
- f. Check the box to select the ObjectServer entry and click **Delete**.



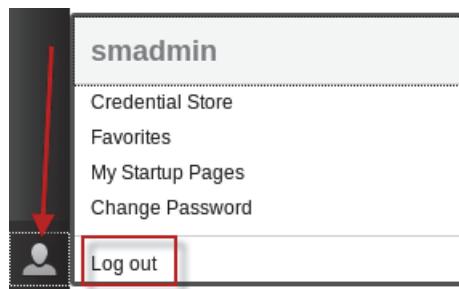
- g. Click **Save**.



- h. Log out of administrative console.



- i. Log out of Dashboard Application Services Hub.



The ObjectServer is removed as a Virtual Member Manager user repository. You must restart Dashboard Applications Services Hub to complete the removal.

5. Restart Dashboard Application Services Hub.

- a. Change to the **root** user.

```
su -
Password: object00
```

- b. Stop Dashboard Application Services Hub.

```
/etc/init.d/jazz stop
```

```
ADMU0116I: Tool information is being logged in file
/opt/IBM/JazzSM/profile/logs/server1/stopServer.log
ADMU0128I: Starting tool with the JazzSMPProfile profile
ADMU3100I: Reading configuration for server: server1
ADMU3201I: Server stop request issued. Waiting for stop status.
ADMU4000I: Server server1 stop completed.
```

- c. Start Dashboard Application Services Hub.

```
/etc/init.d/jazz start
```

```
ADMU0116I: Tool information is being logged in file
/opt/IBM/JazzSM/profile/logs/server1/startServer.log
ADMU0128I: Starting tool with the JazzSMPProfile profile
ADMU3100I: Reading configuration for server: server1
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process id is 6653
```

- d. Exit the **root** user and return to the **netcool** user.

```
exit
```

Dashboard Application Services Hub is now configured with a single user repository: internal file-based. The only valid user ID is **smadmin**, because that user is defined in the file-based repository.

6. Save another copy of the VMM configuration file.

```
cd /opt/IBM/JazzSM/profile/config/cells/JazzSMNode01Cell/wim/config
cp wimconfig.xml /home/netcool/wimconfig.xml.fileonly
```

You can compare the two configuration files to see how the ObjectServer is configured as a repository.

```
cd /home/netcool
```

```
diff wimconfig.xml wimconfig.xml.fileonly
```

```
<-- (C) Copyright IBM Corp. 2005 All Rights Reserved.
-->
> (C) Copyright IBM Corp. 2005_2010 All Rights Reserved.
38,45d37
<   <config:repositories adapterClassName="com.ibm.tivoli.tip.vmm4ncos.ObjectServerAdapter"
<     id="NetcoolObjectServer" supportPaging="false">
<       <config:baseEntries name="o=netcoolObjectServerRepository"/>
<       <config:CustomProperties name="password" value="(AES)430F76A91818857F79165B70890D458D"/>
<       <config:CustomProperties name="username" value="root"/>
<       <config:CustomProperties name="host1" value="host1.tivoli.edu"/>
<       <config:CustomProperties name="port1" value="4100"/>
<     </config:repositories>
49d40
<       <config:participatingBaseEntries name="o=netcoolObjectServerRepository"/>
```

The diff utility shows the differences between the contents of the two files. The text represents the configuration entries that are removed from the second file. Those entries were removed

when the ObjectServer repository was removed. You can see the access information for the ObjectServer in the removed lines.

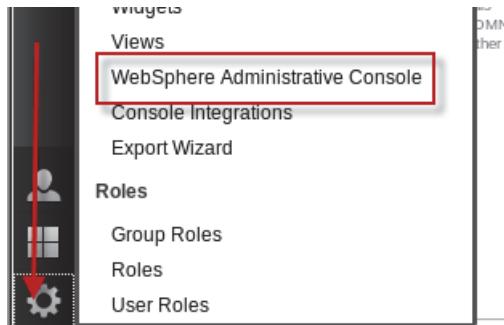
Adding the LDAP user repository

1. Connect to WebSphere Administrative Console as follows:
 - a. Open a Firefox browser and connect to Dashboard Application Services Hub.
<http://host2.tivoli.edu:16310/ibm/console>

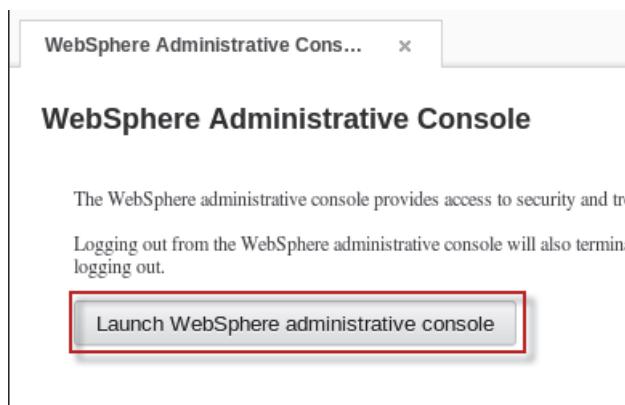


Hint: If you did not change the Firefox home page previously, change it now.

- b. Log in as the **smadmin** user with password **object00**.
 - c. Click the icon and select **WebSphere Administrative Console**.



- d. Click **Launch WebSphere Administrative Console**.



- e. Accept all security messages. The administrative console opens in a new Firefox tab.
 2. Adding the LDAP directory as a user repository.



Important: The LDAP directory is on the **host1** image.

- Expand **Security** and click **Global Security**.



- Scroll down on the page to the User account repository section and click **Configure**.

The screenshot shows the 'User account repository' configuration page. It includes fields for 'Realm name' (defaultWIMFileBasedRealm), 'Current realm definition' (Federated repositories), and 'Available realm definitions' (Federated repositories dropdown, Configure... button, Set as current button). A red arrow points to the 'Configure...' button.

- Scroll down on the page to the *Repositories in the realm*, and click **Add repositories**.

The screenshot shows the 'Repositories in the realm' section. It has a header with 'Add repositories (LDAP, custom, etc)...' (highlighted with a red box), 'Use built-in repository', and 'Remove'. Below is a table with columns: Select, Base Entry, Repository Identifier, and Repository Type. A row shows a checkbox, o=defaultWIMFileBasedRealm, InternalFileRepository, and File. At the bottom, it says 'Total 1'.

- Click **New Repository** and select **LDAP repository**.

The screenshot shows the 'General Properties' dialog for a new repository. It has fields for 'Repository' (none defined dropdown) and 'Unique distinguished name' (dropdown with options: LDAP repository, Custom repository, File repository). A red arrow points to the 'New Repository...' button, which is highlighted with a red box. Another red box highlights the 'LDAP repository' option in the dropdown menu.

- Change the repository name to **TIVIDS**.



Hint: The repository name is only relevant when you configure two copies of WebSphere to share a common LDAP for single sign-on. In that case, the repository name must be defined with the same text in each copy of WebSphere.

- f. Set the primary host name to **host1.tivoli.edu**.
- g. Verify that the port is set to **389**.
- h. Set the **Bind distinguished name** field to **cn=root**.
- i. Set the **Bind password** field to **object00**.
- j. Set the **Federated repository properties for login** field to **uid;cn**.
- k. Scroll to the bottom of the page and click **OK**.

General Properties

* Repository identifier
TIVIDS

Repository adapter class name
com.ibm.ws.wim.adapter.ldap.LdapAdapter

LDAP server

* Directory type
IBM Tivoli Directory Server

* Primary host name
host1.tivoli.edu Port
389

Failover server used when primary is not available:

Security

Bind distinguished name
cn=root

Bind password

Federated repository properties for login
uid;cn

- I. Enter **dc=ibm,dc=com** for the **Unique distinguished name** field, and click **OK**.

General Properties

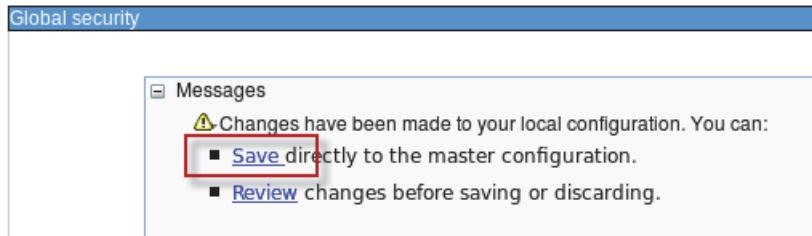
* Repository
TIVIDS New Repository...

* Unique distinguished name of the base (or parent) entry in federated
dc=ibm,dc=com

Distinguished name in the repository is different
Distinguished name of a subtree in the main repository

Apply OK Reset Cancel

m. Click **Save**.



Important: The base entry is mapped to the root of the LDAP directory. All operations are performed at root, which causes errors on most LDAP servers. More configuration is required.

The next step is to configure how the LDAP object names are mapped to each corresponding Virtual Member Manager resource type. You configure each resource type to specify search criteria. The search criteria is used to locate values for each of the object classes. These definitions essentially define the LDAP subtree where the Netcool user information is located.

3. Defining LDAP object class mappings.

a. Scroll down on the page and click **TIVIDS**.

Repositories in the realm:

Add repositories (LDAP, custom, etc)...		Use built-in repository	Remove
Select	Base Entry	Repository Identifier	Repository Type
You can administer the following resources:			
<input type="checkbox"/>	dc=ibm,dc=com	TIVIDS	LDAP:IDS
<input type="checkbox"/>	o=defaultVMFileBasedRealm	InternalFileRepository	File

b. Scroll down and click **Federated repositories entity types to LDAP object classes mapping**.

- [Additional Properties](#)
- [Performance](#)
- [Federated repositories entity types to LDAP object classes mapping](#)
- [Federated repositories property names to LDAP attributes mapping](#)
- [Group attribute definition](#)

Important: The following steps are unique to the configuration of the LDAP server. The steps that are shown here are relevant to the LDAP configuration used for the class. The process is the same regardless of the LDAP configuration. It is the values that are used in these steps that must change for some other LDAP server.

c. Click **Group**.

Select	Entity Type ▾	Object Classes ▾
You can administer the following resources:		
<input type="checkbox"/>	Group	groupOfNames
<input type="checkbox"/>	OrgContainer	organization;organizationalUnit;domain;container
<input type="checkbox"/>	PersonAccount	inetOrgPerson

d. Enter **ou=tipgroups,cn=tipRealm,DC=IBM,DC=COM** for Search bases and click **OK**.

* Entity type

* Object classes

Search bases

Search filter

e. Click **OrgContainer**.

Select	Entity Type ▾	Object Classes ▾
You can administer the following resources:		
<input type="checkbox"/>	Group	groupOfNames
<input type="checkbox"/>	OrgContainer	organization;organizationalUnit;domain;container
<input type="checkbox"/>	PersonAccount	inetOrgPerson

f. Verify that Search bases is empty and click **OK**.

* Entity type

* Object classes

Search bases

Search filter

g. Click PersonAccount.

Select	Entity Type ▾	Object Classes ▾
You can administer the following resources:		
<input type="checkbox"/>	Group	groupOfNames
<input type="checkbox"/>	OrgContainer	organization;organizationalUnit;domain;container
<input type="checkbox"/>	PersonAccount	inetOrgPerson

h. Enter **ou=tipusers,cn=tipRealm,DC=IBM,DC=COM** for the **Search bases** field and click **OK**.

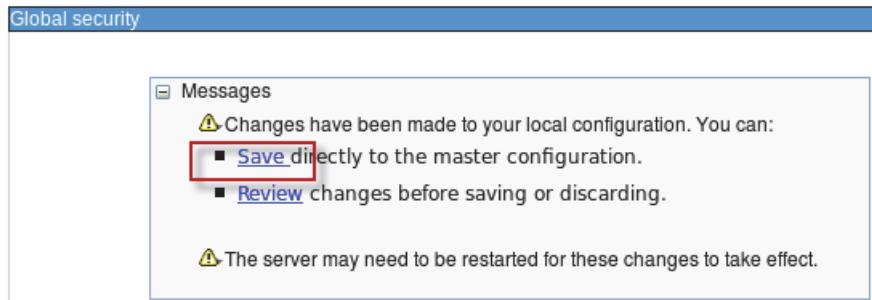
* Entity type

* Object classes

Search bases

Search filter

i. Click Save.



Now the Virtual Member Manager is configured to retrieve user information from a specific subtree within LDAP.

The last step is to configure Dashboard Application Services Hub to write new users and groups to the LDAP directory. This configuration process is similar to the previous steps. You define which LDAP object classes are modified when a new user or group is created.

4. Configure IBM Dashboard Application Services Hub to write to LDAP as follows:
 - a. Click **Federated repositories**.

Global security

Global security > Federated repositories > TIVIDS > Federated repositories entity types to LDAP object classes mapping

Use this page to list federated repositories entity types that are supported by the LDAP repository, to select an entity type to view or change its configuration properties, or to add or remove the entity type.

Preferences

New... **Delete**

- b. Scroll to the bottom of the page and click **Supported entity types**.

Repositories in the realm:

Add repositories (LDAP, custom, etc)...		Use built-in repository	Remove
Select	Base Entry	Repository Identifier	Repository Type
<input type="checkbox"/>	dc=ibm,dc=com	TIVIDS	LDAP:IDS
<input type="checkbox"/>	o=defaultWIMFileBasedRealm	InternalFileRepository	File

Total 2

Related Items

Additional Properties

- [Property extension repository](#)
- [Entry mapping repository](#)
- [Supported entity types](#)**
- [User repository attribute mapping](#)

- c. Click **Group**.

Preferences

Entity Type	Base Entry for the Default Parent	Relative Distinguished Name Properties
Group	o=netcoolObjectServerRepository	cn
OrgContainer	o=netcoolObjectServerRepository	o;ou;dc;cn
PersonAccount	o=netcoolObjectServerRepository	uid

Total 3



Important: Observe the values in the table that say `o=netcoolObjectServerRepository`. In the present state, if a new user is added to Dashboard Application Services Hub, an attempt is made to write the entry to the `netcoolObjectServerRepository`. This repository was removed in a previous step. Until the following steps are completed, it is not possible to add new Dashboard Application Services Hub users.

- d. Enter `ou=tipgroups,cn=tipRealm,DC=IBM,DC=COM` for Base entry for the default parent and click **OK**.

* Entity type
Group

* Base entry for the default parent
`ou=tipgroups,cn=tipRealm,DC=IBM,DC=COM`

* Relative Distinguished Name properties
cn

Apply OK Reset Cancel

- e. Click **OrgContainer**.

Entity Type	Base Entry for the Default Parent	Relative Distinguished Name Properties
You can administer the following resources:		
Group	<code>ou=tipgroups,cn=tipRealm,DC=IBM,DC=COM</code>	cn
OrgContainer	<code>o=netcoolObjectServerRepository</code>	<code>o;ou;dc;cn</code>
PersonAccount	<code>o=netcoolObjectServerRepository</code>	uid

- f. Enter `dc=ibm,dc=com` for Base entry for the default parent and click **OK**.

* Entity type
OrgContainer

* Base entry for the default parent
`dc=ibm,dc=com`

* Relative Distinguished Name properties
o;ou;dc;cn

Apply OK Reset Cancel

g. Click **PersonAccount**.

Entity Type	Base Entry for the Default Parent	Relative Distinguished Name properties
You can administer the following resources:		
Group	ou=tipgroups,cn=tipRealm,DC=IBM,DC=COM	cn
OrgContainer	dc=ibm,dc=com	o;ou;dc;cn
PersonAccount	ou=netcoolObjectServerRepository	uid

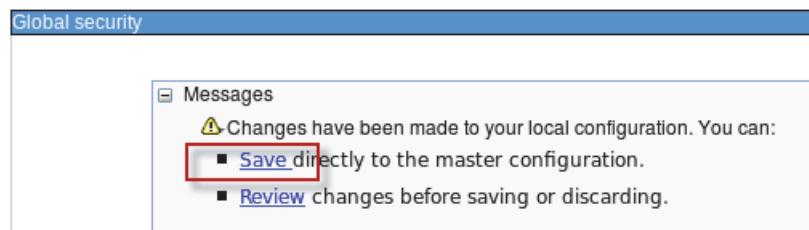
h. Enter **ou=tipusers,cn=tipRealm,DC=IBM,DC=COM** for Base entry for the default parent and click **OK**.

* Entity type	PersonAccount
* Base entry for the default parent	ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
* Relative Distinguished Name properties	uid
<input type="button" value="Apply"/> <input type="button" value="OK"/> <input type="button" value="Reset"/> <input type="button" value="Cancel"/>	

The revised entries are listed as shown.

Entity Type	Base Entry for the Default Parent	Relative Distinguished Name properties
You can administer the following resources:		
Group	ou=tipgroups,cn=tipRealm,DC=IBM,DC=COM	cn
OrgContainer	dc=ibm,dc=com	o;ou;dc;cn
PersonAccount	ou=tipusers,cn=tipRealm,DC=IBM,DC=COM	uid

i. Click **Save**.



5. Log out of administrative console.
6. Close the Firefox tab for the administrative console.
7. Log out of IBM Dashboard Application Services Hub.

8. Restart Dashboard Application Services Hub.

- Change to the **root** user.

```
su -
Password: object00
```

- Stop Dashboard Application Services Hub.

```
/etc/init.d/jazz stop
```

```
ADMU0116I: Tool information is being logged in file
/opt/IBM/JazzSM/profile/logs/server1/stopServer.log
ADMU0128I: Starting tool with the JazzSMPProfile profile
ADMU3100I: Reading configuration for server: server1
ADMU3201I: Server stop request issued. Waiting for stop status.
ADMU4000I: Server server1 stop completed.
```

- Start Dashboard Application Services Hub.

```
/etc/init.d/jazz start
```

```
ADMU0116I: Tool information is being logged in file
/opt/IBM/JazzSM/profile/logs/server1/startServer.log
ADMU0128I: Starting tool with the JazzSMPProfile profile
ADMU3100I: Reading configuration for server: server1
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process id is 6653
```

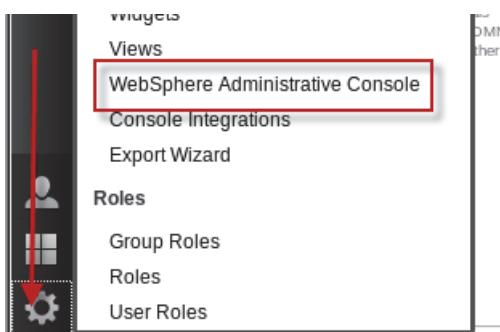
- Exit the **root** user and return to the **netcool** user.

```
exit
```

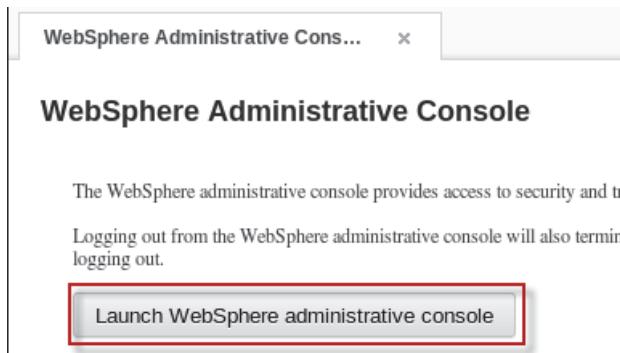
Dashboard Application Services Hub is now configured with two user repositories: internal file-based and LDAP. The LDAP users and groups that are located within the defined subtree are available within Dashboard Application Services Hub.

9. Verify that the LDAP users are available within Dashboard Application Services Hub.

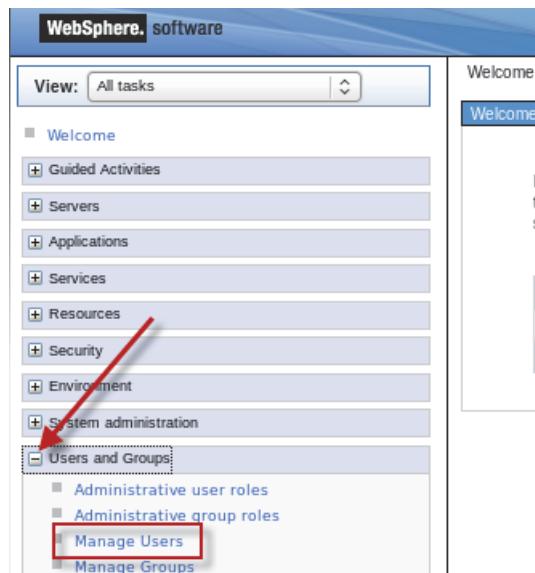
- Log in as the **smadmin** user with password **object00**.
- Click the icon and select **WebSphere Administrative Console**.



c. Click **Launch WebSphere Administrative Console**.



d. Expand **Users and Groups** and click **Manage Users**.



e. Observe the list of users.

Select	User ID	First name	Last name	E-mail	Unique Name
<input type="checkbox"/>	abraman	Ariana	Braman	abraman@ibm.com	cn=Ariana Braman,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	adurling	Adeline	Durling	adurling@ibm.com	cn=Adeline Durling,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	bwinebarger	Bart	Winebarger	bwinebarger@ibm.com	cn=Bart Winebarger,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	dselan	Dick	Selan	dselan@ibm.com	cn=Dick Selan,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	eange	Earline	Ange	eange@ibm.com	cn=Earline Ange,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	elotempio	Emelda	Lottempio	elotempio@ibm.com	cn=Emelda Lottempio,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	ezegarelli	Else	Zegarelli	ezegarelli@ibm.com	cn=Else Zegarelli,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	gbaillio	Gerald	Baillio	gbaillio@ibm.com	cn=Gerald Baillio,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	hdold	Houston	Dold	hdold@ibm.com	cn=Houston Dold,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	jguglielmo	Jasper	Guglielmo	jguglielmo@ibm.com	cn=Jasper Guglielmo,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	jmulberry	Jorge	Mulberry	jmulberry@ibm.com	cn=Jorge Mulberry,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM

Dashboard Application Services Hub is now aware of 27 users. Note the values in the Unique Name column of the table. These values indicate that the user is defined in the LDAP directory.

When one of these users logs in to Dashboard Application Services Hub, the Virtual Member Manager component uses the password that is defined in LDAP to authenticate the login.

The users are known to Dashboard Application Services Hub, but they do not belong to any group and they do not have any roles that are assigned yet. Therefore, they cannot perform any useful functions within Dashboard Application Services Hub. You add roles to some of these users in a subsequent unit.

10. Log out of administrative console.
11. Log out of Dashboard Application Services Hub.

Configuring Dashboard Application Services Hub to allow logins if LDAP is down

Dashboard Application Services Hub is configured to use two user repositories:

```
internal file-based  
LDAP
```

Dashboard Application Services Hub is based on WebSphere. There is a WebSphere property called *allowOperationIfReposDown*. The default setting for this property is false. When set to false, if one of the repositories is unavailable, users cannot log in to Dashboard Application Services Hub. If the property is true, and the LDAP server goes down, you can log in to Dashboard Application Services Hub as the **smadmin** user because that user is defined in the file-based repository.

To facilitate this exercise, a script is provided which runs a utility to change the value of the property to true.

1. Change directory to the location of the supplied script:

```
cd /workshop/dash
```

2. Examine the script as follows:

```
more wsadmin.sh
```

```
#!/bin/sh  
#  
# This script configures DASH to allow logins if not all  
# repositories are available.  
# This is required in order to use LDAP  
#  
# This script runs the wsadmin utility and passes it a jython command file  
cd /opt/IBM/JazzSM/profile/bin  
../wsadmin.sh -lang jython -user smadmin -password object00 -f /workshop/dash/ldap.py  
  
echo "Restart DASH to activate the changes"
```

This script calls the utility wsadmin.sh and passes the name of a Jython file.

3. Examine the Jython file as follows:

```
more ldap.py
```

```
AdminTask.updateIdMgrRealm ('[-name defaultWIMFileBasedRealm  
-allowOperationIfReposDown true]')  
AdminConfig.save()
```

The first line in the file contains the command sequence to change the property to true. The second line contains the command to save the revised property setting.

4. Run the script as follows:

```
./wsadmin.sh
```

```
WASX7209I: Connected to process "server1" on node JazzSMNode01 using SOAP  
connector; The type of process is: UnManagedProcess  
Restart DASH to activate the changes
```



Important: Dashboard Application Services Hub must be running to run this utility.

5. Restart Dashboard Application Services Hub.

- a. Change to the **root** user.

```
su -  
Password: object00
```

- b. Stop Dashboard Application Services Hub.

```
/etc/init.d/jazz stop
```

- c. Start Dashboard Application Services Hub.

```
/etc/init.d/jazz start
```

- d. Exit the **root** user back to the **netcool** user.

```
exit
```

To verify that the change works, you must temporarily stop the LDAP server on host1.

6. Switch to the **host1** image.

7. Stop the LDAP server as follows:

- a. Change to the **root** user.

```
su -  
Password: object00
```

- b. Stop the LDAP server.

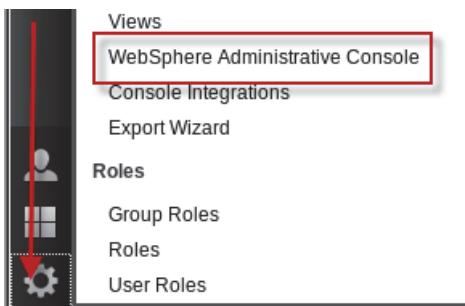
```
/etc/init.d/ibmslapd stop
```

```
Stopping SDS instance dsrdbm01 Stopping SDS Admin Server instance dsrdbm01  
[root]
```

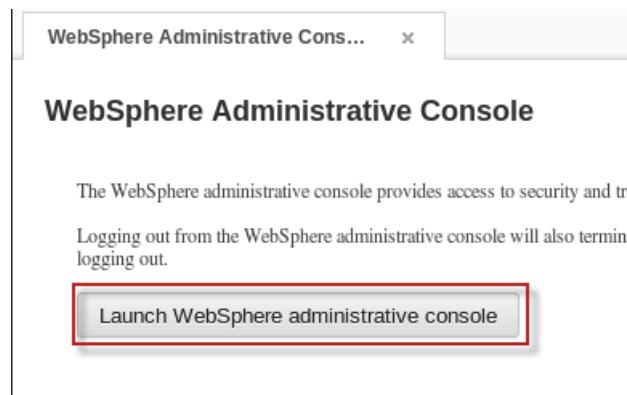


Important: Leave the terminal window as is. You return shortly and use it to restart the LDAP server.

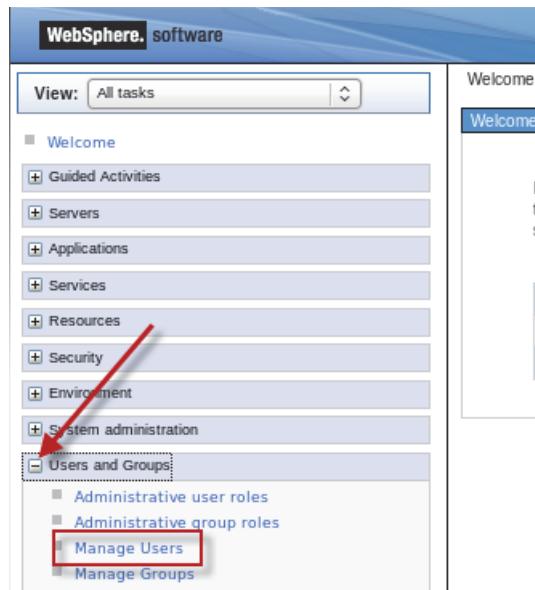
8. Switch to the **host2** image.
9. Verify that you can log in to Dashboard Application Services Hub.
 - a. Log in as the **smadmin** user with password **object00**.
The successful login verifies that the property change is correct.
 - b. Click the icon and select **WebSphere Administrative Console**.



- c. Click **Launch WebSphere Administrative Console**.



- d. Expand **Users and Groups** and click **Manage Users**.



- e. Observe the list of users.

1 users matched the search criteria.						
Select	User ID	First name	Last name	E-mail	Unique Name	
<input type="checkbox"/>	smadmin	smadmin	smadmin		uid=smadmin,o=defaultWIMFileBasedRealm	
Page 1 of 1				Total: 1		

Dashboard Application Services Hub is aware of only one user: **smadmin**.



Important: Leave the browser session as is. You return to it shortly.

10. Switch to the **host1** image.

11. Restart the LDAP server as follows:

- a. Start the LDAP server.

```
/etc/init.d/ibmslapd start
```

```
Starting SDS instance dsrdbm01 Starting SDS Admin Server instance dsrdbm01
[root]
```

- b. Exit the **root** user and return to the **netcool** user.

```
exit
```

12. Switch to the **host2** image.

13. Return to the administrative console session and click **Search**.

The screenshot shows a search interface titled 'Search for Users'. It has fields for 'Search by' (User ID), 'Search for' (containing 'abraman'), and 'Maximum results' (set to 100). A red arrow points to the 'Search' button. Below the search bar, a message says '27 users matched the search criteria.' A red box highlights this message. The main area is a table with columns: Select, User ID, First name, Last name, E-mail, and Unique Name. The table lists 27 users, each with a checkbox in the 'Select' column and a link in the 'User ID' column.

Select	User ID	First name	Last name	E-mail	Unique Name
<input type="checkbox"/>	abraman	Ariana Braman	Braman	abraman@ibm.com	cn=Ariana Braman,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	adurling	Adeline Durling	Durling	adurling@ibm.com	cn=Adeline Durling,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	bwinebarger	Bart Winebarger	Winebarger	bwinebarger@ibm.com	cn=Bart Winebarger,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	dselan	Dick Selan	Selan	dselan@ibm.com	cn=Dick Selan,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	eange	Earline Ange	Ange	eange@ibm.com	cn=Earline Ange,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	elotempio	Emelda Lotempio	Lotempio	elotempio@ibm.com	cn=Emelda Lotempio,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	ezegarelli	Else Zegarelli	Zegarelli	ezegarelli@ibm.com	cn=Else Zegarelli,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	gbaillio	Gerald Baillio	Baillio	gbaillio@ibm.com	cn=Gerald Baillio,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	hdold	Houston Dold	Dold	hdold@ibm.com	cn=Houston Dold,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM

All 27 users are again available.

14. Log out of administrative console.

15. Log out of Dashboard Application Services Hub.



Important: You must also modify the WebSphere property if you are using an ObjectServer repository instead of LDAP. The same process of modifying the WebSphere property applies.

Configuring ObjectServer synchronization

When the Web GUI component was installed, the postinstallation utility created two users, **ncoadmin** and **ncouser**, and two groups, **Netcool_OMNIbus_Admin** and **Netcool_OMNIbus_User**. The users and groups were created in the Dashboard Application Service Hub. The Virtual Member Manager contained a definition for the ObjectServer as a user repository. The Virtual Member Manager was also configured to write new users into the ObjectServer.

In the previous exercise, you removed the ObjectServer user repository, which removed the users and groups from Dashboard Application Services. When LDAP is used as a user repository for Dashboard Application Services Hub, it is suggested that you enable ObjectServer synchronization. When synchronization is enabled, any user changes in Dashboard Application Services Hub are *copied* to the ObjectServer.

1. Enable ObjectServer synchronization as follows:

a. Change to the required directory.

```
cd /opt/IBM/netcool/omnibus_webgui/etc
```

- b. Modify the Web GUI initialization file.

```
gedit server.init
```

Locate the following line:

```
users.credentials.sync:false
```

Change the property value to true:

```
users.credentials.sync:true
```

- c. Save the file.

Synchronization is performed at a defined frequency. The default frequency is every 3600 seconds. To facilitate the class exercises, you modify that setting and reduce the frequency.

- d. Change to the required directory.

```
cd /opt/IBM/netcool/omnibus_webgui/etc/datasources
```

- e. Modify the Web GUI initialization file.

```
gedit ncwDataSourceDefinitions.xml
```

Locate the following line:

```
<config maxAge="3600"/>
```

Change the property value to 600:

```
<config maxAge="600"/>
```

- f. Save the file.

2. Restart Dashboard Application Services Hub.

- a. Change to the **root** user.

```
su -  
Password: object00
```

- b. Stop Dashboard Application Services Hub.

```
/etc/init.d/jazz stop
```

- c. Start Dashboard Application Services Hub.

```
/etc/init.d/jazz start
```

- d. Exit the **root** user and return to the **netcool** user.

```
exit
```

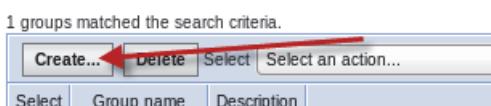
Configuring default users and groups

Now that the synchronization process is configured, you must re-create the default users and groups. You create the users and groups in Dashboard Application Services Hub. The synchronization process creates the same entries in the ObjectServer.

1. Log in to Dashboard Application Services Hub as user **smadmin** with password **object00**.
2. Start WebSphere Administrative console.
3. Add the default groups as follows:
 - a. Expand **Users and Groups** and click **Manage Groups**.



- b. Click **Create**.



- c. Enter **Netcool_Admin** as the group name and click **Create**.

- d. Click **Close**.



- e. Repeat the previous steps and create the **Netcool_User** group.

When complete, the groups are listed as follows:

3 groups matched the search criteria.				
Select	Group name	Description	Unique	
<input type="checkbox"/>	Netcool Admin	cn=Netcool_Admin,ou=tipgroups,cn=Groups,dc=tip,dc=com		
<input type="checkbox"/>	Netcool User	cn=Netcool_User,ou=tipgroups,dc=tip,dc=com		
<input type="checkbox"/>	WPAdministrators	cn=WPAdministrators,ou=tipgroups,dc=tip,dc=com		



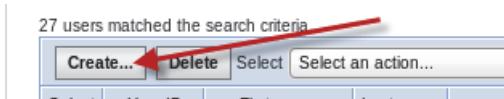
Note: The two groups are created in the LDAP directory.

4. Add the default users as follows:

a. Click **Manage Users**.



b. Click **Create**.



c. Enter **ncoadmin** as the user ID.

d. Enter values for first and last names.

e. Enter **object00** for the password and click **Group Membership**.

The screenshot shows the 'Create a User' form with the following fields filled in:

- *User ID: ncoadmin
- *First name: Netcool
- *Last name: Admin
- E-mail: (empty)
- *Password: object00
- *Confirm password: object00

A red arrow points from the 'Group Membership' button to the 'Group Membership' section of the form.

f. Click **Search**.

Group Membership

Specify the search criteria that you want to use to find the group.

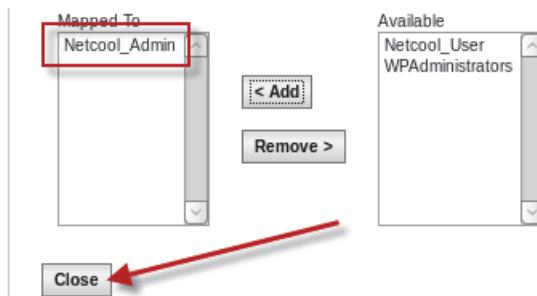
Search by * Search for * Maximum results
Group name [] 100

Search

g. Click **Netcool_Admin** to select it and click **Add**.



h. Click **Close**.



i. Click **Create**.

E-mail

* Password

* Confirm password

Create **Cancel**

j. Click **Close**.



k. Repeat the previous steps to create the **ncouser** user and assign the user to the **Netcool_User** group.

When complete, the user entries are listed as follows:

<input type="checkbox"/>	ncoadmin	Netcool	Admin		uid=ncoadmin,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM
<input type="checkbox"/>	ncouser	Netcool	User		uid=ncouser,ou=tipusers,cn=tipRealm,DC=IBM,DC=COM

l. Log out of administrative console.

m. Close the **Firefox** tab.

The users and groups are created in the LDAP directory and are now known to Dashboard Application Services Hub. However, no Dashboard Application Services Hub roles are assigned to either the users or the groups yet.

5. Assign Dashboard Application Services Hub roles to the default groups as follows.

- a. Click the icon and select **Group Roles**.



- b. Click **Search**.



- c. Click **Netcool_Admin**.

Group Name	Roles	Unique Name
Netcool_Admin		cn=Netcool_Admin,ou=tipgroups,cn=tipRealm,DC=IBM,DC=COM
Netcool_User		cn=Netcool_User,ou=tipgroups,cn=tipRealm,DC=IBM,DC=COM

- d. Scroll down and select the following roles:

iscadmin
ncw_admin
ncw_dashboard_editor
ncw_gauges_editor
netcool_rw



Important: The example screen capture does not show all required roles.

- e. Scroll to the bottom of the page and click **Save**.



- f. Click **Netcool_User**.

Group Name	Roles	Unique Nar
Netcool_Admin	ncw_gauges_editor, ncw_admin, ncw_dashboard_editor, iscadmins, netcool_rw	cn=Netcool
Netcool_User		cn=Netcool

- g. Scroll down and select the following roles:

ncw_user
ncw_gauges_viewer
netcool_rw

<input checked="" type="checkbox"/> ncw_gauges_viewer
<input checked="" type="checkbox"/> ncw_user
<input type="checkbox"/> netcool_ro
<input checked="" type="checkbox"/> netcool_rw

- h. Scroll to the bottom of the page and click **Save**.

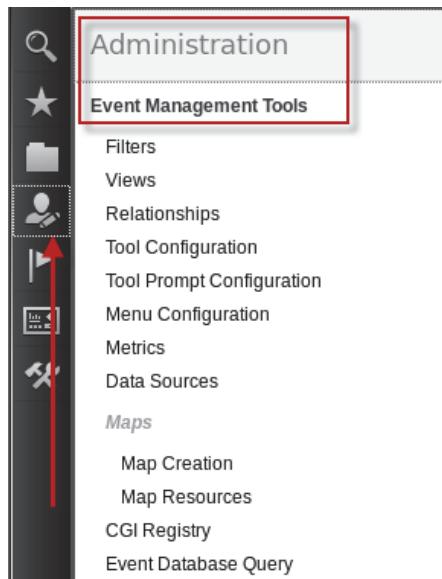


The role assignments are listed as follows:

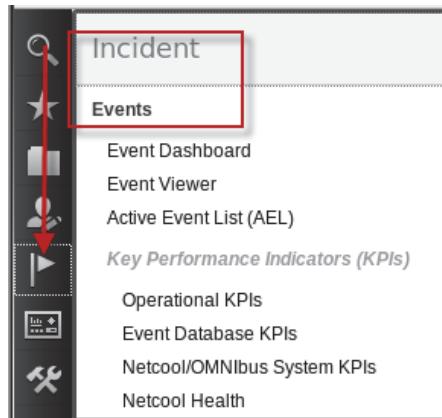
Group Name	Roles	Un
Netcool_Admin	ncw_gauges_editor, ncw_admin, ncw_dashboard_editor, iscadmins, netcool_rw	cn=
Netcool_User	ncw_user, ncw_gauges_viewer, netcool_rw	cn=

6. Log out of Dashboard Application Services Hub as the **smadmin** user.
7. Log in to Dashboard Application Services Hub as user **ncoadmin** with password **object00**.

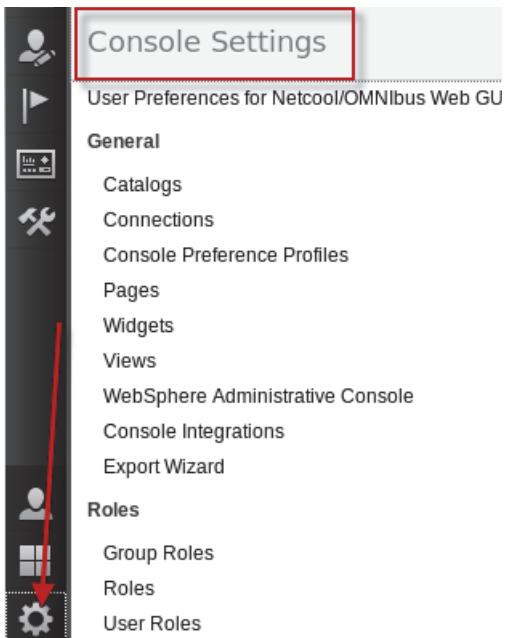
8. Click the icon and verify access to Netcool administrative features.



9. Click the icon and verify access to Netcool user features.



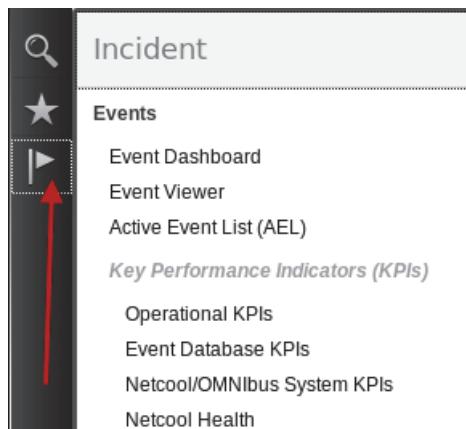
10. Click the icon and verify access to Dashboard Application Services Hub administrative features.



11. Log out of Dashboard Application Services Hub as the **ncoadmin** user.

12. Log in to Dashboard Application Services Hub as user **ncouser** with password **object00**.

13. Click the icon and verify access to Netcool user features.



14. Log out of Dashboard Application Services Hub.

15. Close the Firefox browser.

The following list is a summary of the accomplishments from this unit:

- Install Dashboard Application Services Hub.
- Install Netcool/OMNIbus Web GUI component.
- Verify basic Web GUI function.
- Configure Dashboard Application Services Hub to use LDAP as a user repository.



5 Basic ObjectServer administration exercises

The exercises in this unit provide an overview of Netcool/OMNibus directory structure, an introduction to the Administrator utility, and a demonstration of ObjectServer modifications.

Exercise 1 Overview of the Netcool/OMNibus directory structure

The structure of the Netcool®/OMNibus directory varies based on the operating system. In general, the files that are stored in the directory structure are of two basic types: text and executable (binary). The directory structure is created so that much of the text is stored in directories that do not vary based on the operating system. The executable files are stored in operating system-specific subdirectories. The product documentation references these directories as <arch>, where arch refers to a specific operating system, for example, **\$OMNIHOME/probes/linux2x86**. The value **linux2x86** is the operating system-specific reference for Linux installations.

The following exercises provide a brief overview of some of the more important directories. You perform these exercises on the host1 image.

1. Netcool/OMNibus application log files

This directory holds the log files for Netcool/OMNibus components, such as ObjectServers, probes, and gateways.

a. Change to the log file directory location:

```
cd $OMNIHOME/log
```

b. List the contents:

```
ls -1
```

There are entries for the primary ObjectServer (NYC_AGG_P), and the Simnet probe.

The entries for the primary ObjectServer include these:

```
NYC_AGG_P_audit_file.log  
NYC_AGG_P_dbinit.log  
NYC_AGG_P.log
```

```
NYC_AGG_P_profiler_report.log1  
NYC_AGG_P_selfmonitoring.log1  
NYC_AGG_P_trigger_stats.log1  
NYC_AGG_P_trigger_stats.log2
```

- c. Examine the audit log:

```
more NYC_AGG_P_audit_file.log
```

This file contains entries for ObjectServer authentication successes, and failures. The following ObjectServer property controls the content of this file: *Sec.AuditLevel*. Possible values are; debug, info, warn, and error. The default is warn. The debug, and info levels generate messages for authentication successes, and failures. The warn, and error levels generate messages for authentication failures only. The file is empty unless an authentication failure occurs.

- d. Examine the dbinit log:

```
more NYC_AGG_P_dbinit.log
```

This file contains the messages that are generated when an ObjectServer is created with the nco_dbinit command. For this workshop, the Initial Configuration Wizard ran the command.

- e. Examine the ObjectServer log:

```
more NYC_AGG_P.log
```

ObjectServer messages appear in this file. The following ObjectServer property controls the content of this file: *MessageLevel*. Possible values are: debug, info, warn, error, and fatal. The default level is Warn.



Hint: This file is the first place to look for any issues that relate to the ObjectServer, such as a failure to start.

- f. Examine the profile log:

```
more NYC_AGG_P_profiler_report.log1
```

This file contains ObjectServer profile statistics. The following two factors control the content of this file:

- ◆ ObjectServer property: *Profile*
- ◆ A set of ObjectServer triggers

The ObjectServer property controls ObjectServer profiling. If TRUE, the amount of time it takes for clients to run SQL is logged to the catalog.profiles table. The default is TRUE. The following trigger group manages the ObjectServer triggers: profiler_triggers. You must enable the trigger group, and individual triggers that are assigned to the group to collect and record profile statistics.

The profile log file shows a breakdown of the time that is spent for each client connection. The profile log also contains the total time that is used by client type. Each client that is

shown in the log file has a standard name, for example, GATEWAY or PROBE. The log file entry also shows the host on which the client is running. You can use the profile log file to analyze how the ObjectServer spent its time during each granularity period, and calculate the percentage of time used. For example, assume that the granularity period is set to 60 seconds, and that the total time spent for all the connections is 30 seconds. You can calculate that the ObjectServer spent 50% of its available time on running SQL commands from client connections.



Note: There might be more than one profile log file. The system is configured to record information into a single file until that file reaches a maximum physical size. After the file reaches the maximum size, the file is renamed and a new file is created. ObjectServer properties control the maximum file size, and the number of copies of the file to retain.

g. Examine the self-monitoring log:

```
more NYC_AGG_P_selfmonitoring.log1
```

This file contains ObjectServer health statistics. ObjectServer triggers collect the statistics. All of the triggers belong to the `self_monitoring_group` trigger group. Most of the triggers collect information about a specific Key Performance Indicator (KPI). The trigger compares the result to a user-defined threshold. If the threshold is exceeded, the trigger generates a synthetic event that indicates an issue with the specific metric.

h. Examine the trigger statistics log:

```
more NYC_AGG_P_trigger_stats.log1
```

This file contains ObjectServer trigger statistics. The following two factors control the contents of this file:

- ◆ ObjectServer property: `Auto.Enabled`
- ◆ A set of ObjectServer triggers

The ObjectServer property controls ObjectServer profiling. If TRUE, the amount of time it takes for clients to run SQL is logged to the catalog.profiles table. The default is TRUE. The following trigger group manages the ObjectServer triggers: `trigger_stats_report`. The trigger group and individual triggers that are assigned to the group must be enabled to collect, and record trigger statistics.

The trigger statistics log file shows the amount of time that each trigger used in the last profiling period. You can use this log file for automation debugging and to determine which triggers are slow because of slow-running SQL queries.

i. Examine the Simnet probe log:

```
more simnet.log
```

This file contains all messages for the Simnet probe. The following probe property controls the content of this file: `MessageLevel`. This log file is the first place to look for any issues that relate to the probe.

2. Netcool/OMNIbus communication files

Component communications are configured in two files.

- Change to the required directory:

```
cd $NCHOME/etc
```

- List the contents:

```
ls -l
```

There are two files of interest:

```
interfaces.linux2x86
```

```
omni.dat
```

- Examine the interfaces file:

```
more interfaces.linux2x86
```

This file is the machine-generated communications file. It is created by running one of the following utilities:

```
nco_igen
```

```
nco_xigen
```



Note: For this workshop, the Initial Configuration Wizard modified the **omni.dat** file. The Initial Configuration Wizard generated the interfaces file.

- Examine the omni.dat file:

```
more omni.dat
```

This file is the source that generates the interfaces file. Use the nco_xigen utility, or any text editor, to modify the file. If a text editor is used to modify the file, you must run the nco_igen utility to generate the interfaces file. If you modify the file with the nco_xigen utility, the nco_igen runs automatically when you exit the utility.



Hint: Many component failures are related to communication issues. If a component fails to start, check the log file for a communication issue. If the log file indicates a communication issue, then verify that the contents of the omni.dat file are correct. Typical issues include misspelled ObjectServer names, incorrect port values, or misspelled host names.

3. Switch to the **host2** image.

4. Netcool Web GUI log files

- Change to the required directory:

```
cd /opt/IBM/JazzSM/profile/logs/ncw
```

The Web GUI component runs as an application within the Dashboard Application Services Hub. The Web GUI log files are saved within the Jazz™ for Service Management directory structure.

- b. List the contents:

```
ls -l
```

The following three files are listed:

```
ncw.0.log  
ncw.0.profile  
ncw.0.trace
```

- c. Examine the log file:

```
more ncw.0.log
```

This file contains all of the messages that relate to the basic operation of the Web GUI component. This log file is the first place to look for any issues with the Web GUI, such as a failure to start, or a failure to connect to an ObjectServer. The following property controls the content of this file: *log.level*. The default value is INFO.



Hint: The Web GUI component connects to an ObjectServer to access event records. If event records are not available in Web GUI applets (event dashboard, active event list, and others), examine the log file for ObjectServer connection failure messages, for example:

```
SEVERE:HEMDA0265E:[NCWBringUpWorker]:Failed to initialize connection pool for  
jdbc:sybase:Tds:omnithost:4100. The server might be down.
```

- d. Examine the profile log:

```
more ncw.0.profile
```

This file contains statistics about the operation of the Web GUI component. These statistics are similar to the ObjectServer profile statistics.

- e. Examine the trace file:

```
more ncw.0.trace
```

This file contains more detailed information that relates to the basic operation of the Web GUI component.

5. Dashboard Application Services Hub log files.

- a. Change to the required directory:

```
cd /opt/IBM/JazzSM/profile/logs/server1
```

The reference to server1 indicates the Dashboard Application Services Hub component.

- b. List the contents:

```
ls -l
```

There are numerous files in this directory. The ones of primary importance are:

SystemErr.log

SystemOut.log

These files provide the details that relate to the operation of the Dashboard Application Services Hub. They should be the first place to look for any issues.



Hint: Both of these files grow continuously. New entries are added to the end of the file. When troubleshooting an issue, use the `tail` command to examine the end of the file. Or, if the issue is consistent, for example, failure to start, delete, or rename both files before restarting Dashboard Application Services Hub. New files are created with only the current entries.

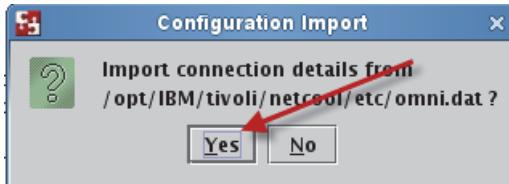
Exercise 2 Modifying the event record

One of the basic user operations is to delete an event record that is no longer needed in the ObjectServer. However, one issue is to ensure that a copy of the event is written to the archive database for historical reporting before it is removed from the ObjectServer. The combination of exercises in this unit demonstrates a technique to delete event records. This technique ensures that all event records that a user deletes are archived before they are removed from the ObjectServer. The exercises in this unit focus on the user configuration that is required to implement the suggested technique. Exercises in later units complete the solution.

As a part of the exercises, you implement a basic organizational structure. You create a functional group to organize your users. Next, you create a user that you assign to your group. As a part of this process, you create a restriction filter, which limits access to specific events.

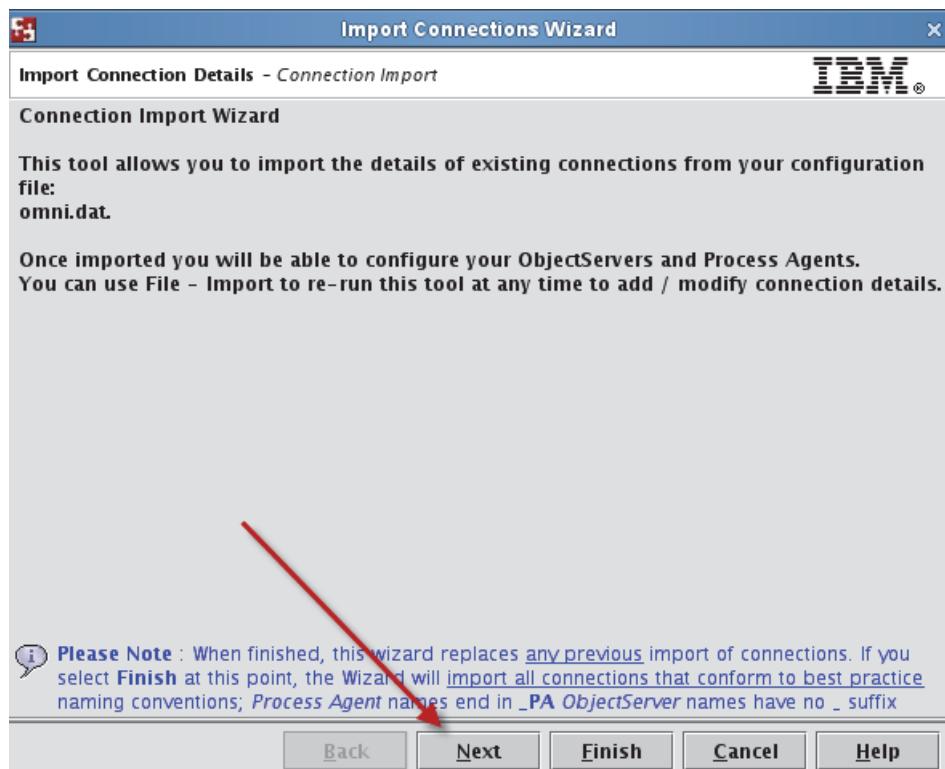
One of the most basic administrative tasks is to extend the ObjectServer event record by adding extra columns to the table definition. Extra columns are added for various reasons, including street address, city, state, postal code, circuit ID, carrier, and others. In this exercise, you add two columns to use as *flags*. Each flag indicates that the event meets a specific condition. One *flag* indicates that the event is a candidate for deletion. The second *flag* indicates that the event is successfully written to the archive database. The flag that indicates that an event is a candidate for deletion is used in the restriction filter.

1. Switch to the **host1** image.
2. Start the ObjectServer Administrator utility:
`nco_config &`
 - a. Click **Yes** to import the **omni.dat** file.



The Import Connections Wizard window opens.

- b. Click **Next**.



The list of available ObjectServers opens.

- c. Verify that the **NYC_AGG_P** ObjectServer is listed and click **Finish**.



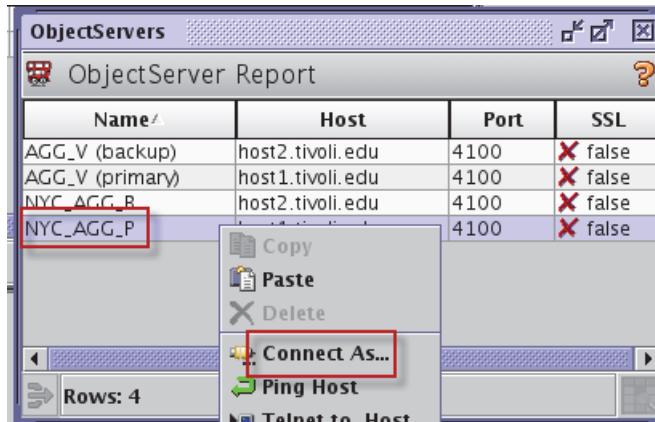
The Administrator utility opens.



Note: There are entries for NYC_AGG_B and AGG_V. The Netcool/OMNibus Initial Configuration wizard created the entries. The ObjectServers do not physically exist yet.

3. Add columns to the primary ObjectServer.

- a. Click **NYC_AGG_P**, right-click, and select **Connect As**



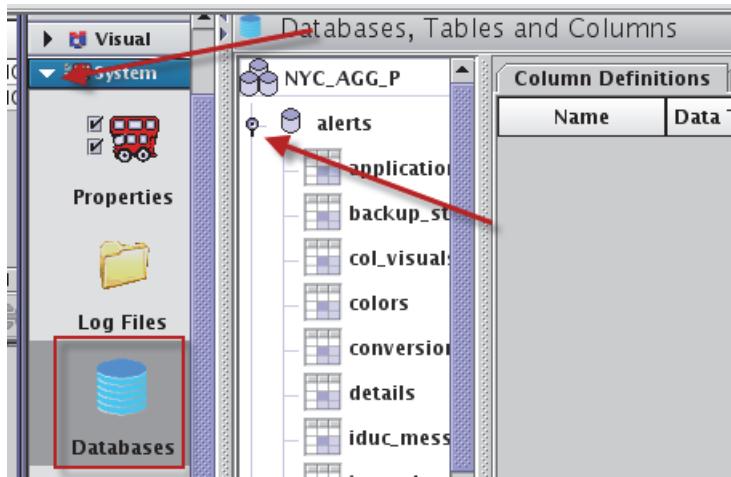
Hint: If **Connect As** is not in the list, it typically means that the ObjectServer is not available. If the ObjectServer is not running, start the ObjectServer, exit the Administrator utility, and try the connection again.

- b. Log in as the **root** user with password **object00**.

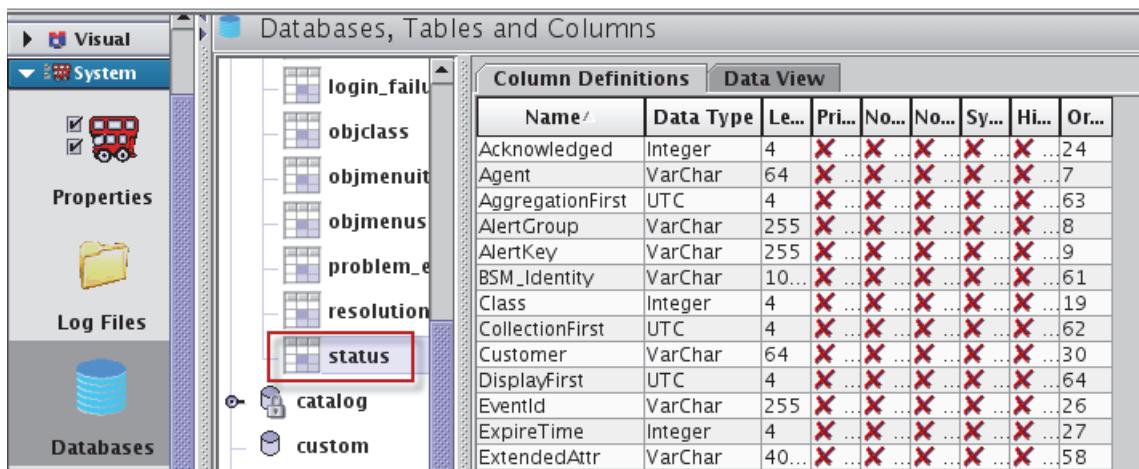


- c. Click the roll-up icon **System** to expand it.
- d. Click **Databases** to open the list of database names.

- e. Click the icon in front of the database that is labeled **alerts** to expand the list of tables.

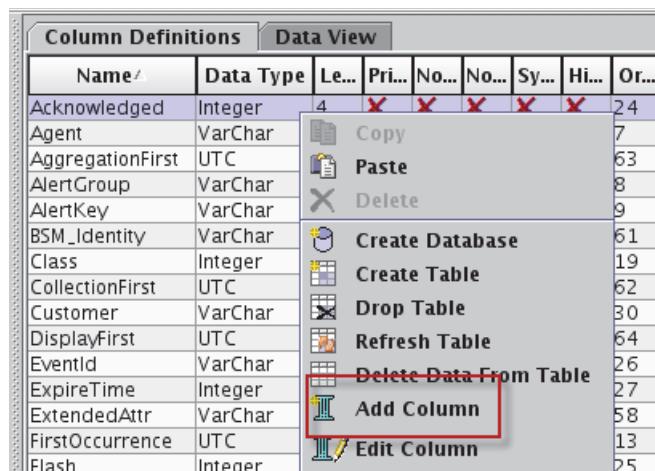


- f. Scroll down in the list of tables, and click the entry for **status**.



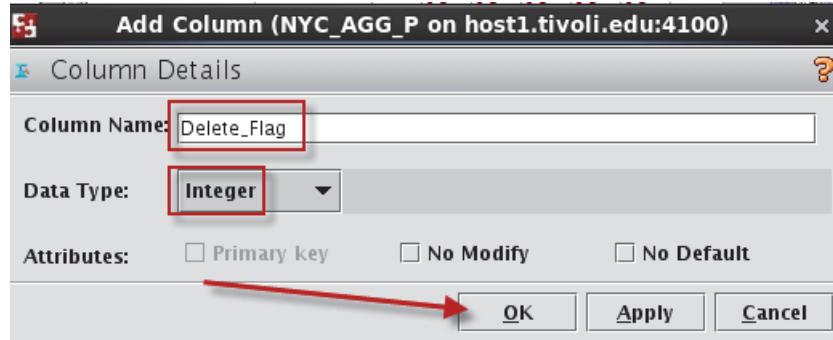
The list of column names in the alerts.status table is shown.

- g. Place the cursor anywhere in the list of column names, right-click, and select **Add Column**.



- h. Enter **Delete_Flag** for the column name.

- Select Integer as the data type and click OK.



- Repeat these steps, and add a second column **Archived_Flag** of type integer.

Name	Data Type	Len...	Pri...	No ...	No ...	Sy...
AlertGroup	VARCHAR	255	X	X	X	X
AlertKey	VARCHAR	255	X	X	X	X
Archived_Flag	Integer	4	X	X	X	X
BSM_Identity	VARCHAR	10...	X	X	X	X
Class	INTEGER	4	X	X	X	X
CollectionFirst	UTC	4	X	X	X	X
Customer	VARCHAR	64	X	X	X	X
Delete_Flag	Integer	4	X	X	X	X
DisallowFirst	UTC	4	X	X	X	X

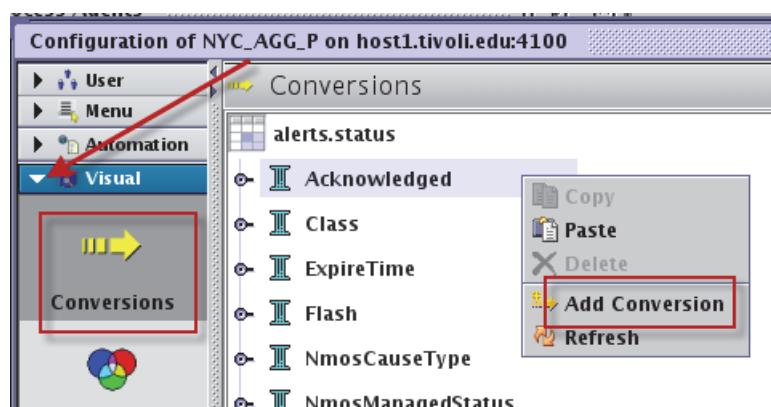
Verify that both column names are added as shown.

- Add entries to the conversions table.



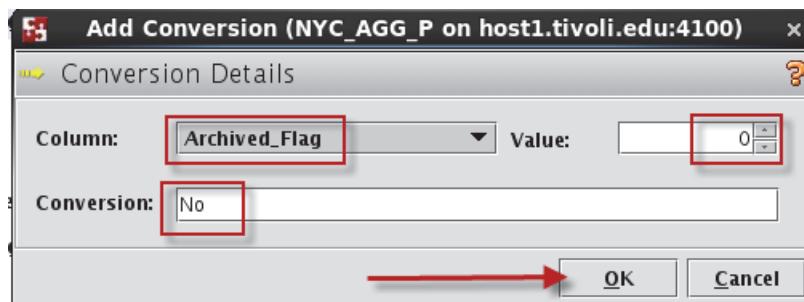
Hint: A best practice is to create conversion entries for all integer columns.

- Click the roll-up icon **Visual** to expand it.
 - Click the entry for **Conversions**.
- The list of conversion definitions opens.
- Place the cursor anywhere in the list of conversions, right-click, and select **Add Conversion**.



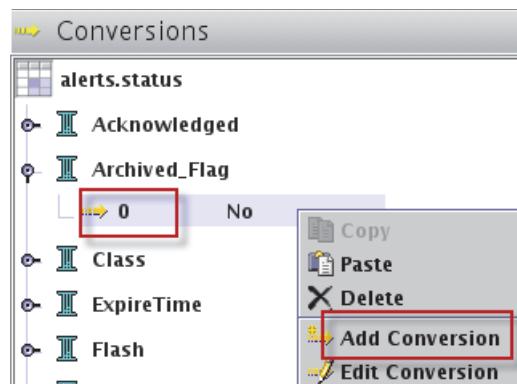
- Click the arrow and select the **Archived_Flag** column.

- e. Enter **0** for the Value.
- f. Enter the text **No** for the conversion, and click **OK**.

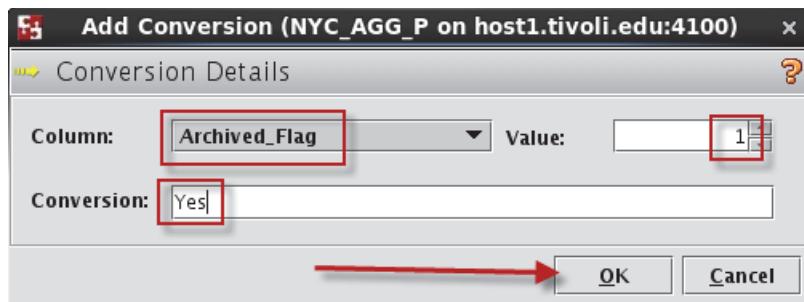


This action creates a conversion of **No** for the **Archived_Flag** column value of 0.

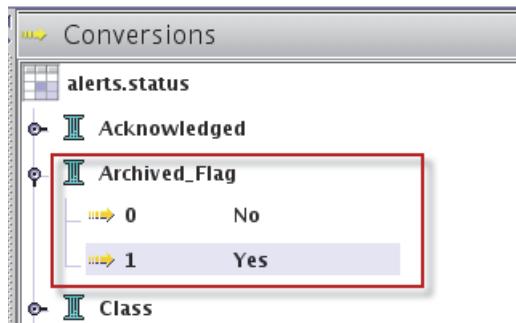
- g. Click the conversion entry for **0**, right-click, and select **Add Conversion**.



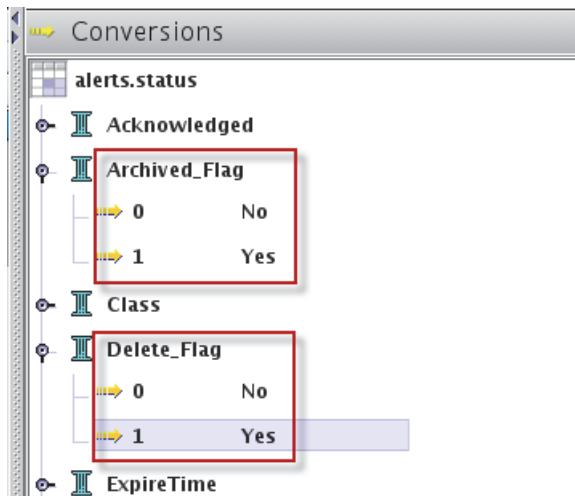
- h. Leave the Column as **Archived_Flag**.
- i. Leave the Value as **1**.
- j. Enter the text **Yes** for Conversion, and click **OK**.



There are now two conversion values for the **Archived_Flag** column.



- k. Repeat these steps, and create the same conversion values for the **Delete_Flag** column.



The completed conversions look like the previous screen capture.



Important: Leave the Administrator tool open. You return to it shortly.

Exercise 3 Creating users, and groups

In this exercise, you implement a basic organizational structure. You create an operations group. You modify a user, and assign that user to the operations group. You also create a restriction filter, and associate that filter with the user. The purpose behind the filter is to restrict access to events that meet a certain criteria. The criteria is based on the **Delete_Flag** column that you created previously.

1. Create the group as follows.

- a. Open a Firefox browser.

The browser home page is configured to open to the URL for Dashboard Application Services Hub.

- b. Log in as the **smadmin** user with password **object00**.
- c. Start WebSphere® Administrative console.
- d. Expand **Users and Groups**.
- e. Click **Manage Groups**.
- f. Click **Create**.
- g. Enter **Operations** for group name and click **Create**.

The screenshot shows a 'Create a Group' dialog box. At the top is a title bar with the text 'Create a Group'. Below it is a form field labeled 'Group name' containing the text 'Operations'. There is also a small red asterisk icon next to the label.

The LDAP directory contains a collection of users. Currently, none of those users are configured for access to Netcool/OMNIbus Web GUI features. For this exercise, you configure user William Hill as a user who belongs to the new Operations group.

- h. Click **Manage Users**.
- i. Enter **whill** in the search for box and click **Search**.

The screenshot shows a 'Search for Users' dialog box. It has fields for 'Search by' (User ID), 'Search for' (containing 'whill'), and 'Maximum results' (set to 100). A large red arrow points to the 'Search' button at the bottom.

- j. Click **whill**.

The screenshot shows a table of user search results. The header row includes columns for 'Select', 'User ID', 'First name', 'Last name', 'E-mail', and 'uid'. A single user entry for 'whill' is shown, with a blue arrow pointing to the 'Select' checkbox in the first column.

Select	User ID	First name	Last name	E-mail	uid
<input type="checkbox"/>	whill	William	Hill		

- k. Click the **Groups** tab.

The screenshot shows a 'User Properties' dialog box. At the top is a title bar with the text 'User Properties'. Below it is a tab navigation bar with 'General' and 'Groups' tabs; the 'Groups' tab is highlighted with a blue background. Below the tabs is a form field labeled 'User ID' containing 'whill'. There is also a small red asterisk icon next to the label.

The user does not currently belong to any groups.

- l. Click **Add**.

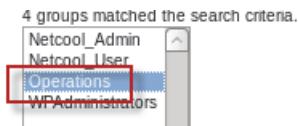
The screenshot shows a 'User Properties' dialog box with the 'Groups' tab selected. At the top is a title bar with the text 'User Properties'. Below it is a tab navigation bar with 'General' and 'Groups' tabs; the 'Groups' tab is highlighted with a blue background. Below the tabs is a form field labeled 'User ID' containing 'whill'. There is also a small red asterisk icon next to the label. A message below the field states 'The user is a member of 0 groups.' At the bottom is a button bar with 'Add...' and 'Remove' buttons, with a large red arrow pointing to the 'Add...' button.

m. Click **Search**.

Search by
Group name
* Search for
* Maximum results
100
Search

n. Click **Operations** to select it.

o. Click **Add**.



p. Click **Close**.

Add a User to Groups
i The user was added to the groups successfully.

q. Click the **General** tab.

User Properties
General Groups
* User ID
whill

r. Enter *object00* for the password and click **OK**.

Password Confirm password
***** *****
OK Apply Cancel

This user exists and you do not know what the password might be. So change the password to a known value.

2. Log out of WebSphere Administrative console

3. Close the Firefox tab.

You created the Operations group, but the group does not contain any roles.

4. Add roles to the Operations group.

a. Click the icon and select **Group Roles**.



b. Click **Search**.

Group ID: _____

Description: _____

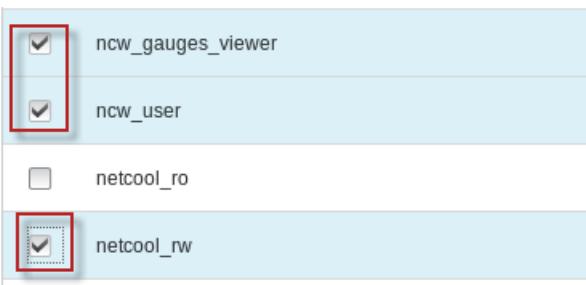
Number of results to display:
20

Search

c. Scroll down and click **Operations**.

Netcool_User	ncw_user, ncw_gauges_viewer, netcool_rw
Operations	
WPAdministrators	

d. Select **ncw_gauges_viewer**, **ncw_user**, and **netcool_rw**. Click **Save**.



5. Log out of Dashboard Application Services Hub as the **smadmin** user.
6. Return to the Netcool/OMNIbus Administrator tool that you used previously.
7. Expand the **User** feature and click **Groups**.

Name	Description
Administrator	Admin Group
Gateway	Permissions
ISQL	Read only
ISQLWrite	Write ISQL access
Netcool_Admin	VMM synchronised groups
Netcool_User	VMM synchronised groups
Normal	Normal Group
Operations	VMM synchronised groups
Probe	Permissions required for a probe user
Public	Public Group
System	System Group
vmmusers	Group to manage VMM users

Observe the group names that are high-lighted in the sample screen capture. The group names were created in Dashboard Application Services Hub. The synchronization process automatically added them to the ObjectServer.

You configured synchronization to run every 600 seconds. You might not see the Operations group yet.



Important: In a production environment where you are not using LDAP, the group names are also added automatically to the ObjectServer. In that configuration, it is not the synchronization process that adds the entries to the ObjectServer. Instead, Dashboard Application Services Hub is configured to write new entries directly to the ObjectServer.

8. Click **Users**.

Name	Full Name	Type	UID
ncoadmin	Netcool Admin	Normal	1
ncouser	Netcool User	Normal	2
nobody	Nobody	Unknown	65534
root	Root User	Super User	0
whill	William Hill	Normal	3

Observe the user names that are high-lighted in the sample screen capture. The user names were created in Dashboard Application Services Hub. The synchronization process automatically added them to the ObjectServer.

You configured synchronization to run every 600 seconds. The **whill** user might not show yet.

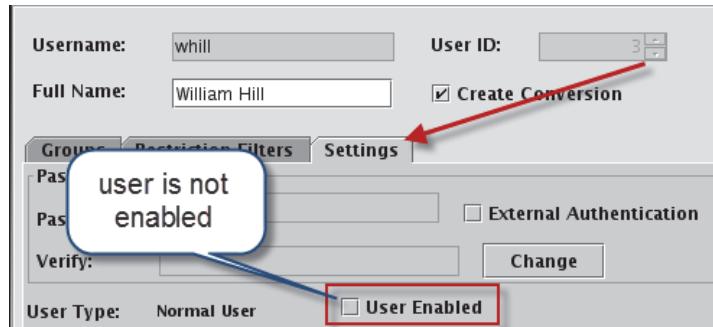
Notice that not all LDAP users are added to the ObjectServer by the synchronization process. Only those users that have the *ncw_admin* or *ncw_user* role that is assigned are added to the ObjectServer.



Important: You configured synchronization to run every 600 seconds. The **whill** user might not be shown yet. Click **Groups** and click **Users** a few times until **whill** is listed.

9. Click **whill**, right-click, and select **Edit User**.

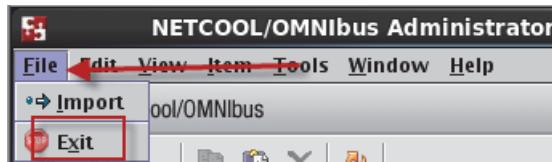
10. Click the **Settings** tab.



Important: The user is not enabled. If William Hill uses only Dashboard Application Services Hub to access the ObjectServer event records, the user definition in the ObjectServer does not have to be enabled. If William Hill uses the Native Desktop, then the user definition must be enabled.

11. Click **Cancel**.

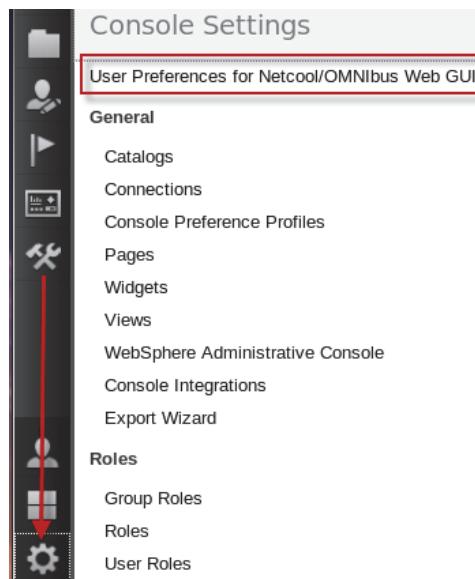
12. Close the Netcool/OMNIBus Administrator tool.



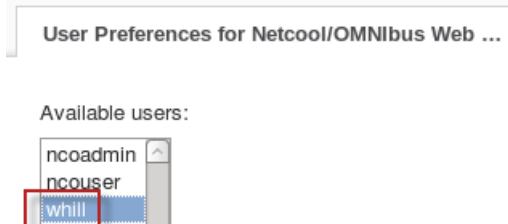
The last step is to create a restriction filter that limits the events records that William Hill can view. This filter is created in Dashboard Application Services Hub.

13. Log in to Dashboard Application Services Hub as user **ncoadmin** with password **object00**.

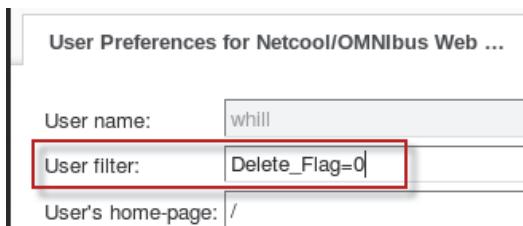
14. Click the icon and select **User Preferences for Netcool/OMNIBus Web GUI**.



15. Click **whill** to select it, scroll to the bottom of the page and click **Modify**.



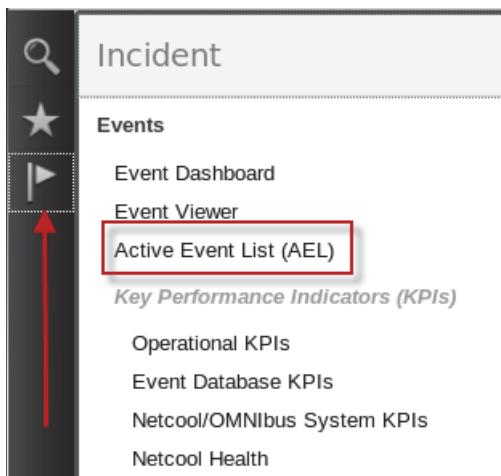
16. Enter **Delete_Flag=0** for user filter, scroll to the bottom of the page, and click **Save**.



17. Log out of Dashboard Application Services Hub as the **ncoadmin** user.

18. Validate the user.

- Log in to Dashboard Application Services Hub as user **whill** with password **object00**.
- Click the icon and select **Active Event List**.



You are able to log in to Dashboard Application Services Hub as **whill**. This action confirms that **whill** is a valid Dashboard Application Services Hub user. Access to the Incident and Events page confirms that the **whill** user has access to the Web GUI capability.

Membership in the **Netcool_User** group, which contains the **ncw_user** role, grants this access authority.

- c. Click any event record to select it, right-click, and observe the menu that opens.

Sev	Ack	Node	Alert Group	Summary
critical	Yes	Sydney	Acknowledge	Ctrl+A gone offline
warning	No	Beijing	De-acknowledge	Ctrl+D alert
warning	No	Moscow	Prioritize	> gone offline
warning	No	link6	Suppress/Escalate	> n port
warning	No	link2	Take ownership	n port
warning	No	London	User Assign	> gone offline
warning	No	link3	Group Assign	> n port
warning	No	Washington	Delete	gone offline
warning	No	link4		n port

Most of the tools that are listed require event update authority. Observe the Delete tool. The fact that these tools are in the list confirms that the **whill** user has update authority (WRITE access). The *netcool_rw* role, which is assigned to the Netcool_User group, grants this authority.

- d. Log out of Dashboard Application Services Hub as the **whill** user.

You modified the ObjectServer event record to include two custom columns: Archived_Flag, and Delete_Flag. You modified the **whill** user to grant access to ObjectServer event records. You created a restriction filter for the **whill** user in Dashboard Application Services Hub. You verified that you are able to log in to Dashboard Application Services Hub as the **whill** user. You verified that the user has the expected access authority.

It is probably not obvious, but you also validated the restriction filter. The default value for all columns of type Integer in the ObjectServer is 0. The restriction filter specifies Delete_Flag=0. At the moment, the value of the Delete_Flag column in all of the event records is 0, the default. The fact that the **whill** user sees event records when logged in to Dashboard Application Services Hub verifies that the restriction filter works as expected.

You could make other changes, such as allowing a user to delete an event record, but ensuring that the record is archived before it is removed from the ObjectServer.

The following list is a summary of accomplishments for this unit:

- Added Archived_Flag column to the ObjectServer
- Added Delete_Flag column to the ObjectServer
- Created Operations group
- Modified **whill** user. This user sees only events where Delete_Flag=0



6 Tools and automations exercises

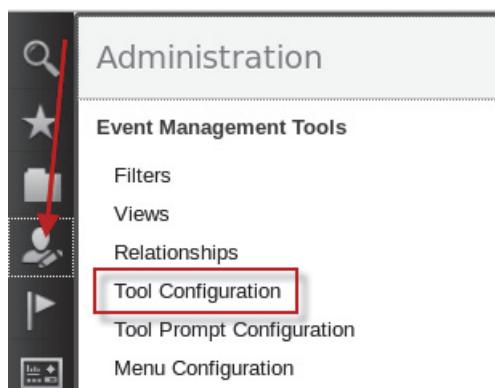
The exercises in this unit continue the implementation of the solution to delete events. In this unit, you create a custom tool that seems to delete an event record. In reality, the tool modifies the Delete_Flag column, and the restriction filter prevents access to the event. This action gives the user the impression that the event is deleted. In addition to the tool, you create an ObjectServer trigger. It is the trigger that removes the event record from the ObjectServer. The trigger removes the event only if certain conditions are met.

Exercise 1 Creating a Web GUI tool

You can define event-based tools in two locations: ObjectServer and Web GUI. A tool that is defined in an ObjectServer is available only to the native desktop client. A tool that is defined in Web GUI is available only to the browser-based client. In this exercise, you create a Web GUI tool.

Perform the following steps on the **host1** image:

1. Create a tool.
 - a. Open a Firefox browser, if necessary.
 - b. Log in to Dashboard Application Services Hub as the **ncoadmin** user.
 - c. Click the icon and select **Tool Creation**.



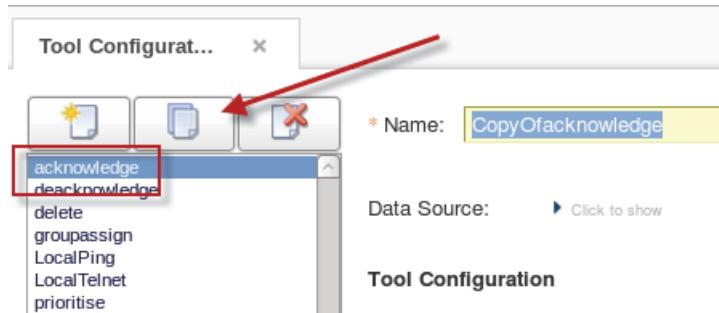
The list of available tools opens.



Hint: The *ncw_admin* Dashboard Application Services Hub role allows access to these pages.

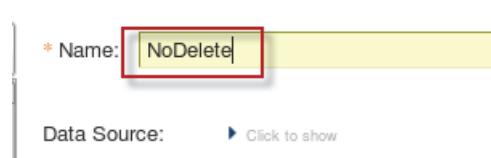
An easy way to create a tool is to copy the configuration of an existing tool and to change the tool to fit your needs.

- d. Click the tool that is named **acknowledge** to select it.
- e. Click the indicated icon to create a copy of the tool.



A copy of the definition for the tool opens.

- f. Change the name to **NoDelete**.



The acknowledge tool modifies the Acknowledged column. You want to modify the Delete_Flag column.

- g. Change the SQL command to update the Delete_Flag column as shown.



- h. Scroll down in the page to the **Access Criteria** dual list box.

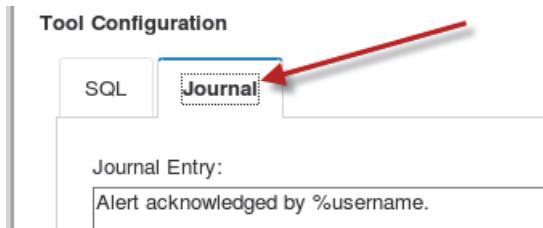
- i. Select **Operations** in the Available list, and click the right arrow to move it to the **Selected** list as shown.



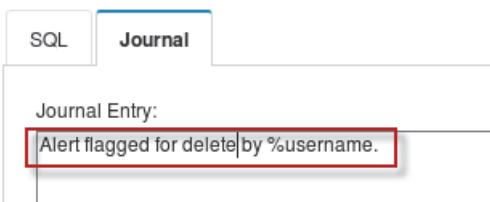
This action configures the tool to limit access only to users that belong to the Operations group.



- j. Scroll up to the top of the page and click the **Journal** tab.

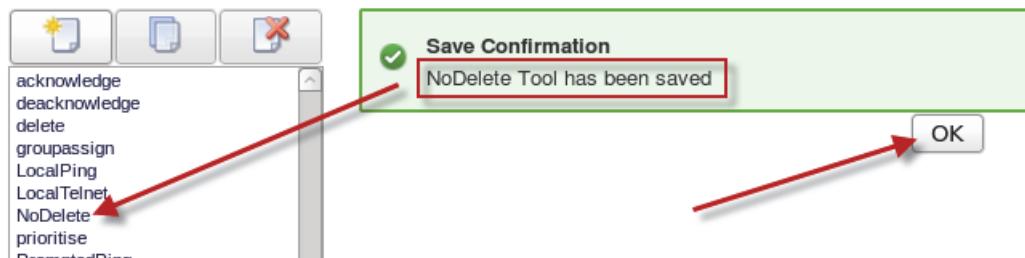


- k. Change the Journal Entry to indicate that the event is deleted.



- l. Scroll down to the bottom of the page, and click **Save**.

- m. Click **OK** to close the window.



2. Modify the existing delete tool.

The next step is to modify the definition for the existing delete tool. You modify the tool so that it is no longer accessible by users that belong to the Operations group. You do not want these users to delete events. But you still want other users to be able to delete events.

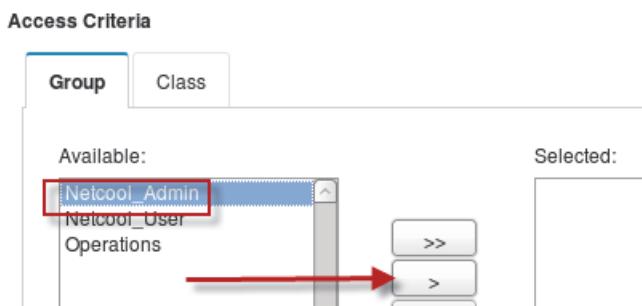
- a. Click the entry for the **delete** tool.

The tool configuration appears.

- b. Scroll down to the bottom of the page.

- c. Click the **Netcool_Admin** group in the **Available** list to select it.

- d. Click the right arrow to move the group to the **Selected** list.



The selected group appears as shown.



- e. Repeat these steps to add the **Netcool_User** group.

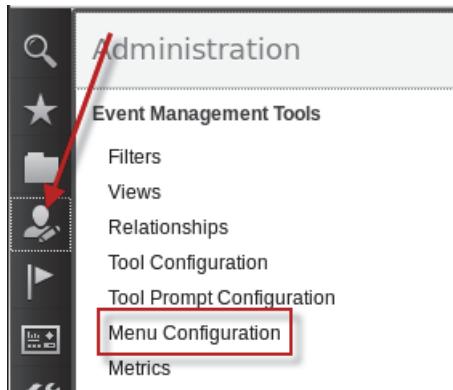
- f. Click **Save** to update the tool.

This action modifies the original delete tool so that it is available only to members of the Netcool_Admin or Netcool_User group. The whill user does not belong to either group, so the tool is no longer available to this user.

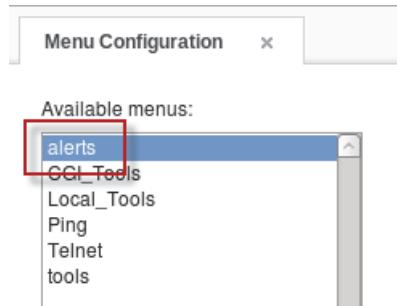
3. Add the NoDelete tool to a menu.

Because the NoDelete tool is not associated with any menu, it cannot be used.

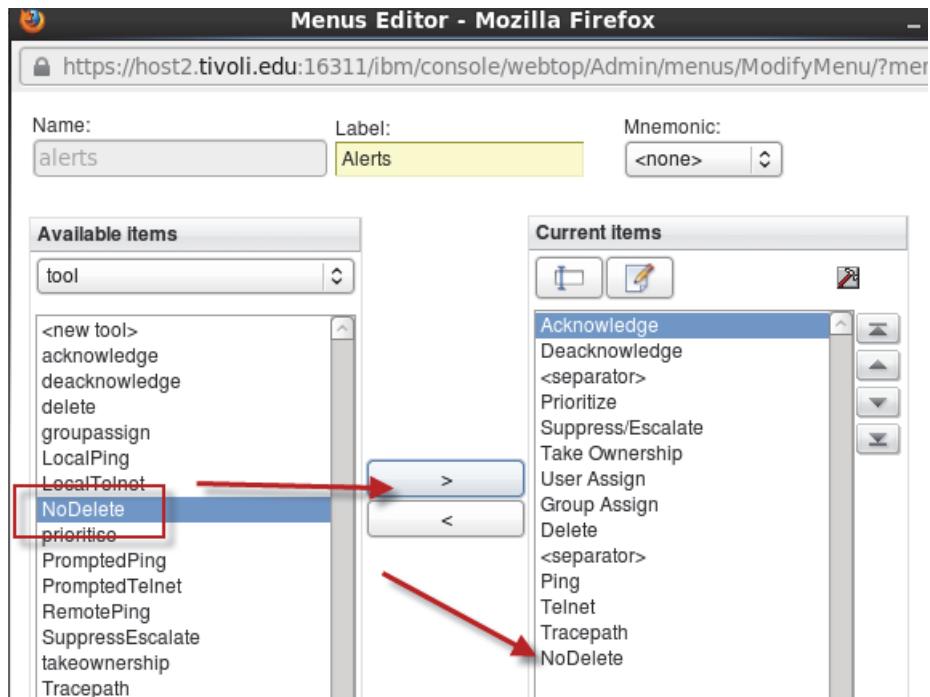
- Click the icon and select **Menu Configuration**.



- Click the **alerts** menu to select it, scroll to the bottom of the page, and click **Modify**.

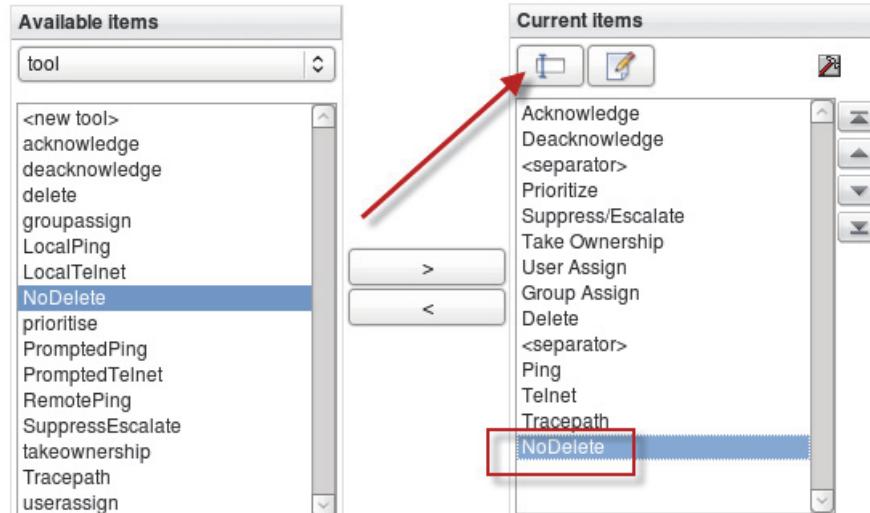


- Click **NoDelete** to select it, and click the right arrow icon to add the tool to the **Current items** list.

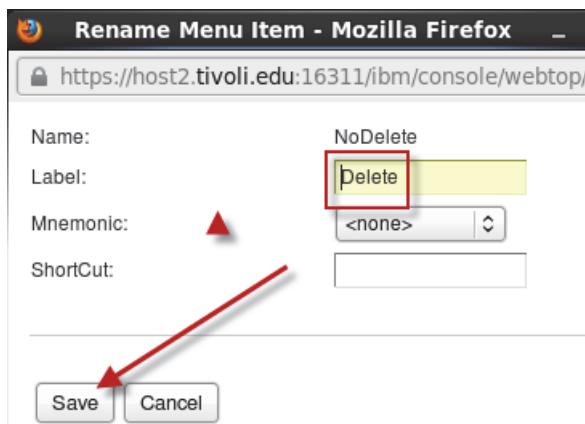


The NoDelete tool is added to the *bottom* of the alerts menu. You can use the up arrow and down arrow on the far right to move the tool up or down in the menu. For the purposes of this exercise, leave the tool on the bottom of the **Current items** list.

- d. Click the icon indicated to change the name of the tool as it is listed in the menu.

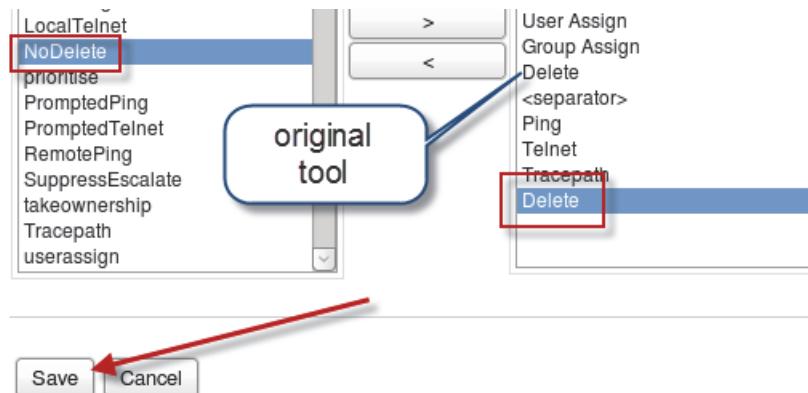


- e. Change the Label to **Delete** and click **Save**.



Important: This action does not change the tool name. It merely changes the text that shows in the menu. The tool is still named NoDelete.

- f. Click **Save** to modify the alerts menu.



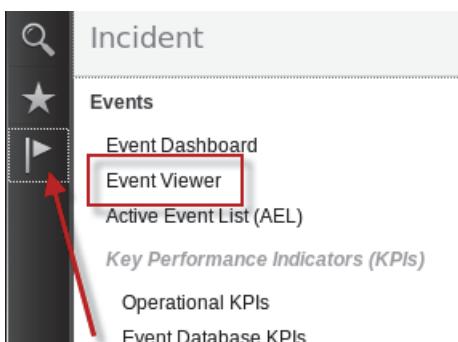
Observe that the tool name on the left side is **NoDelete**, and the list entry on the right side is **Delete**. The text *Delete* is what the user sees in the desktop.

- g. Log out of Dashboard Application Services Hub as the **ncoadmin** user.

4. Validate the tool.

The NoDelete tool sets the value to **1** for the Delete_Flag column. The tool is configured to be accessible only by members of the Operations group. The **whill** user, a member of the group, has a restriction filter that is associated with it that prevents access to any event where the value of the Delete_Flag column equals **1**.

- Log in to Dashboard Application Services Hub as user **whill** with password **object00**.
- Click the icon and select **Event Viewer**.



- c. Click an event record to select it, right-click, and select **Delete**.

Sev	Ack	Node	Alert Group	Sum
!	No	London	Systems	MacI
!	No	Beijing	Disk	
!	No	link3	Link	
!	No	link2	Link	
!	No	Washington	MacI	
!	No	Moscow	MacI	
!	No	link6	Link	
!	No	link4	Link	

In this example, the event for Beijing is deleted.

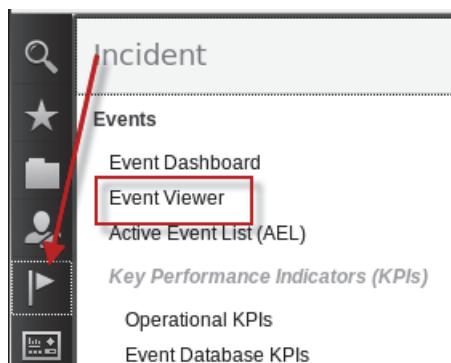


Note: Notice where the Delete tool shows in the menu: below the Ping tool.

The event record for Beijing is no longer listed.

Sev	Ack	Node	Alert Group
!	No	London	Systems
!	No	link3	Link
!	No	link2	Link
!	No	Washington	Systems
!	No	Moscow	Systems

- d. Log out of Dashboard Application Services Hub as the **whill** user.
e. Log in to Dashboard Application Services Hub as the **ncoadmin** user.
f. Click the icon and select **Event Viewer**.



The event record for Beijing appears in the event list.

Sev	Ack	Node	Alert Group	Sur
✖	Yes	Sydney	Systems	Ma
⚠	No	London	Systems	Ma
⚠	No	Beijing	Stats	Dis
⚠	No	link3	Link	Lin
⚠	No	link2	Link	Lin

- g. Double-click the event record for Beijing and click **Journals**.

Properties for event 40 on OMNIBUS

User ID	Date/Time	Journal Entry
whill	Wed Sep 24 18:36:37 GMT 2014	Alert deleted by whill.

The journal entry states that the **whill** user deleted the event. The event is not visible to the **whill** user, but it is still present in the ObjectServer.

5. Log out of Dashboard Application Services Hub as the **ncoadmin** user.

The next step in the configuration of the event behavior is to implement the feature to remove the event from the ObjectServer. A custom ObjectServer trigger removes the event.

Exercise 2 Creating a custom trigger

In this exercise, you create an ObjectServer trigger that removes events that match a specific criteria. The constraints of the criteria are as follows:

- The event is a candidate for deletion. This criteria is defined as Delete_Flag = 1.
- A copy of the event is written to the event archive database. This criteria is defined as Archived_Flag = 1.

In addition to the trigger for removing the *deleted* events, you create a trigger to remove old ObjectServer event records. This type of trigger is considered a best practice, and helps to keep the ObjectServer from becoming cluttered with non-essential event records. This trigger is often called the *reaper* trigger.

1. Start the ObjectServer administrator utility.

a. Run the following command:

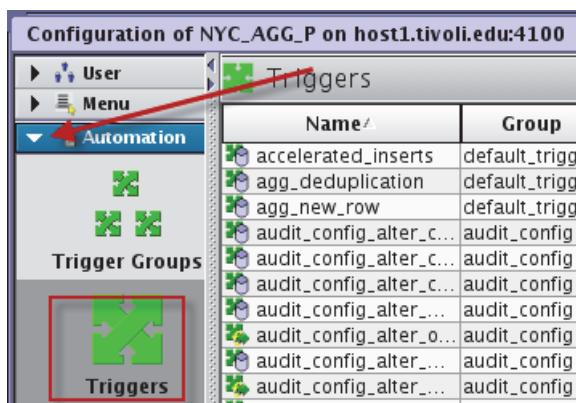
```
nco_config &
```

b. Connect to the **NYC_AGG_P** ObjectServer as the **root** user with password **object00**.

2. Create the trigger.

An easy way to create a custom trigger is to copy an existing trigger definition.

a. Expand the **Automation** section and click **Triggers**.



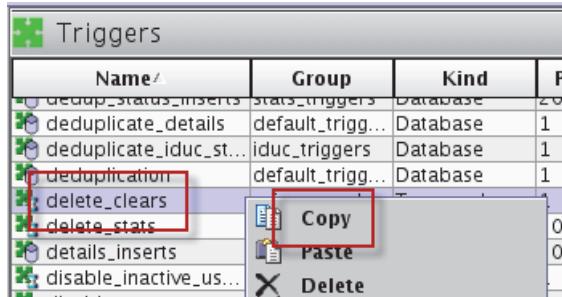
Configuration of NYC_AGG_P on host1.tivoli.edu:4100	
User	Triggers
▶ User	
▶ Menu	
▼ Automation	
	Trigger Groups
	Triggers

The screenshot shows the 'Automation' section expanded. A red arrow points to the 'Triggers' icon under the 'Automation' section. To the right is a table listing various triggers with their names and groups.

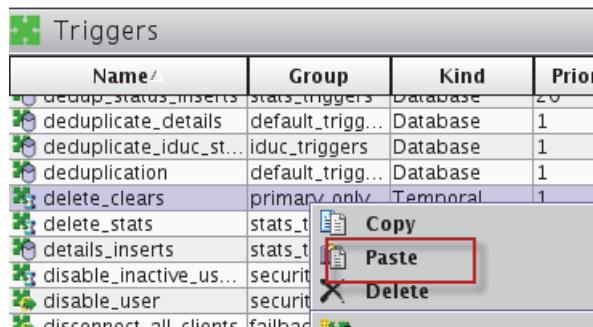
Name	Group
accelerated_inserts	default_trigg.
agg_deduplication	default_trigg.
agg_new_row	default_trigg.
audit_config_alter_c...	audit_config
audit_config_alter_o...	audit_config

The list of available triggers opens.

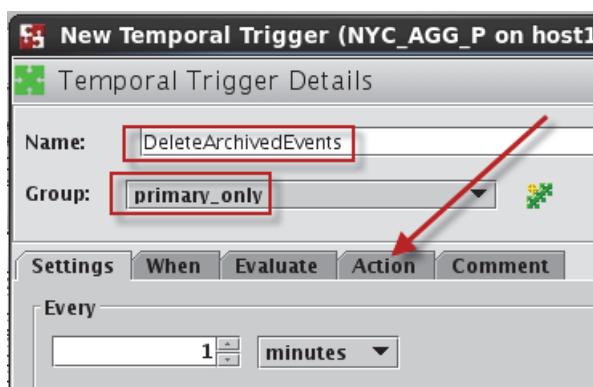
- b. Click the existing trigger **delete_clears** to select it, right-click, and select **Copy**.



- c. Right-click and select **Paste**.



- d. Enter **DeleteArchivedEvents** for the name.
e. Leave **Group** set to **primary_only**.
f. Click the **Action** tab.



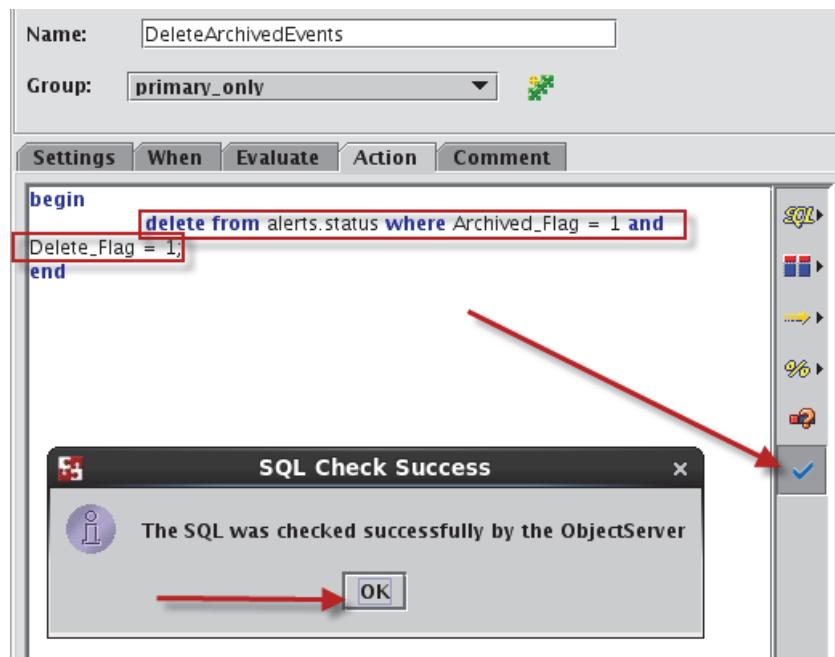
Important: Be sure to configure triggers that modify or delete event records so that they are active on only the primary ObjectServer, which minimizes the potential of a race condition with the bidirectional gateway. The **primary_only** trigger group is the preferred location for any triggers that modify event records.

You can now see the SQL from the `delete_clears` trigger. You must modify the text after the word **where** to suit your needs.

- g. Change the text as follows:

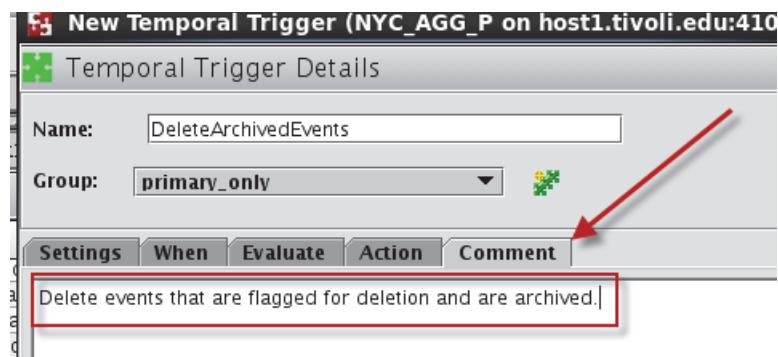
```
delete from alerts.status where Archived_Flag =1 and Delete_Flag = 1;
```

- h. Select the check mark to verify the syntax.



- i. Click **OK**.

- j. Click the **Comment** tab, and enter a description.



- k. Click **OK** to save the trigger.

Name	Group	Kind	Pr
DeleteArchivedEvents	primary_only	Temporal	1
accelerated_inserts	default_trigg...	Database	1
agg_deduplication	default_trigg...	Database	2
add new row	default tri...	Database	2

The trigger is shown in the list.

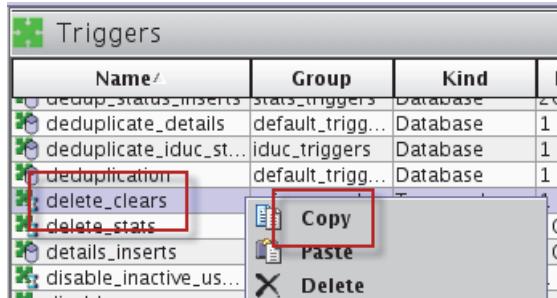
3. Create the reaper trigger.

As mentioned previously, there are advantages to creating a trigger that removes *old* events. The exact definition of what constitutes *old* events is subjective. Some users might feel that any

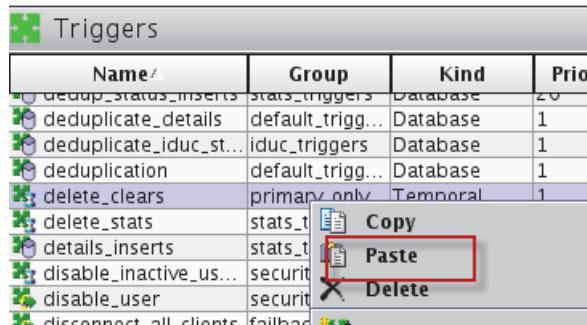
event more than 1 week old is no longer relevant. For other users, that time period might be longer or shorter. For this exercise, you use 1 week.

Start by copying an existing trigger definition.

- Click the existing trigger **delete_clears** to select it, right-click, and select **Copy**.



- Right-click and select **Paste**.



- Enter **Reaper** for the name.
- Leave Group set to **primary_only**.
- Change the frequency to **Every 24 hours**.
- Click the **Action** tab.



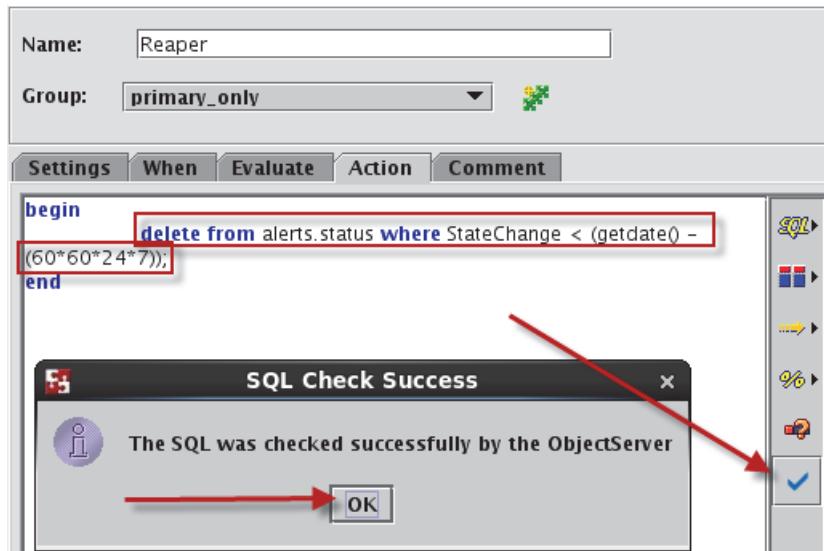
Important: It is important to configure temporal (time-based) triggers with conservative frequencies. Because this trigger is not critical to operations, it is not necessary to run it frequently.

You can now see the SQL from the delete_clears trigger. You must modify the text after the word **where** to suit your needs.

- g. Change the SQL as follows:

```
delete from alerts.status where StateChange < (getdate() - (60*60*24*7));
```

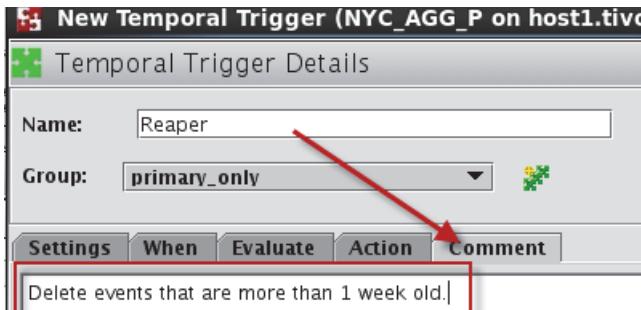
- h. Click the check mark to verify the syntax.



Note: The value $60*60*24*7$ equates to the number of seconds in 1 week.

- i. Click **OK**.

- j. Click the **Comment** tab, and enter a description for this trigger.



- k. Click **OK** to save the trigger.

Name	Group	Type
DeleteArchivedEvents	primary_only	Temporal
Reaper	primary_only	Temporal
accelerated_inserts	default_trigger	Data

The reaper trigger appears in the list of triggers.

4. Exit the administrator utility.

The following list is a summary of accomplishments for this unit:

- Created a tool to *flag* events for deletion. This tool is accessible to only members of the Operations group.
- Modified the existing delete tool to *hide* it from members of the Operations group.
- Created a trigger that removes events that are flagged for deletion, and are archived.
- Created the reaper trigger to remove old events.

The configuration to delete events is not complete. You created a trigger in this exercise to delete the events from the ObjectServer. However, one of the conditions that must be true in order for the trigger to remove the event is **Archived_Flag=1**. Nothing sets that value currently. The last requirement is configured in a subsequent unit.



7 Common integrations exercises

The exercises in this unit guide you through the installation, and configuration of two of the most popular Netcool/OMNIbus probes: the IBM® Tivoli® Netcool®/OMNIbus SNMP Probe, and the IBM Tivoli Netcool/OMNIbus Syslog Probe. In addition, you install the prebuilt collection of rules files for these probes. These files are the IBM Tivoli Netcool/OMNIbus Knowledge Library.

Exercise 1 Installing the Syslog probe

When installing the Syslog probe, be sure to configure the syslog daemon to record specific messages in a copy of the syslog file. The probe is configured to use the copy of the syslog file. You must configure the Syslog daemon as the **root** user.



Important: The following steps describe how to modify the Rsyslog daemon. These steps are specific to that particular daemon. In a production environment, the Syslog daemon might not be configured as shown.

Perform the following steps on the **host1** server:

1. Change to the **root** user:

```
su -  
Password: object00
```

2. Create the first-in, first-out (FIFO) file:

```
cd /var/log  
mkfifo netcool  
chown netcool:ncoadmin netcool
```



Hint: The name of the file is not relevant. The file name can be any valid UNIX name.

3. Modify the Rsyslog daemon as follows:

- a. Change to the required directory:

```
cd /etc
```

- b. Make a copy of the existing file:

```
cp rsyslog.conf rsyslog.conf.orig
```

- c. Modify the configuration file as follows:

```
gedit rsyslog.conf
```

Add the following lines to the end of the file:

```
# Forward debug messages for the Netcool Syslog Probe
*.debug /var/log/netcool
```

- d. Save the changes and exit the gedit utility.

- e. Restart the Rsyslog daemon as follows:

```
service rsyslog restart
```

Shutting down system logger:

[OK]

Starting system logger:

[OK]

- f. Verify that the changed configuration is now active:

```
tail /var/log/messages
```

```
Sep 24 20:11:28 host1 kernel: Kernel logging (proc) stopped.
```

```
Sep 24 20:11:28 host1 rsyslogd: [origin software="rsyslogd"
```

```
swVersion="5.8.10" x-pid="1210" x-info="http://www.rsyslog.com"] exiting on
signal 15.
```

```
Sep 24 20:11:29 host1 kernel: imklog 5.8.10, log source = /proc/kmsg started.
```

```
Sep 24 20:11:29 host1 rsyslogd: [origin software="rsyslogd"
```

```
swVersion="5.8.10" x-pid="4029" x-info="http://www.rsyslog.com"] start
```

4. Exit out of the **root** user back to the **netcool** user:

```
exit
```



Important: Make sure that you are the **netcool** user before proceeding.

5. Install the Syslog probe.

- a. Change to the correct directory:

```
cd /software/syslog
```

- b. Expand the installation file:

```
unzip im-nco-p-syslog-7_0.zip
```

- c. Run the IBM Installation Manager utility:

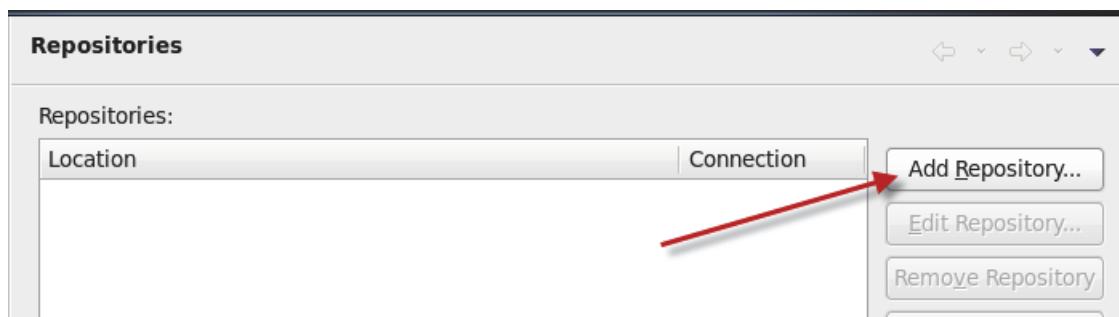
```
cd /home/netcool/IBM/InstallationManager/eclipse
./IBMMI
```

Define the repository location for the Syslog installation files.

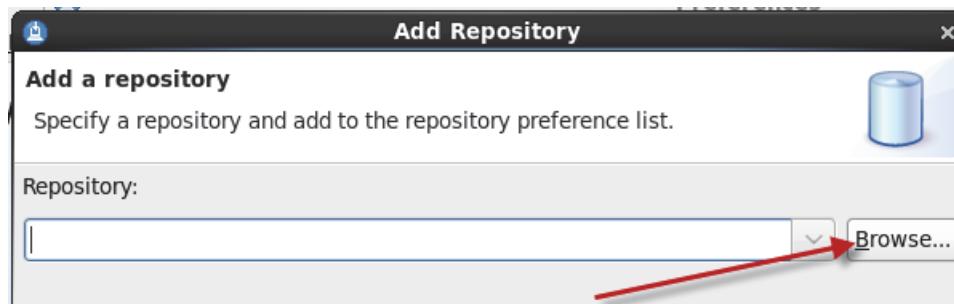
- d. Click **File** and select **Preferences**.



- e. Click **Add Repository**.

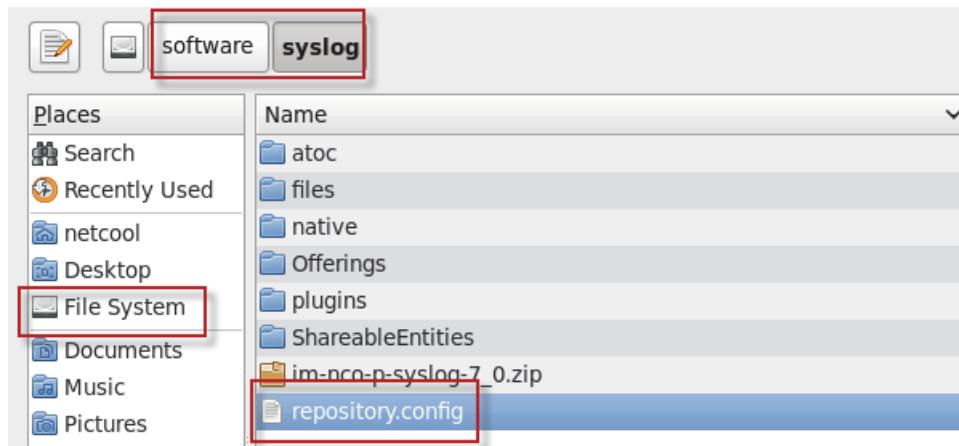


- f. Click **Browse**.

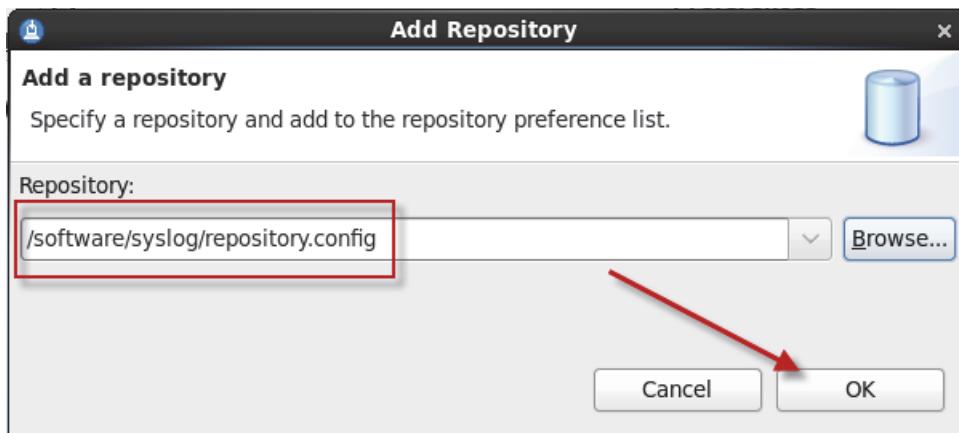


- g. Select **File System** and browse to **/software/syslog**.

- h. Click **repository.config** to select it and click **OK**.



- i. Verify that the correct file is selected and click **OK**.

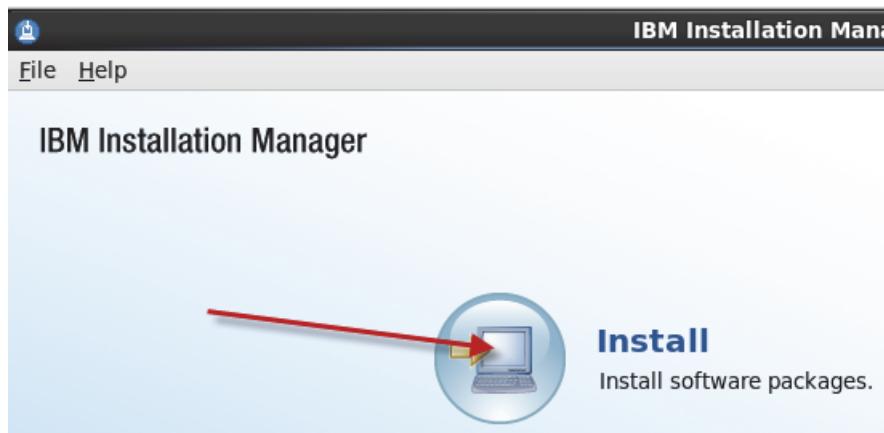


- j. Scroll to the bottom of the page and click **OK**.

Repositories	
Repositories:	
Location	Connection
<input checked="" type="checkbox"/> /software/syslog/repository.config	

6. Install the probe package.

- a. Click **Install**.



- b. Select the check box to select the probe package for installation.

- c. Scroll to the bottom of the page and click **Next**.

This screenshot shows the 'Install Packages' screen. The title 'Install Packages' is at the top, followed by the sub-instruction 'Select packages to install:'. Below is a table titled 'Installation Packages' with columns for 'Status' and 'Vendor'. Two rows are listed: 'Netcool/OMNibus Probe nco-p-syslog' (Status: Will be installed, Vendor: IBM) and 'Version 1.7.0.0' (Status: Will be installed, Vendor: IBM). Both rows have a checked checkbox in the first column, which is highlighted with a red box. A red arrow also points to the checkbox of the first row.

	Status	Vendor
<input checked="" type="checkbox"/> Netcool/OMNibus Probe nco-p-syslog	Will be installed	IBM
<input checked="" type="checkbox"/> Version 1.7.0.0	Will be installed	IBM

- d. Accept the license agreement and click **Next**.

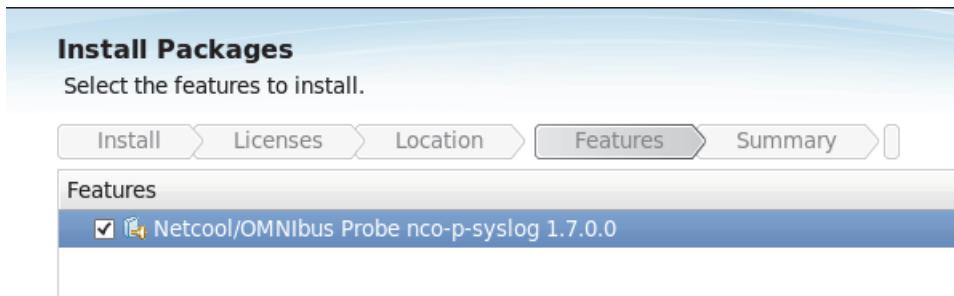
I accept the terms in the license agreement
 I do not accept the terms in the license agreement

- e. Keep the selection to use the existing package group and click **Next**.

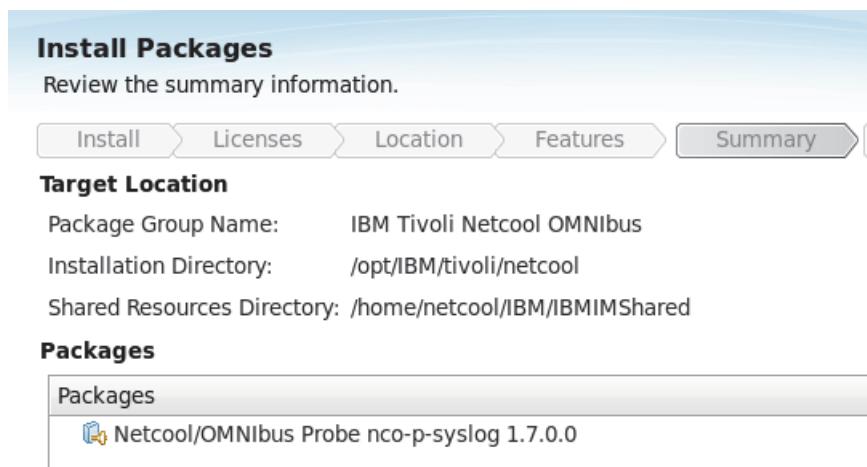
This screenshot shows the 'Install Packages' screen again. The title 'Install Packages' is at the top, followed by a descriptive text about package groups. Below is a navigation bar with tabs: 'Install', 'Licenses', 'Location', 'Features', 'Summary'. The 'Install' tab is active. Underneath, there are two radio button options: 'Use the existing package group' (selected) and 'Create a new package group'. Below this is a table titled 'Package Group Name' with columns for 'Installation Directory'. One row is listed: 'IBM Tivoli Netcool OMNIBus' (Installation Directory: /opt/IBM/tivoli/netcool).

	Installation Directory
IBM Tivoli Netcool OMNIBus	/opt/IBM/tivoli/netcool

f. Click **Next**.



g. Review the installation summary and click **Install**.



h. Verify that the installation is successful and click **Finish**.



The packages are installed. [View Log File](#)

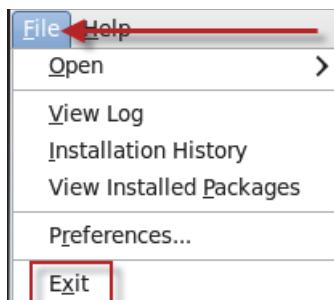
The following package was installed:

- ▽ IBM Tivoli Netcool OMNibus
 - Netcool/OMNibus Probe nco-p-syslog 1.7.0.0



Hint: Click [View Log File](#) to review the installation log.

i. Click **File** and select **Exit**.



7. Configure the probe.

The following step configures the probe to perform the following tasks:

- Use the default rules file
- Use the pipe file from the Rsyslog daemon
- Enable remote access to the probe with OSLC (Open Source Lifecycle Collaboration)

- a. Change to the required directory:

```
cd $OMNIHOME/probes/linux2x86
```

- b. Save a copy of the original property file:

```
cp syslog.props syslog.props.orig
```

- c. Modify the property file:

```
gedit syslog.props
```

- d. Add the following lines to the *bottom* of the file:

```
Server : "NYC_AGG_P"  
FifoName : "/var/log/netcool"  
RulesFile :  
"$OMNIHOME/probes/linux2x86/syslog.rules"  
  
NHttpd.EnableHTTP : TRUE  
NHttpd.ListeningHostname : "host1.tivoli.edu"  
NHttpd.ListeningPort : 4199
```



Hint: When modifying a probe property file, place all modifications at the end of the file. This practice makes it easier to locate all the changes.

Save the modified property file and exit the gedit utility.

- e. Start the probe as follows:

```
$OMNIHOME/probes/nco_p_syslog &
```

- f. Check the log file for errors:

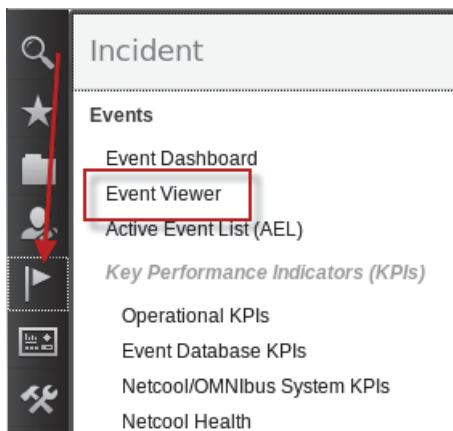
```
cd $OMNIHOME/log
```

```
more syslog.log
```

8. Test the probe.

The easiest way to test the probe is to insert a known message into the syslog file, and verify that the probe sends the correct event to the ObjectServer.

- Log in to Dashboard Application Services Hub as the **ncoadmin** user.
- Open the **Event Viewer**.



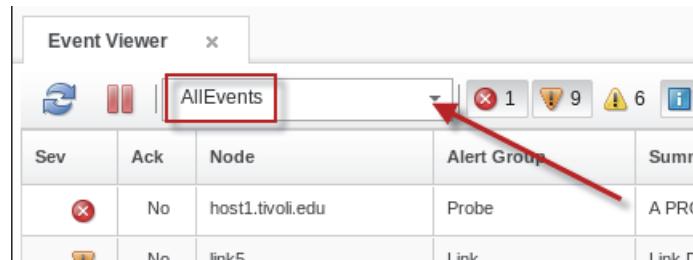
Leave the Event Viewer open. You use it again shortly.

- Enter the following text in a terminal window to create a syslog entry:

```
logger This is a test
```

Return to the Event Viewer window and examine the event records. The event that is generated from the test message has a severity of unknown.

- Click the arrow and select **AllEvents**.



Scroll down and locate the test event.

↶	No	host1	run-parts(/etc/cron.hourly)	starting mcelog.cron
↶	No	host1	run-parts(/etc/cron.hourly)	finished mcelog.cron
↶	No	host1	netcool:	This is a test
↶	No	host1	rsyslogd:	[origin software rsyslogd swVersion 5.8.10 x-pi]

- Log out of Dashboard Application Services Hub.

- f. Close the Firefox browser.
 - ◆ The Rsyslog daemon is directing messages to **/var/log/netcool**.
 - ◆ The Syslog probe is reading entries from **/var/log/netcool**.
 - ◆ The Syslog probe is parsing the message with the default rules file.
 - ◆ The Syslog probe is generating the correct event, and forwarding that event to the correct ObjectServer.
9. Verify access to the Syslog probe with OSLC:
Netcool/OMNibus provides several command-line tools that can communicate with a probe with OSLC. One of those tools requests a list of the active probe properties.
 - a. Run the following command:

```
nco_http -uri http://host1.tivoli.edu:4199/probe/common
```
 - b. Examine this list, and verify that the following values appear:

```
"Server":"NYC_AGG_P",  
"FifoName":"/var/log/netcool",
```

The correct property values verify that you can communicate with the Syslog probe with OSLC.



Note: In this example, the probe is running on the same server where the command is run. In a production environment, probes typically run on remote servers.

Exercise 2 Installing the SNMP probe

SNMP traps are usually delivered over port 162. On UNIX or Linux operating systems, any process that listens to a port with a value of 1024 or smaller, must run as the **root** user. The following exercise demonstrates how to install the SNMP probe, configure the probe to use port 162, and run as a non-root user.

1. Install the SNMP probe.
 - a. Change to the correct directory:

```
cd /software/snmp
```
 - b. Expand the installation file:

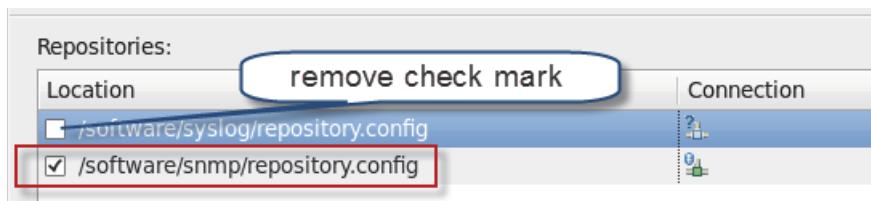
```
unzip im-nco-p-mttrapd-18_0.zip
```

- c. Run the IBM Installation Manager utility:

```
cd /home/netcool/IBM/InstallationManager/eclipse
./IBMMIM
```

Follow same steps that were used previously to define the repository location for the SNMP installation files.

- d. Remove the check mark for the Syslog repository.

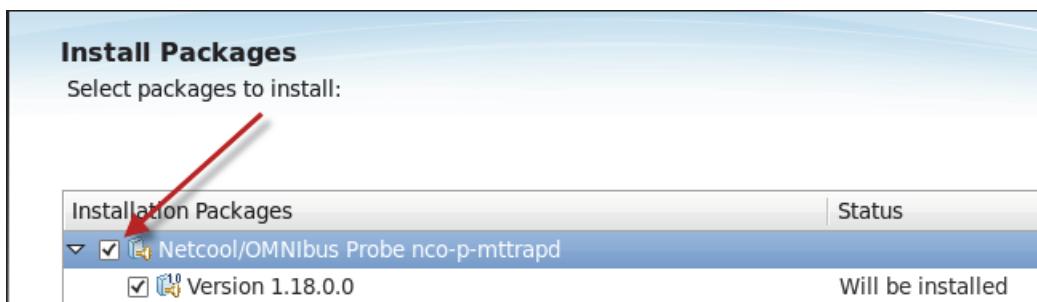


Hint: You can define multiple repositories, run the installation manager utility one time, and install multiple probes in a single execution.

- e. Click **OK**.

- f. Click **Install**.

- g. Select the check box to install the SNMP probe package and click **Next**.



Note: The correct product name is nco-p-mttrapd.

- h. Repeat the steps that are used previously to install the Syslog probe and complete the installation of the SNMP probe.

- i. Verify that the installation is successful and click **Finish**.



The packages are installed. [View Log File](#)

The following package was installed:

▼	IBM Tivoli Netcool OMNIbus
	Netcool/OMNibus Probe nco-p-mttrapd 1.18.0.0

- j. Exit the IBM Installation Manager utility.

2. Configure the probe.

The following step configures the probe to perform these tasks:

- Use the default rules file
- Use the default SNMP trap port number (162)
- Enable remote access to the probe with OSLC (Open Source Lifecycle Collaboration)

- a. Change to the required directory:

```
cd $OMNIHOME/probes/linux2x86
```

- b. Save a copy of the original property file:

```
cp mttrapd.props mttrapd.props.orig
```

- c. Modify the property file:

```
gedit mttrapd.props
```

- d. Add the following lines to the *bottom* of the file:

```
Server : "NYC_AGG_P"  
RulesFile :  
"$OMNIHOME/probes/linux2x86/mttrapd.rules"  
  
NHttpd.EnableHTTP : TRUE  
NHttpd.ListeningHostname : "host1.tivoli.edu"  
NHttpd.ListeningPort : 4198
```



Hint: When modifying a probe property file, place all modifications at the end of the file. This practice makes it easier to locate all the changes.

- e. Save the modified property file and exit the gedit utility.

3. Configure the probe to run as a non-root user.

If using the default port number for SNMP traps, 162, you must run the probe as the **root** user. If you do not want to run the probe as the **root** user, you must either use a different port number for traps, something greater than 1024 or configure the probe to run with setuid. The following steps describe how to configure the probe to run with setuid.

- a. Change to the **root** user:

```
su -  
Password: object00
```

- b. Change the ownership of the probe binary file:

```
cd /opt/IBM/tivoli/netcool/omnibus/platform/linux2x86/probes64
```

```
chown root nco_p_mttrapd
```

- c. Enable the probe to run as setuid root:

```
chmod +s nco_p_mttrapd
```

- d. Modify the dynamic linker runtime bindings file:

```
cd /etc
```

```
cp ld.so.conf ld.so.conf.orig
```

```
gedit ld.so.conf
```

Add the following lines to the bottom of the file:

```
/opt/IBM/tivoli/netcool/platform/linux2x86/lib64
```

```
/opt/IBM/tivoli/netcool/omnibus/platform/linux2x86/lib64
```

- e. Save the changes and exit the gedit utility.

- f. Reload the modified file:

```
ldconfig -v
```

- g. Exit the *root* user back to the *netcool* user:

```
exit
```



Important: Make sure that you are the *netcool* user before proceeding.

4. Start the probe as the **netcool** user.

- a. Start the probe:

```
nco_p_mttrapd &
```

- b. Check the log file for errors:

```
cd $OMNIHOME/log
```

```
more mttrapd.log
```

5. Test the probe.

The easiest way to test the probe is to generate a specific SNMP trap and verify that the probe sends the correct event to the ObjectServer.

- a. Log in to Dashboard Application Services Hub as the **ncoadmin** user.

- b. Open the Event Viewer.

Leave the Event Viewer window open.

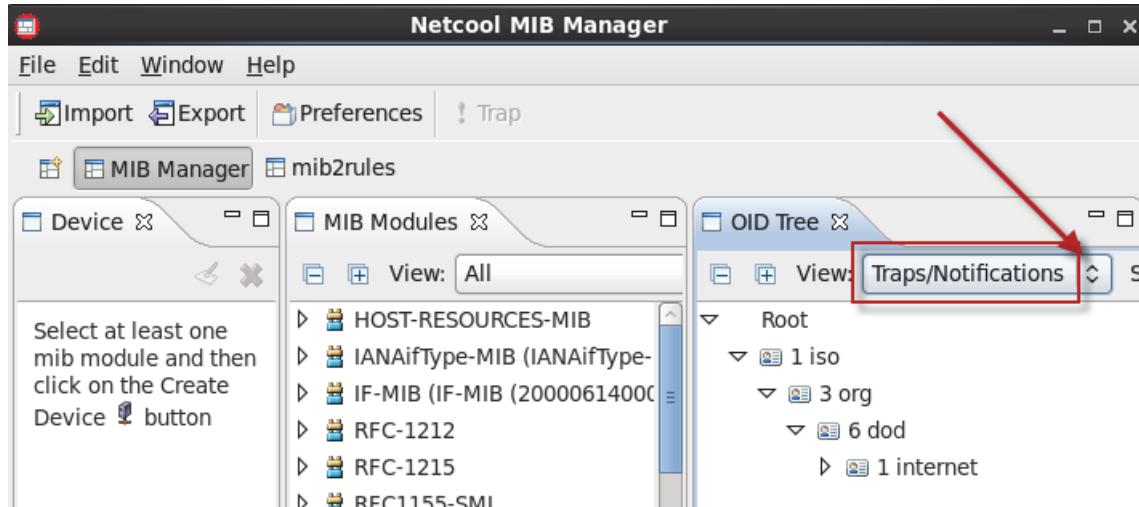
The MIB Manager utility is bundled with Netcool/OMNibus. One of the features is the ability to generate SNMP trap messages. You use the MIB Manager utility to generate a coldStart trap.

6. Generate a coldStart SNMP trap as follows:

- a. Start the MIB Manager utility.

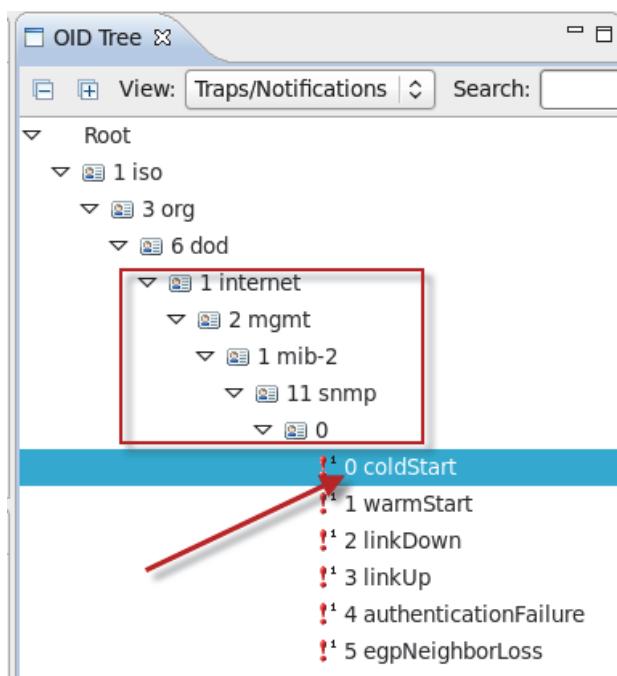
```
nco_mibmanager &
```

- b. Change the view to **Traps/Notifications**.

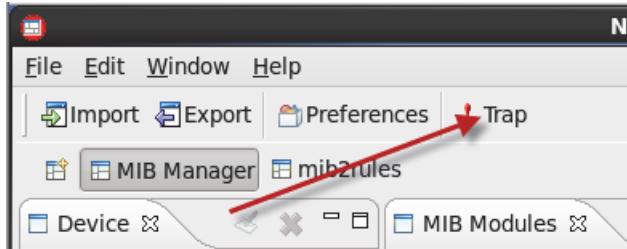


- c. Expand the entries in the OID Tree until the trap messages open.

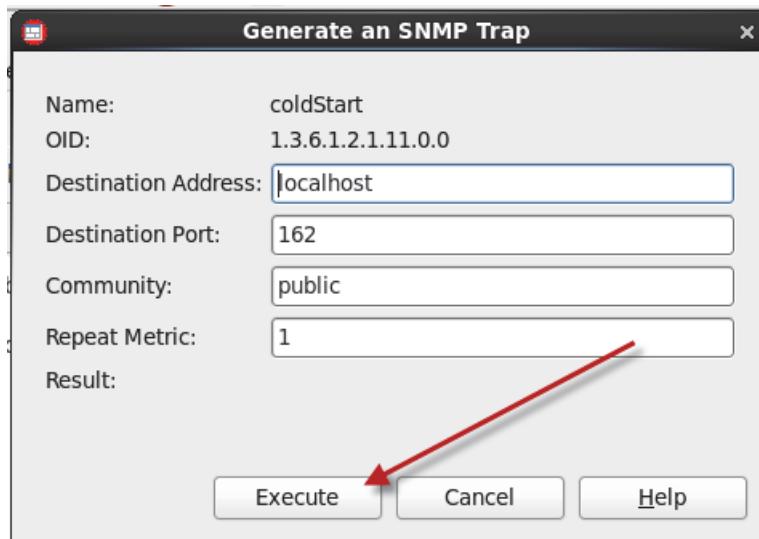
- d. Click **coldStart** to select it.



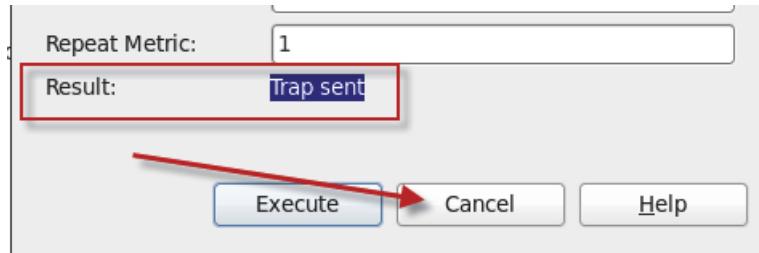
e. Click Trap.



f. Leave all of the default settings and click Execute.



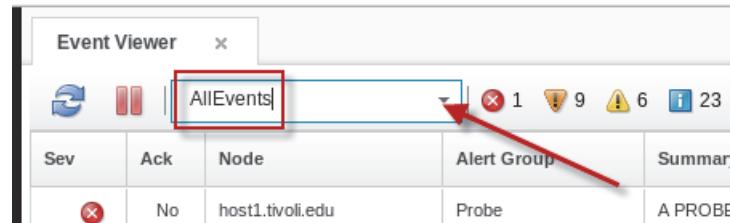
g. Verify that the trap is sent and click Cancel.



h. Exit the MIB Manager utility.

i. Return to the Event Viewer window.

j. Click the arrow and select AllEvents.



- k. Scroll down and locate the Cold Start event.

	No	Moscow	Systems	Machine has gone offline
	No	Tokyo	Stats	Diskspace alert
	No	127.0.0.1	Generic	Cold Start
	No	Tokyo	TopNodes	ALERT: last 5 mins: high number of events seen

7. Log out of Dashboard Application Services Hub.

8. Close the Firefox browser.

The SNMP probe generates the event from the SNMP trap. The following items are verified:

- The SNMP probe is receiving traps on port 162 while running as the *netcool* user
- The SNMP probe is parsing the trap with the default rules file
- The SNMP probe is generating the correct event, and forwarding it to the correct ObjectServer

9. Verify access to the SNMP probe with OSLC:

Netcool/OMNIbus provides several command-line tools to communicate with a probe with OSLC. One of those tools requests a list of the active probe properties.

- a. Run the following command:

```
nco_http -uri http://host1.tivoli.edu:4198/probe/common
```

It produces a list of the active probe property values. Examine this list, and verify that the following value is shown:

```
"Name":"mttrapd",
```

The correct property values verify that you are able to communicate with the SNMP probe with OSLC.

 **Note:** If you do not receive the correct list of property values, verify that you used port 4198 in the `nco_http` command. The Syslog probe uses port 4199.

Exercise 3 Installing the Netcool Knowledge Library

The Netcool Knowledge Library is a collection of prebuilt probe rules files for the SNMP and Syslog probes. These files are delivered as a compressed file.

1. Install the Knowledge Library.

- a. Change to the correct directory:

```
cd /software/nckl
```

- b. Expand the installation file:

```
gunzip NcKL4.2.tar.gz  
tar -xvf NcKL4.2.tar
```

- c. Apply modifications to the primary ObjectServer:

```
cd NcKL
```

```
nco_sql -server NYC_AGG_P -user root -password object00 < advcorr.sql
```

 Note: Ignore the error messages that relate to missing objects. These messages are normal.

- d. Expand the compressed Knowledge Library files:

```
gunzip rules.tar.gz
```

- e. Expand the Knowledge files into the target directory:

```
cd $NCHOME
```

```
tar -xvf /software/nckl/NcKL/rules.tar
```

This action creates a subdirectory that is called *rules*, which contain the Knowledge Library files.

- f. Update the environment variable settings:

```
cd /home/netcool
```

```
gedit .bashrc
```

Add the following lines to the bottom of the file:

```
# Netcool Knowledge Library  
NC_RULES_HOME=$NCHOME/rules  
export NC_RULES_HOME
```

g. Save the changes and exit the gedit utility.

h. Source the updated values:

```
source .bashrc
```

i. Verify the environment variable:

```
cd $NC_RULES_HOME
```

If the environment variable is correct, the change directory command takes you to:

```
pwd
```

```
/opt/IBM/tivoli/netcool/rules
```

2. Configure the Syslog probe to use the Knowledge Library.

a. Change to the required directory:

```
cd $OMNIHOME/probes/linux2x86
```

b. Modify the probe property file:

```
gedit syslog.props
```

Modify the *RulesFile* property as follows:

```
RulesFile : "$NC_RULES_HOME/syslog.rules"
```

c. Save the changes and exit the gedit utility.

d. Stop the running probe.

```
pkill nco_p_syslog
```

e. Restart the probe:

```
nco_p_syslog &
```

f. Check the log file for errors:

```
cd $OMNIHOME/log
```

```
more syslog.log
```



Hint: The warning messages are normal and you can ignore them.

The Netcool Knowledge Library contains SNMP probe rules files that support SNMP traps from numerous vendors. The vendor-specific rules are contained within vendor-specific files. The files are incorporated into a comprehensive rules file through the use of INCLUDE statements. In the default distribution of these files, the include statements for all supported vendors are contained within a single master rules file. However, most of the INCLUDE statements are commented out. To enable support for a specific vendor, you must modify the master rules file. The following steps demonstrate how to modify the file to support SNMP traps for Cisco devices. The same technique is used to enable support for other vendors.

3. Configure the Knowledge Library rules for Cisco devices.

- a. Change to the required directory:

```
cd $NC_RULES_HOME
```

- b. Save a copy of the original file.

```
cp snmptrap.rules snmptrap.rules.orig
```

- c. Modify the master rules file:

```
chmod +w snmptrap.rules
```

```
gedit snmptrap.rules
```

The master rules file contains two types of INCLUDE statements. The first type of include statements is for lookup tables. These INCLUDE statements appear at the top of the file. The second type of include statement is for rules files. These INCLUDE statements occur further down in the master rules file. There are typically both types of INCLUDE statements for each supported vendor.

- d. Locate the include statement for the Cisco lookup file.

```
#include "$NC_RULES_HOME/include-snmptrap/cisco/cisco.master.include.lookup"
```

- e. Remove the # character from the beginning of the line.

```
include "$NC_RULES_HOME/include-snmptrap/cisco/cisco.master.include.lookup"
```



Note: The Cisco rules files also refer to tables that are contained in another lookup file. You must also uncomment the INCLUDE statement for that file.

- f. Locate the INCLUDE statement for the IEEE lookup file.

```
#include "$NC_RULES_HOME/include-snmptrap/IEEE/IEEE.master.include.lookup"
```

- g. Remove the # character from the beginning of the line.

```
include "$NC_RULES_HOME/include-snmptrap/IEEE/IEEE.master.include.lookup"
```

- h. Locate the INCLUDE statement for the **first** Cisco rules file.

```
#include "$NC_RULES_HOME/include-snmptrap/cisco/cisco.master.include.rules"
```

- i. Remove the # character from the beginning of the line.

```
include "$NC_RULES_HOME/include-snmptrap/cisco/cisco.master.include.rules"
```

- j. Locate the INCLUDE statement for the **second** Cisco rules file.

```
#include
```

```
"$NC_RULES_HOME/include-snmptrap/generic/cisco.generic.include.snmptrap.rules"
```

- k. Remove the # character from the beginning of the line.

```
include  
"${NC_RULES_HOME}/include-snmptrap/generic/cisco.generic.include.snmptrap.rules  
s"
```

- l. Locate the INCLUDE statement for the **third** Cisco rules file.

```
include  
"${NC_RULES_HOME}/include-snmptrap/cisco/cisco-preclass.include.snmptrap.rules  
"
```

- m. Remove the # character from the beginning of the line.

```
include  
"${NC_RULES_HOME}/include-snmptrap/cisco/cisco-preclass.include.snmptrap.rules  
"
```

- n. Save the changes and exit the gedit utility.

4. Verify the syntax of the modified master rules file.

```
nco_p_syntax -server NYC_AGG_P -rulesfile snmptrap.rules
```

The syntax check produces numerous output lines. After the command completes, verify that no rules appear in the master rules file by locating this line in the output:

```
2014-09-25T15:45:13: Information: I-UNK-000-000: Rules file syntax OK
```



Important: The required elements that are contained within the various files vary from vendor to vendor. There is no precise method for determining which files are required for each vendor. The process is typically to uncomment the obvious file references and then test the syntax. If errors are found, locate and uncomment the additional files.

5. Configure the SNMP probe to use the Knowledge Library.

- a. Change to the required directory:

```
cd $OMNIHOME/probes/linux2x86
```

- b. Modify the probe property file:

```
gedit mttrapd.props
```

Modify the RulesFile property as follows:

```
RulesFile : "${NC_RULES_HOME}/snmptrap.rules"
```

- c. Save the changes and exit the gedit utility.

- d. Stop the running probe.

```
pkill nco_p_mttrapd
```

- e. Restart the probe:

```
nco_p_mttrapd &
```

- f. Check the log file for errors:

```
cd $OMNIHOME/log
```

```
more mttrapd.log
```



Hint: The warning messages are normal and you can ignore them. You can also ignore the error messages about the **mttrapd.conf** file.

The summary of accomplishments for this unit includes these tasks:

- Install Syslog probe.
- Install SNMP probe.
- Configure SNMP probe to run as a non-root user.
- Install Netcool Knowledge Library.
- Configure both probes for OSLC access.
- Configure both probes to use the Netcool Knowledge Library rules.



8 ObjectServer high availability exercises

The exercises in this unit guide you through the implementation of ObjectServer high availability.

Exercise 1 Creating the backup ObjectServer

There are three aspects to implementing ObjectServer high availability. First, you must create the backup ObjectServer. Next, you must ensure that the backup ObjectServer is configured like the primary ObjectServer. And, last, you implement an ObjectServer gateway that replicates changes between the primary and backup ObjectServers.

You created the primary ObjectServer in a previous unit when you installed Netcool®/OMNibus core on the **host1**. The Initial Configuration Wizard created the ObjectServer. In this unit, you implement high availability by installing Netcool/OMNibus core on the **host2** image.



Important: Make sure that you are the **netcool** user on the **host2** image before proceeding.

1. Expand the core installation file as follows:

```
cd /software/omnibus
```

```
unzip OMNIBus-v8.1-Core.linux64.zip
```

2. Install the core software:

When you installed the core components on the host1 image previously, you also had to install the IBM® Installation Manager utility. The IBM Installation Manager utility is already installed on the host2 image. To install the core components on host2, you run the existing copy of IBM Installation Manager and define the Netcool/OMNibus installation files as a repository.

- a. Run the IBM Installation Manager utility:

```
cd /home/netcool/IBM/InstallationManager/eclipse  
./IBMIM
```

- b. Define the repository location for the Netcool/OMNIbus core installation files.

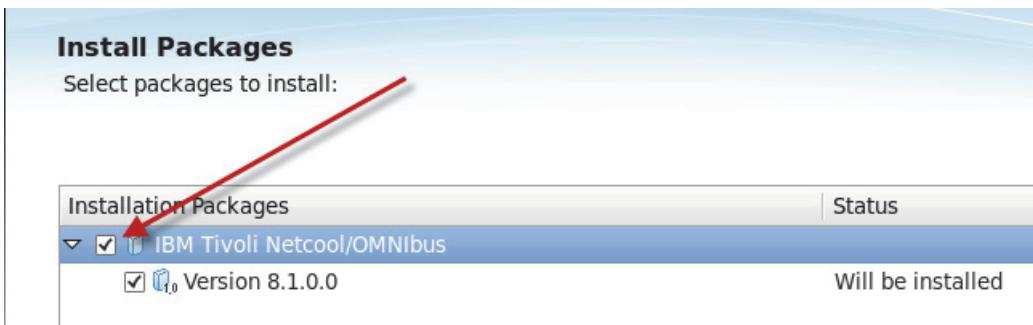


- c. Remove the check marks for the other repositories and click **OK**.

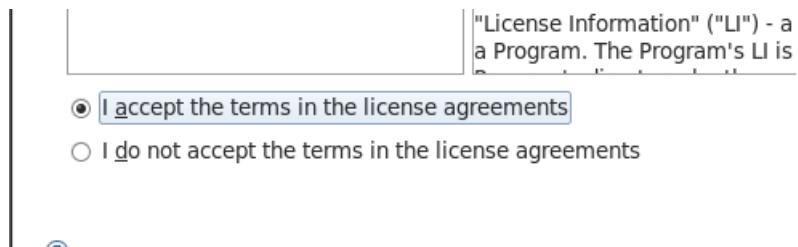


- d. Click **Install**.

- e. Check the box to select the Netcool/OMNIbus package and click **Next**.



- f. Accept the license agreement and click **Next**.



- g. Leave the default value set to create a new package group and click **Next**.

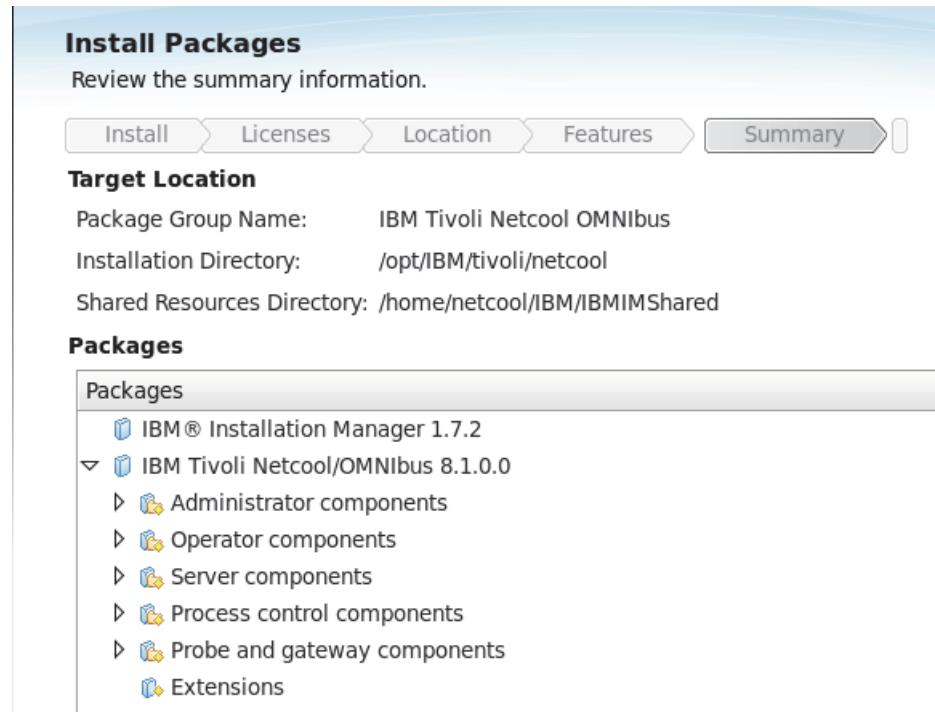
- h. Leave all of the features selected and click **Next**.

- i. Leave the option for Data migration cleared and click **Next**.



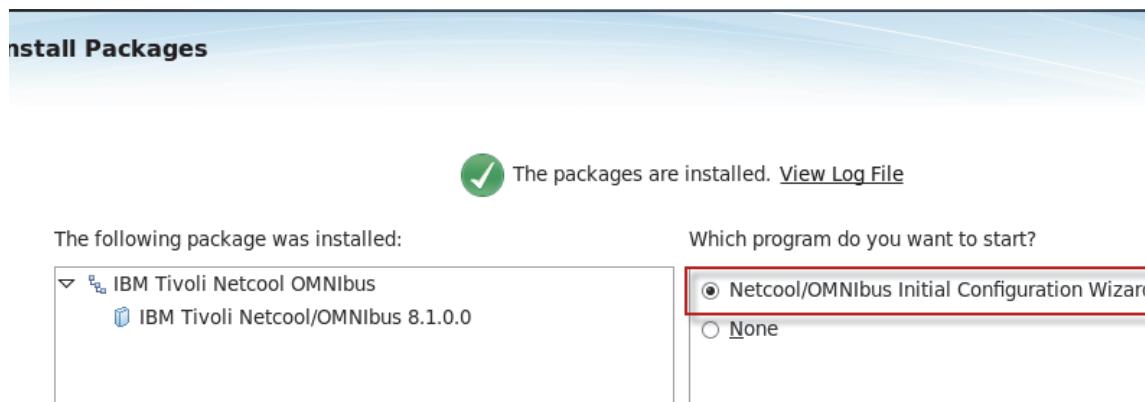
Hint: The option is used when upgrading from a previous version of Netcool/OMNibus.

- j. Review the installation summary and click **Install**.



Hint: An installation on most servers runs approximately 10 minutes.

- k. Leave the option selected to run the configuration wizard and click **Finish**.



Verify that the installation is successful.

- I. Click **Finish** to exit the installation wizard.
- m. Click **Exit** to close the installation manager utility.

Initial Configuration wizard

At the conclusion of the installation process, the configuration wizard starts automatically.



Hint: The configuration wizard is started manually by using the following command:

`/opt/IBM/tivoli/netcool/omnibus/bin/nco_icw`

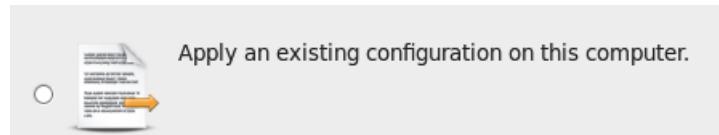
1. Complete the configuration with the wizard as follows:

- a. Scroll to the bottom of the view and click **Next**.



- b. Leave the option selected to create a new configuration and click **Next**.

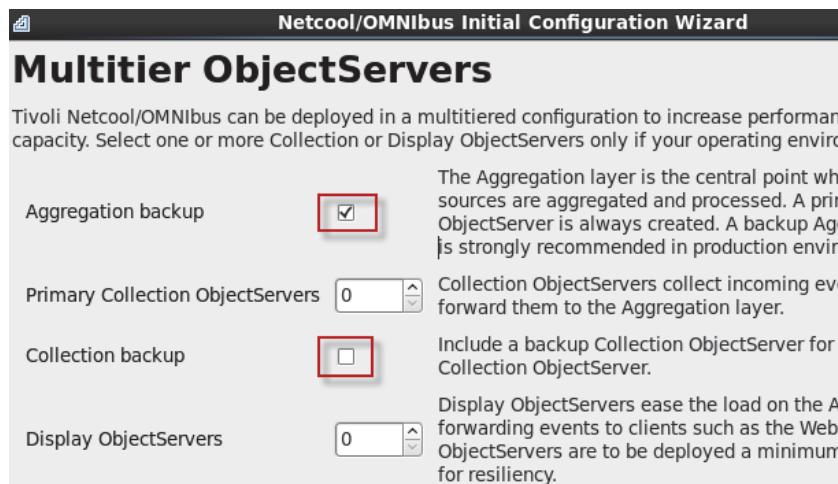
The wizard was already run during a previous exercise. You can retrieve a copy of the configuration file from the **host1** image and save it on the **host2** image. If you do that, you can select the option to apply an existing configuration:



However, for this exercise you create a new configuration file.

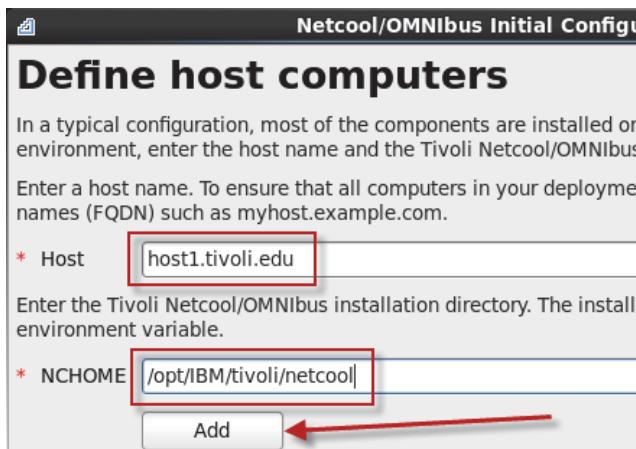
- c. Leave the check for **Aggregation backup** and remove the check for **Collection backup**.

- d. Click **Next**.



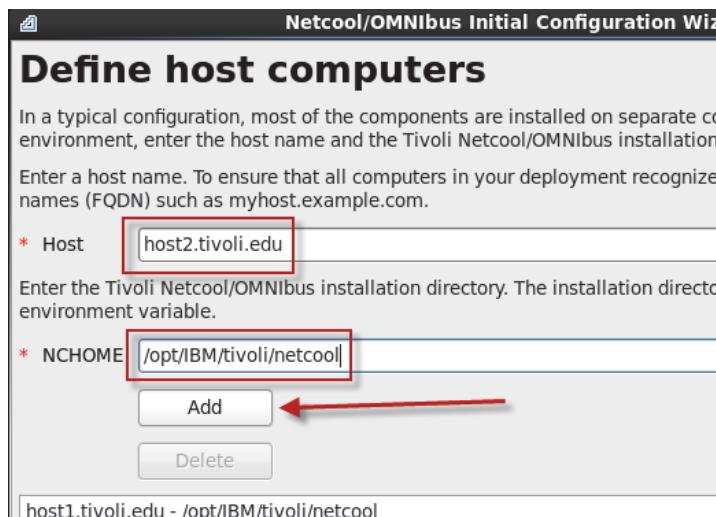
- e. Enter **host1.tivoli.edu** and **/opt/IBM/tivoli/netcool**.

- f. Click **Add**.



- g. Enter **host2.tivoli.edu** and **/opt/IBM/tivoli/netcool**.

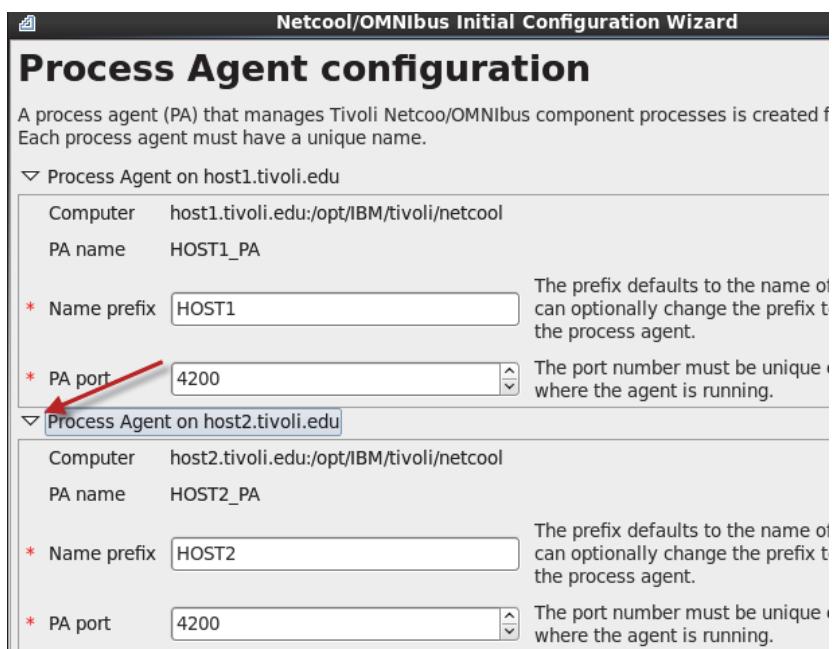
- h. Click **Add**.



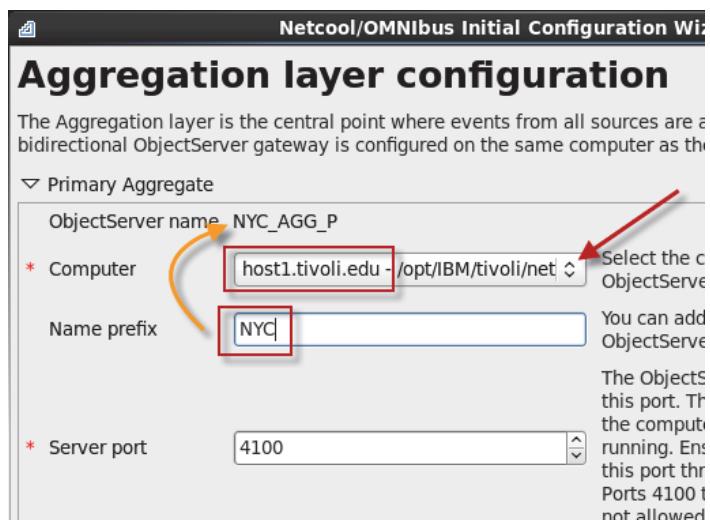
- i. Verify that both entries are listed as shown and click **Next**.



- j. Verify that the settings for process agents are listed as shown and click **Next**.

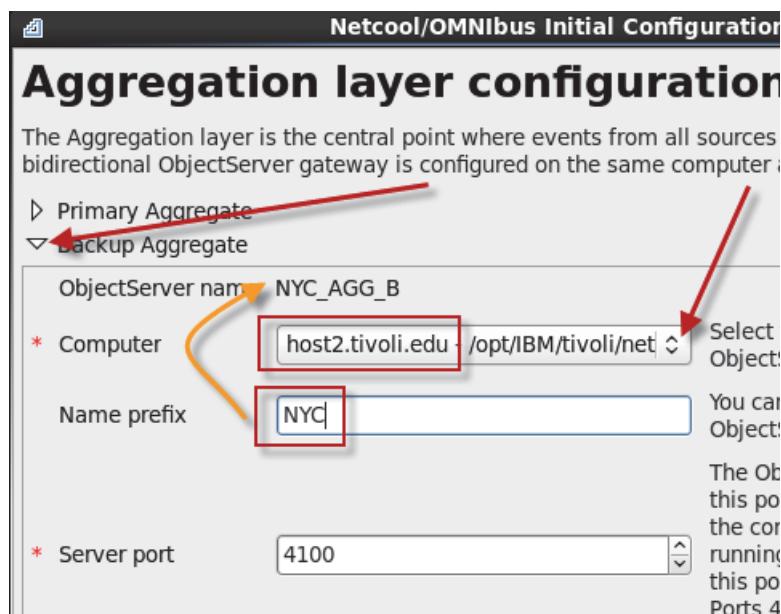


- k. Select **host1.tivoli.edu** for the computer and enter **NYC** for the name prefix.



The primary ObjectServer name is set to AGG_P and cannot be changed. You can enter text in the **Name prefix** field, and that text adds a prefix to AGG_P.

- I. Expand the option for **Backup Aggregate**.
- m. Select **host2.tivoli.edu** for the computer and enter **NYC** for the name prefix.



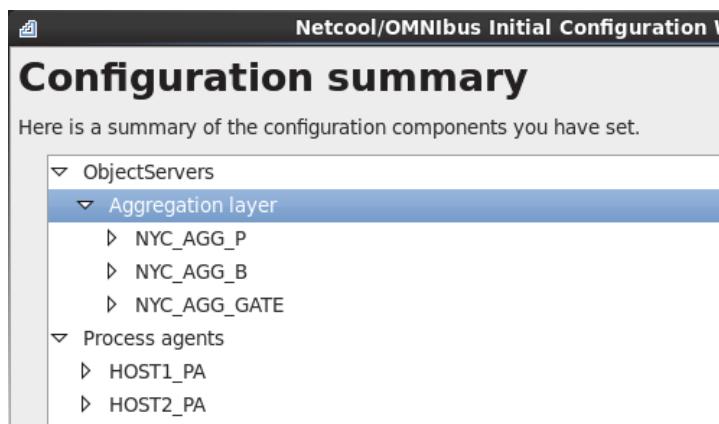
The backup ObjectServer name is set to AGG_B and cannot be changed. You can enter text in the **Name prefix** field, and that text adds a prefix to AGG_B.

- n. Scroll down in the view, enter NYC for the gateway prefix, and click **Next**.

Gateway name	NYC_AGG_GATE
Gateway prefix	NYC
* Gateway port	4300

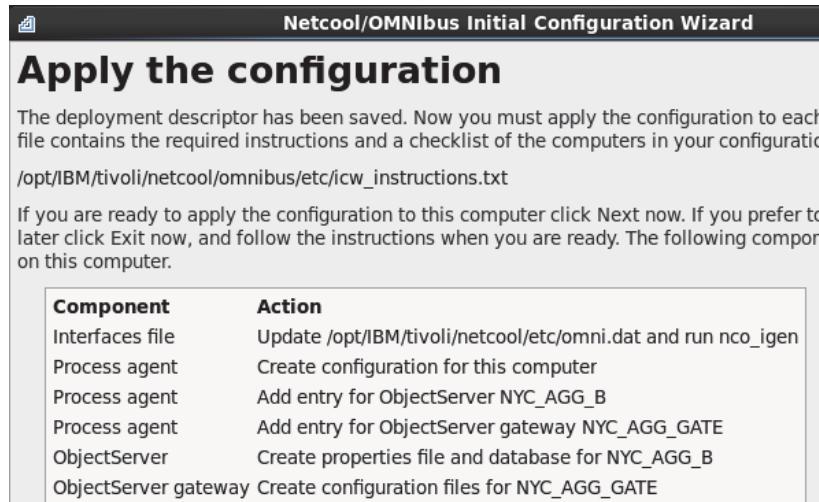
The gateway name is set to AGG_GATE and cannot be changed. You can enter text in the **Gateway prefix** field, and that text adds a prefix to AGG_GATE.

- o. Review the configuration summary and click **Next**.

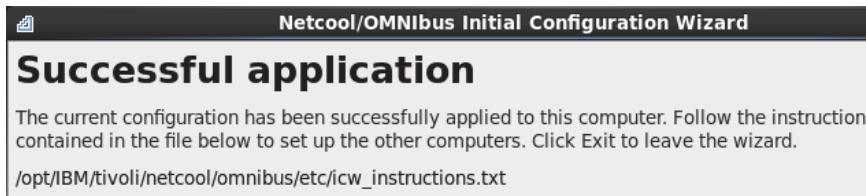


- p. Click **Next**.

- q. Click **Next** to apply the configuration to **host2**.



- r. Verify that the configuration is successfully applied and click **Exit**.



The configuration is applied to the host2 system.

Verifying the initial configuration

The initial configuration wizard performs a number of steps that can be performed manually.

1. Examine the communications file as follows:

```
cd /opt/IBM/tivoli/netcool/etc
more omni.dat
```

```
[netcool@host1 etc]$ more omni.dat
[NYC_AGG_P]
{
    Primary: host1.tivoli.edu 4100
}
[NYC_AGG_B]
{
    Primary: host2.tivoli.edu 4100
}
[AGG_V]
{
    Primary: host1.tivoli.edu 4100
    Backup: host2.tivoli.edu 4100
}
[NYC_AGG_GATE]
{
    Primary: host2.tivoli.edu 4300
}
[HOST1_PA]
{
    Primary: host1.tivoli.edu 4200
}
[HOST2_PA]
{
    Primary: host2.tivoli.edu 4200
}
```

primary ObjectServer

backup ObjectServer

virtual ObjectServer

synchronizer gateway

host1 Process Agent

host2 Process Agent

The **omni.dat** file contains the entries for the primary ObjectServer (NYC_AGG_P), the backup ObjectServer (NYC_AGG_B), and the *virtual* ObjectServer (AGG_V). The file also contains the entry for the synchronizer gateway (NYC_AGG_GATE) and the two process agents (HOST1_PA and HOST2_PA).

2. Examine the interfaces file as follows:

```
more interfaces.linux2x86

#
# interfaces file generated by nco_igen on Tue Jun 10 17:18:47 2014
#
#
# DO NOT AMEND THIS FILE !!!!!!!
# MAKE CHANGES IN THE $NCHOME/etc/omni.dat FILE AND
# RE-GENERATE THIS FILE WITH NCO_IGEN
#
#
# NYC_AGG_P => host1.tivoli.edu 4100
NYC_AGG_P
    master tcp sun-ether host1.tivoli.edu 4100
    query tcp sun-ether host1.tivoli.edu 4100

# NYC_AGG_B => host2.tivoli.edu 4100
NYC_AGG_B
    master tcp sun-ether host2.tivoli.edu 4100
    query tcp sun-ether host2.tivoli.edu 4100

# AGG_V => host1.tivoli.edu 4100
AGG_V
    master tcp sun-ether host1.tivoli.edu 4100
    query tcp sun-ether host1.tivoli.edu 4100
    master tcp sun-ether host2.tivoli.edu 4100
    query tcp sun-ether host2.tivoli.edu 4100

# NYC_AGG_GATE => host2.tivoli.edu 4300
NYC_AGG_GATE
    master tcp sun-ether host2.tivoli.edu 4300
    query tcp sun-ether host2.tivoli.edu 4300

# HOST1_PA => host1.tivoli.edu 4200
HOST1_PA
    master tcp sun-ether host1.tivoli.edu 4200
    query tcp sun-ether host1.tivoli.edu 4200

# HOST2_PA => host2.tivoli.edu 4200
```

```
HOST2_PA
  master tcp sun-ether host2.tivoli.edu 4200
  query tcp sun-ether host2.tivoli.edu 4200
```

```
#
# EOF
#
```

The interfaces file is created from the **omni.dat** file by running the **nco_igen** utility. The configuration wizard runs the utility.

3. Verify that the process agent is configured as follows:

- a. Examine the process agent configuration file

```
cd /opt/IBM/tivoli/netcool/omnibus/etc
more nco_pa.conf
```

```
#NCO_PA3
#
# Process Agent Daemon Configuration File 1.1
#
#
#
# List of processes
#
nco_process 'BackupObjectServer'
{
  Command '$OMNIHOME/bin/nco_objserv -name NYC_AGG_B -pa HOST2_PA' run as 0
  Host = 'host2.tivoli.edu'
  Managed= True
  RestartMsg='${NAME} running as ${EUID} has been restored on ${HOST}.'
  AlertMsg='${NAME} running as ${EUID} has died on ${HOST}.'
  RetryCount=0
  ProcessType=PaPA_AWARE
}

nco_process 'BackupGateway'
{
  Command '$OMNIHOME/bin/nco_g_objserv_bi -propsfile $NCHOME/omnibus/etc/NYC_AGG_GATE.props' run as 0
  Host = 'host2.tivoli.edu'
  Managed= True
```

```
    RestartMsg='${NAME} running as ${EUID} has been restored on
${HOST}.'
    AlertMsg='${NAME} running as ${EUID} has died on ${HOST}.

'
    RetryCount=0
    ProcessType=PaPA_AWARE
}

#
# List of Services
#
# NOTE: To ensure that the service is started automatically, change the
#        "ServiceStart" attribute to "Auto".
#
nco_service 'Core'
{
    ServiceType=Master
    ServiceStart=Auto
    process 'BackupObjectServer' NONE
    process 'BackupGateway' NONE
}

#
# This service should be used to store processes that you want to temporarily
# disable. Do not change the ServiceType or ServiceStart settings of this
# process.
#
nco_service 'InactiveProcesses'
{
    ServiceType=Non-Master
    ServiceStart=Non-Auto
}

#
# ROUTING TABLE
#
# 'user'      - (optional) only required for secure mode PAD on target host
#                 'user' must be member of UNIX group 'ncoadmin'
# 'password'   - (optional) only required for secure mode PAD on target
host
#                 use nco_pa_crypt to encrypt.
nco_routing
{
```

Exercise 1 Creating the backup ObjectServer

```

host 'host1.tivoli.edu' 'HOST1_PA'
host 'host2.tivoli.edu' 'HOST2_PA'
}

```

The wizard creates the process agent configuration file. The wizard assumes that the processes under the control of the process agent that is run as the **root** user. Most users want to limit the processes that run as **root**. In the next step, you modify the configuration file to run the ObjectServer as the **netcool** user.

- Determine the UID value of the **netcool** user.

```
more /etc/passwd | grep netcool
netcool:x:500:501::/home/netcool:/bin/bash
```

In this example, the UID for the **netcool** user is 500.

- Modify the configuration file:

```
cd /opt/IBM/tivoli/netcool/omnibus/etc
gedit nco_pa.conf
```

- Locate the command that starts the ObjectServer:

```
Command '$OMNIHOME/bin/nco_objserv -name NYC_AGG_B -pa HOST2_PA' run as 0
```

- Change run as 0 to run as 500.

```
Command '$OMNIHOME/bin/nco_objserv -name NYC_AGG_B -pa HOST2_PA' run as
500
```

- Locate the command that starts the Gateway:

```
Command '$OMNIHOME/bin/nco_g_objserv_bi -propsfile
$NCHOME/omnibus/etc/NYC_AGG_GATE.props' run as 0
```

- Change run as 0 to run as 500.

```
Command '$OMNIHOME/bin/nco_g_objserv_bi -propsfile
$NCHOME/omnibus/etc/NYC_AGG_GATE.props' run as 500
```

- Save the changes,

- Verify that the primary ObjectServer is created as follows:

- Examine the ObjectServer property file:

```
cd /opt/IBM/tivoli/netcool/omnibus/etc
more NYC_AGG_B.props
```

- Examine the ObjectServer database directory

```
cd /opt/IBM/tivoli/netcool/omnibus/db/NYC_AGG_B
ls -1
```

```
master_store.tab
table_store.tab
```

5. Start the process agent

```
nco_pad -name HOST2_PA
```



Hint: The directory is not required because the PATH environment variable contains /opt/IBM/tivoli/netcool/omnibus/bin.

6. Verify that the backup ObjectServer is running:

```
nco_ping NYC_AGG_B
```

NCO_PING: Server available.

7. Verify that the primary ObjectServer is running:

```
nco_ping NYC_AGG_P
```

NCO_PING: Server available.

The interfaces file contains the access information for both ObjectServers.

Configuring the Firefox plug-in

Several components in the Web GUI require a special Java plug-in. The plug-in is bundled with Netcool/OMNIbus core libraries. You configure the plug-in to Firefox as the **root** user.



Important: You configured this plug-in on the host1 image in a previous exercise.

1. Configure Firefox plug-in as follows:

a. Change to the **root** user:

```
su -  
Password: object00
```

b. Create the **plugins** directory:

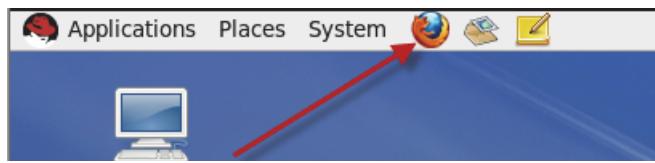
```
cd /usr/lib64/firefox  
mkdir plugins
```

c. Define a symbolic link to the plug-in:

```
cd plugins  
ln -s /opt/IBM/tivoli/netcool/platform/linux2x86/jre64_1.7.0/jre/lib/amd64/libnpjp2.so
```

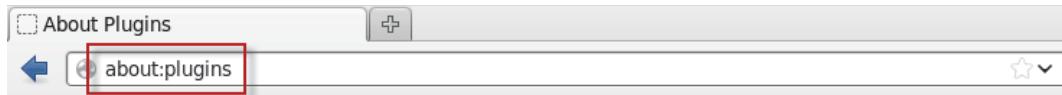
2. Verify that the plug-in is available to Firefox.

- a. Open a Firefox browser.



- b. Enter the following text in the URL field:

about:plugins



Enabled plugins

Find more information about browser plugins at mozilla.org.
Find updates for installed plugins at mozilla.com/plugincheck.
Help for installing plugins is available from plugindoc.mozdev.org.

IBM Developer Kit for Linux,Java,1.7.0

File: libnpjp2.so
Version: 10.45.2
Next Generation Java Plug-in 10.45.2 for Mozilla browsers

- c. Close the Firefox browser.

3. Exit the **root** user and return to the **netcool** user.

`exit`

Exercise 2 Replicating ObjectServer configuration

You revised the existing primary ObjectServer (NYC_AGG_P) in previous exercises. The backup ObjectServer (NYC_AGG_B) does not contain these modifications. You must manually replicate the changes from NYC_AGG_P to NYC_AGG_B. You replicate the changes by running the nco_confpack utility.



Important: Some of the following steps are completed on the host1 image and others on the host2 image. Make sure that you run the required steps on the correct image.

1. Switch to the **host1** image.

2. Export the NYC_AGG_P configuration by running the following command:

```
nco_confpack -export -server NYC_AGG_P -user root -password 'object00' -package /tmp/nyc_agg_p.jar
```

This step creates a file (**/tmp/nyc_agg_p.jar**) that contains the configuration of the NYC_AGG_P ObjectServer.

You move this file to the host2 image with the VSFTP utility. The utility is installed on host1, but it is not currently active.

3. Activate the VSFTP utility as follows:

a. Change to the **root** user.

```
su -  
Password: object00
```

b. Enable the utility.

```
service vsftpd start  
Starting vsftpd for vsftpd: [ OK ]
```

c. Exit the **root** user.

```
exit
```

4. Switch to the **host2** image.

5. Retrieve the configuration file from host1.

```
cd /tmp  
ftp host1  
Connected to host1 (192.168.100.160).  
220 (vsFTPd 2.2.2)  
Name (host1:netcool): netcool  
331 Please specify the password.  
Password: object00  
230 Login successful.  
Remote system type is UNIX.  
Using binary mode to transfer files.
```

```
ftp> cd /tmp  
250 Directory successfully changed.
```

```
ftp> get nyc_agg_p.jar  
local: nyc_agg_p.jar remote: nyc_agg_p.jar  
227 Entering Passive Mode (192,168,100,160,74,5).  
150 Opening BINARY mode data connection for nyc_agg_p.jar (48603 bytes).  
226 Transfer complete.  
48603 bytes received in 0.00102 secs (47650.00 Kbytes/sec)
```

```
ftp> quit  
221 Goodbye.
```

- Import the configuration into NYC_AGG_B by running the following command:

```
nco_confpack -import -server NYC_AGG_B -user root -password 'object00' -package /tmp/nyc_agg_p.jar
```

Enter Y to confirm the import.

The configuration of the NYC_AGG_P ObjectServer is now replicated in the NYC_AGG_B ObjectServer.

Exercise 3 Configuring the ObjectServer gateway

The ObjectServer gateway in a high-availability configuration is referred to as a *synchronizer gateway* because the gateway synchronizes changes between the two ObjectServers. One of the advantages of using the Initial Configuration Wizard is that the wizard creates the gateway configuration files with the suggested best practice configuration. In addition, the Netcool/OMNIbus communication file is updated with the entry for the gateway.

View the server communication information as follows:

- Start the server editor utility:

```
nco_xigen &  
a. View the entry for NYC_AGG_GATE.
```

Server	Hostname	Port	SSL
AGG_Y	host1.tivoli.edu	4100	
Backup1:	host2.tivoli.edu	4100	
HOST1_PA	host1.tivoli.edu	4200	
HOST2_PA	host2.tivoli.edu	4200	
NYC_AGG_B	host2.tivoli.edu	4100	
NYC_AGG_GATE	host2.tivoli.edu	4300	
NYC_AGG_P	host1.tivoli.edu	4100	

The Initial Configuration Wizard created all of the entries in this file, not just the entry for the gateway.

Note: The same definitions exist on the host1 image as well.

- b. Click **Close** to exit the utility.

The details for the entry are shown in the lower portion of the view.

2. Examine the gateway configuration files.

- a. Change to the required directory:

```
cd $OMNIHOME/etc
```

- b. Examine the gateway property file.

```
more NYC_AGG_GATE.props
```

Scroll to the bottom of the file and locate the uncommented lines.

```
MaxLogFileSize: 16384
MessageLevel : 'info'
MessageLog    : '$OMNIHOME/log/NYC_AGG_GATE.log'
Name          : 'NYC_AGG_GATE'
Ipc.Timeout   : 90
Gate.CacheHashTblSize: 50021
Gate.MapFile  : '$OMNIHOME/etc/AGG_GATE.map'
Gate.StartupCmdFile: '$OMNIHOME/gates/objserv_bi/objserv_bi
_startup.cmd'
Gate.Mapper.ForwardHistoricDetails: TRUE
Gate.Mapper.ForwardHistoricJournals: TRUE
Gate.ObjectServerA.Server: 'NYC_AGG_P'
Gate.ObjectServerA.BufferSize: 50
Gate.ObjectServerA.Description: 'failover_gate'
Gate.ObjectServerA.TblReplicateDefFile:
'$OMNIHOME/etc/NYC_AGG_GATE.tblrep.def'
Gate.ObjectServerA.UseBulkInsCmd: TRUE
Gate.ObjectServerB.Server: 'NYC_AGG_B'
Gate.ObjectServerB.BufferSize: 50
Gate.ObjectServerB.Description: 'failover_gate'
Gate.ObjectServerB.TblReplicateDefFile:
'$OMNIHOME/etc/NYC_AGG_GATE.tblrep.def'
Gate.ObjectServerB.UseBulkInsCmd: TRUE
Gate.Resync.Enable: TRUE
Gate.Resync.Type: 'TWOWAYUPDATE'
Gate.Resync.LockType: 'PARTIAL'
```

- c. Observe the name for the gateway mapping file:

```
Gate.MapFile: '$OMNIHOME/etc/AGG_GATE.map'
```

- d. Observe the names for the gateway description:

```
Gate.ObjectServerA.Description: 'failover_gate'
```

```
Gate.ObjectServerB.Description: 'failover_gate'
```

- e. Observe the values for the ObjectServer names:

```
Gate.ObjectServerA.Server: 'NYC_AGG_P'  
Gate.ObjectServerB.Server: 'NYC_AGG_B'
```

- f. Observe the names for the table replication files:

```
Gate.ObjectServerA.TblReplicateDefFile:  
'$OMNIHOME/etc/NYC_AGG_GATE.tblrep.def'  
Gate.ObjectServerB.TblReplicateDefFile:  
'$OMNIHOME/etc/NYC_AGG_GATE.tblrep.def'
```



Note: The gateway is configured to use the same definition file for each direction of replication: ObjectServerA to ObjectServerB and ObjectServerB to ObjectServerA.

- g. Observe the settings for gateway resynchronization:

```
Gate.Resync.Enable: TRUE  
Gate.Resync.Type: 'TWOWAYUPDATE'  
Gate.Resync.LockType: 'PARTIAL'
```



Important: All of these gateway configuration settings are based on suggested best practices.

3. Modify the gateway configuration to include the new column names.

The ObjectServer gateway contains a configuration file that identifies every column in every table that transfers between ObjectServers. Whenever you add extra columns to the ObjectServers, you must update the gateway configuration to include those column names.

The gateway property file identifies the name and location of this file:

```
Gate.MapFile : '$OMNIHOME/etc/AGG_GATE.map'
```

- a. Change to the required directory:

```
cd $OMNIHOME/etc
```

- b. Modify the gateway mapping file:

```
gedit AGG_GATE.map
```

- c. Observe the entries in the top of the file:

```
CREATE MAPPING StatusMap
(
    'Identifier'      = '@Identifier'          ON INSERT ONLY,
    'Node'            = '@Node'                 ON INSERT ONLY,
    'NodeAlias'       = '@NodeAlias'           ON INSERT ONLY
                                NOTNULL '@Node',
```

Each table in the ObjectServer is defined in a MAPPING section. The *StatusMap* definition relates to the alerts.status table. You must modify this section of the file.

- d. Scroll down in the *StatusMap* definition to the end:

```
#####
##  
#  
#      CUSTOM alerts.status FIELD MAPPINGS GO HERE  
#  
#####  
  
#####  
'ServerName'      =      '@ServerName'          ON INSERT ONLY,  
'ServerSerial'   =      '@ServerSerial'        ON INSERT ONLY  
) ;
```

Enter the column names between the two rows of # characters, as follows:

```
#####
#  
#  
#      CUSTOM alerts.status FIELD MAPPINGS GO HERE  
#  
#####  
'Archived_Flag'   =      '@Archived_Flag',  
'Delete_Flag'     =      '@Delete_Flag',  
#####
```

- e. Save the modifications and exit the gedit utility.
4. Examine the gateway replication file.

The gateway property file identifies the name and location of this file:

```
Gate.ObjectServerA.TblReplicateDefFile:  
'$OMNIHOME/etc/NYC_AGG_GATE.tblrep.def'  
Gate.ObjectServerB.TblReplicateDefFile:  
'$OMNIHOME/etc/NYC_AGG_GATE.tblrep.def'
```

Typically a separate file is used for each direction of transfer. The gateway is configured to use a single file. The use of one file ensures that the gateway always transfers the same file contents in each direction.

- a. Change to the required directory:

```
cd $OMNIHOME/etc
```

- b. View the gateway replication file:

```
more NYC_AGG_GATE.tblrep.def
```

Typically the replication file contains only the following entries:

```
REPLICATE ALL FROM TABLE 'alerts.status'
    USING MAP 'StatusMap';

REPLICATE ALL FROM TABLE 'alerts.journal'
    USING MAP 'JournalMap';

REPLICATE ALL FROM TABLE 'alerts.details'
    USING MAP 'DetailsMap';
```

These statements ensure that any changes to the event record tables are replicated between the ObjectServers.

This gateway replication file contains many more entries. These entries are used to configure the gateway to replicate changes to most ObjectServer tables. With these entries in place, the gateway replicates changes such as these examples:

- ◆ New users
- ◆ New groups
- ◆ New menus
- ◆ New Tools
- ◆ Others

This configuration facilitates ObjectServer administration. For example, when a Netcool administrator adds a tool to the primary ObjectServer, the gateway replicates the tool configuration to the backup ObjectServer.

5. Restart the gateway as follows.

The Initial Configuration Wizard configured the process activity daemon to manage the gateway. The gateway was started when the process activity daemon was started previously. You stop the running gateway that, and process activity restarts it.

- a. Locate the current PID value:

```
ps -ef | grep nco_g
```

```
netcool  19727 19698  0 17:57 ?          00:00:06
/opt/IBM/tivoli/netcool/omnibus/platform/linux2x86/bin64/nco_g_objserv_bi
-propsfile /opt/IBM/tivoli/netcool/omnibus/etc/NYC_AGG_GATE.props
```

- b. Stop the gateway.

```
kill -9 19727
```

- c. Locate the current PID value:

```
ps -ef | grep nco_g
```

```
netcool 24328 19698 2 19:24 ? 00:00:00  
/opt/IBM/tivoli/netcool/omnibus/platform/linux2x86/bin64/nco_g_objserv_bi  
-propsfile /opt/IBM/tivoli/netcool/omnibus/etc/NYC_AGG_GATE.props
```

The PID value is changed that verifies that the gateway is restarted.

The NYC_AGG_GATE is now replicating changes between NYC_AGG_P and NYC_AGG_B.

6. Verify replication.

- a. Start a Simnet probe and send event to NYC_AGG_P.

```
nco_p_simnet -server NYC_AGG_P &
```



Important: You just started a probe on *host2* and directed the probe to send events to an ObjectServer on *host1*.

- b. Start a native event list as follows:

```
nco_event &
```

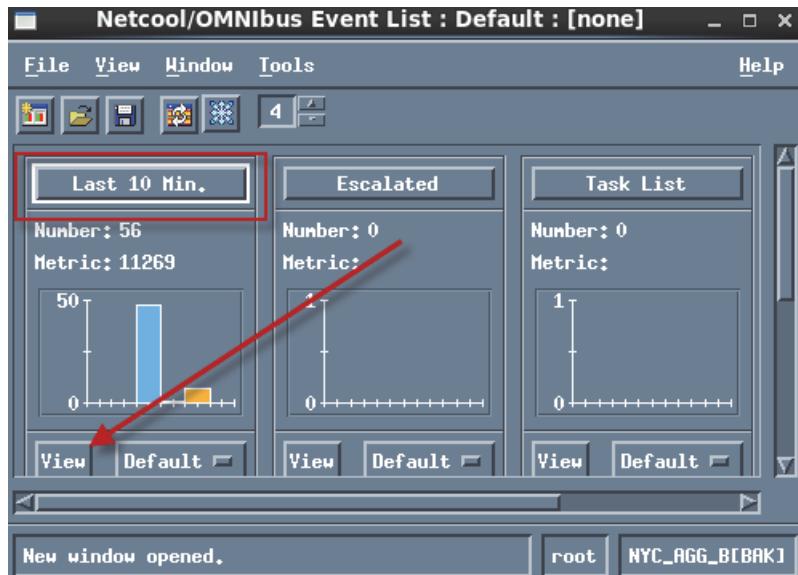
- c. Connect to the NYC_AGG_B ObjectServer as the **root** user, with password **object00**:



- d. Click **OK**.

The **Event List** window opens.

- e. Locate the box that is labeled **Last 10 Mins** and click **View**:



The **Sub-Event List** view opens. The Simnet probe produces the events that are shown in the screen capture. However, the probe is sending these events to the NYC_AGG_P ObjectServer. The fact that they show in the NYC_AGG_B ObjectServer indicates that the NYC_AGG_GATE is replicating these events from NYC_AGG_P to NYC_AGG_B. The correct events in NYC_AGG_B verify that the NYC_AGG_GATE is replicating event data correctly.

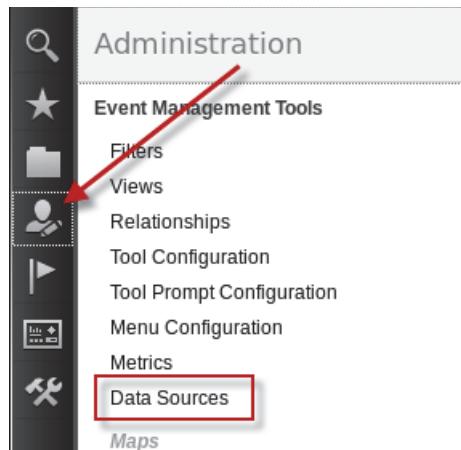
Node	Alert Group	Summary
London	Systems	Machine has gone offline
link3	Link	Link Down on port
Washington	Systems	Machine has gone offline
Beijing	Stats	Diskspace alert
Sydney	Systems	Machine has gone offline
Moscow	Systems	Machine has gone offline
Beijing	Stats	Diskspace alert
host2.tivoli.ed	ConnectionStatus	GATEWAY: failover_gate connected from host host2.tivoli.ed

- f. Close the **Sub-Event List** window.
g. Close the **Event List** window.

Exercise 4 Configuring Web GUI for ObjectServer high availability

The Web GUI does not use the interfaces file like probes, gateways, and desktops. Instead, an XML file contains the access information for the ObjectServer. The installation of the Web GUI component updated the XML file with the access information for the NYC_AGG_P ObjectServer. You can manually edit the XML file to add the NYC_AGG_B ObjectServer. Or you can add the NYC_AGG_B ObjectServer with the Dashboard Application Services Hub.

1. Open a Firefox browser.
2. Log in to Dashboard Application Services Hub as the **ncoadmin** user.
3. Click the icon and select **Data Sources**.

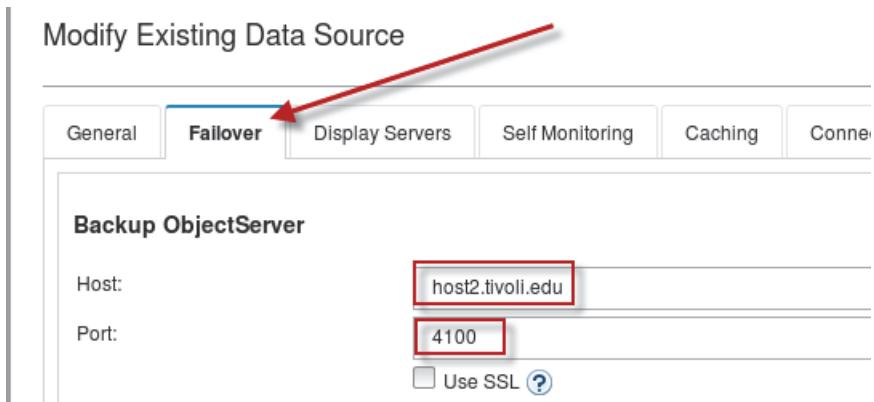


4. Click **OMNIBUS** to select it and click the icon the edit the entry.

Name	Enabled
OMNIBUS	true

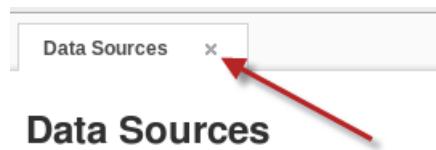
Note: The Web GUI configuration tool created the existing entry after the Web GUI component was installed.

5. Click **Failover**, enter **host2.tivoli.edu** for the host and **4100** for the port.



6. Scroll to the bottom of the page and click **Save Datasource**.

7. Click the X to close the **Data Sources** page.



The Web GUI component is now configured to use both the primary and backup ObjectServers. The change is made dynamically and there is no requirement to restart Dashboard Application Services Hub. In addition, the XML definition file is modified.

8. Examine the XML file.

- a. Change to the required directory.

```
cd /opt/IBM/netcool/omnibus_webgui/etc/datasources
```

- b. Examine the modifications.

```
tail -20 ncwDataSourceDefinitions.xml
```

```

<ncwPrimaryServer>
  <ncwOSConnection maxPoolSize="10" port="4100" host="host1.tivoli.edu"
ssl="false" minPoolSize="5"/>
</ncwPrimaryServer>

<!-- The optional failover ObjectServer to connect to -->
<ncwBackUpServer>
  <ncwOSConnection maxPoolSize="10" port="4100" host="host2.tivoli.edu"
ssl="false" minPoolSize="5"/>
</ncwBackUpServer>
</ncwFailOverPairDefinition>
```

The definition for *ncwPrimaryServer* is the access information for the primary ObjectServer, host, and port. The definition for *ncwBackUpServer* contains the modifications. The values for host and port point to the backup (NYC_AGG_B) ObjectServer.



Hint: The text shown here is the same changes that are made if you want to manually configure the Web GUI component for a backup ObjectServer. However, if you manually modify this file, you must restart Dashboard Application Services Hub.

Exercise 5 Verifying ObjectServer high availability

The primary reason for implementing ObjectServer high availability is to ensure that users always have access to event data, even if an ObjectServer is down. There are two user interfaces available, Native Desktop and Web GUI. The following steps verify that each type of interface provides access to event data when the primary ObjectServer is not available.



Note: The verification steps can be performed from either image.

1. Verify that both ObjectServers are available as follows:

```
nco_ping NYC_AGG_P  
NCO_PING: Server available.
```

```
nco_ping NYC_AGG_B  
NCO_PING: Server available.
```

2. Verify the native Event List.

- a. Start a native Event List:

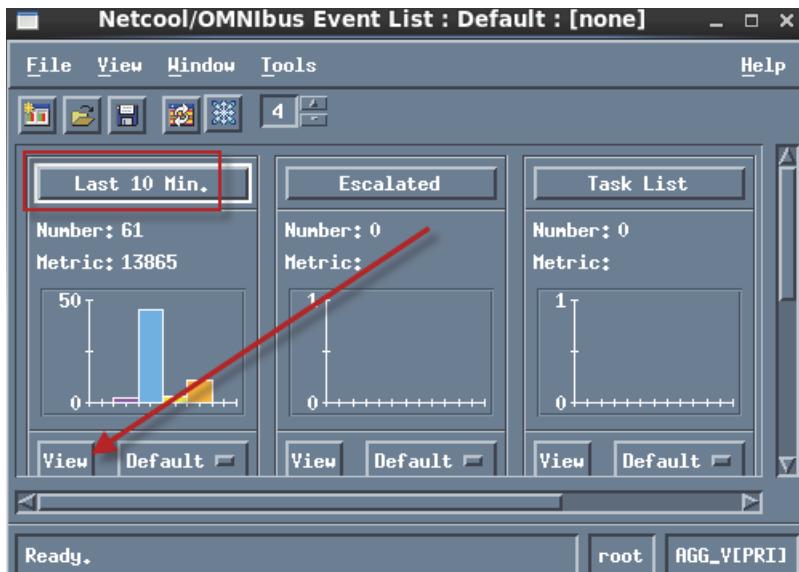
```
nco_event &
```

- b. Connect to the *virtual* ObjectServer: AGG_V as the **root** user and click **OK**.



The Event List window opens.

- c. Locate the box that is labeled **Last 10 Min.**, and click **View**.



The **Sub-Event List** window opens.

The screenshot shows the Netcool/OMNIbus Event List window with the following details:

- Title Bar:** Netcool/OMNIbus Event List : Filter="Last 10 Min.", View="Default"
- Menu Bar:** File, Edit, View, Alerts, Tools, Help
- Toolbar:** Includes icons for search, filters, and navigation.
- Filter Bar:** Last 10 Min., Default
- Table Headers:** Node, Alert Group, Summary
- Data Rows:**
 - link2, Link, Link Down on port
 - link4, Link, Link Down on port
 - Moscow, Systems, Machine has gone offline
 - London, Systems, Machine has gone offline
 - Washington, Systems, Machine has gone offline
 - link5, Link, Link Down on port
 - link1, Link, Link Down on port
 - Sydney, Systems, Machine has gone offline
- Bottom Navigation:** Buttons for 0, 2, 45, 3, 9, 0, All Events, and a dropdown for row count (0, root, AGG_V[PRI]).
- Status Bar:** 59 row(s) matched.

Notice the text in the lower right corner of the window: **AGG_V [PRI]**. The reference to AGG_V indicates that the user is connected to an ObjectServer called AGG_V. AGG_V is the name for the *virtual* ObjectServer. The reference to PRI indicates that the user is connected to an ObjectServer that is considered the primary ObjectServer.

- d. Locate the process name for the primary ObjectServer as follows:

```
nco_pa_status -server HOST1_PA -password 'object00'
```

Service Name	Process Name	Hostname	User	Status	PID
Core 879	MasterObjectServer	host1.tivoli.edunetcool		RUNNING	3

- e. Stop the primary ObjectServer as follows:

```
nco_pa_stop -server HOST1_PA -password 'object00' -process  
MasterObjectServer
```

- f. Verify the availability of the primary ObjectServer:

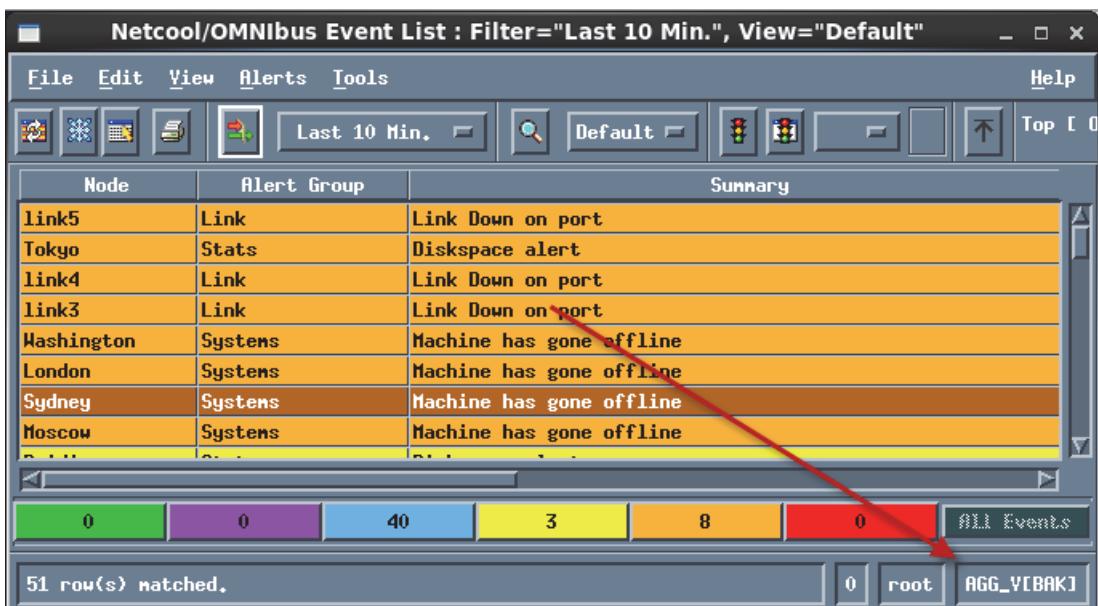
```
nco_ping NYC_AGG_P  
NCO_PING: Server unavailable.
```

- g. Observe the pop-up window:



This message, which is generated by the native Event List, indicates that the ObjectServer is no longer available.

- h. Click **Connect**.



Notice the text in the lower right corner of the window: **AGG_V [BAK]**. The reference to AGG_V indicates that the user is connected to an ObjectServer called AGG_V. The reference to BAK indicates that the user is connected to an ObjectServer that is considered the backup ObjectServer. The user still has access to event data even though the primary ObjectServer is down.

- i. Restart the primary ObjectServer:

```
nco_pa_start -server HOST1_PA -password 'object00' -process  
MasterObjectServer
```

- j. Verify the availability of the primary ObjectServer:

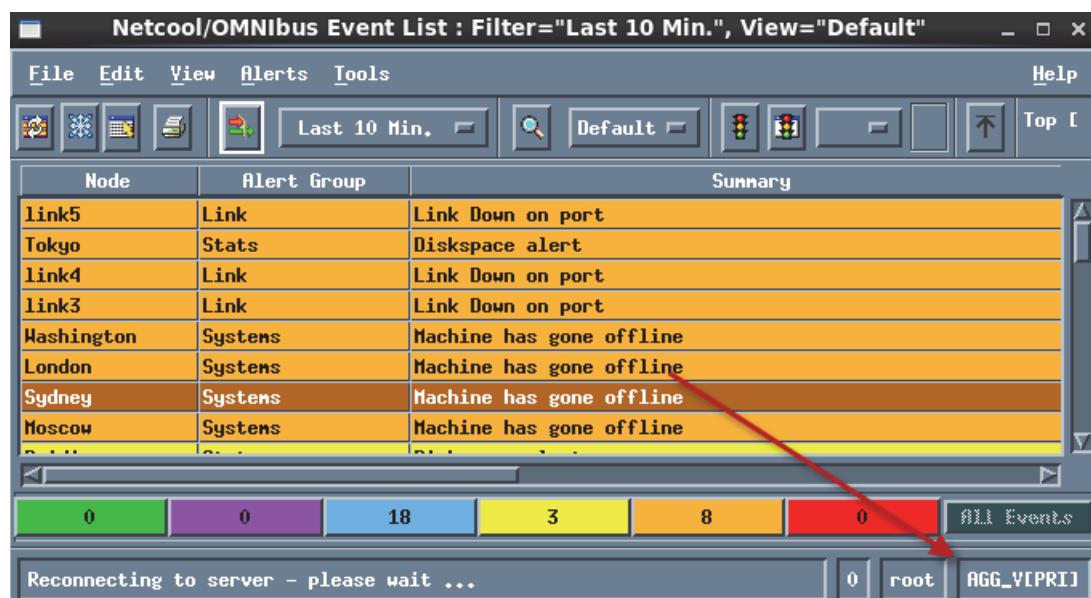
```
nco_ping NYC_AGG_P  
NCO_PING: Server available.
```

- k. Observe the pop-up window:



This message, which is generated by the native Event List, indicates that the primary ObjectServer is available.

- l. Click **OK**.



The text in the lower right corner indicates that the user is now connected to the primary (**PRI**) ObjectServer.

The Native Event List uses an ObjectServer property to determine when it is connected to a backup ObjectServer. This property is *BackupObjectServer*. In the primary ObjectServer, the property is set to FALSE, the default value. In the backup ObjectServer, the property is set to TRUE. When the native Event List connects to either ObjectServer, it queries the value of this property. When connected to the primary, the value is FALSE, indicating that the ObjectServer is *not* the backup. When connected to the backup, the value is TRUE, indicating that the ObjectServer *is* the backup.



Important: Another advantage to using the Initial Configuration Wizard relates to the BackupObjectServer property. When you manually create an ObjectServer by running the nco_dbinit utility, the default value for the property is FALSE. If you create an ObjectServer, and do not use the Initial Configuration Wizard, you must manually change this property for your backup ObjectServer.

m. Close all windows that are related to the Native Event list.

3. Verify Dashboard Application Services Hub.



Note: You should still be logged in as the **ncoadmin** user. If not, do so now.

a. Open the Event Viewer.

Group	Cou	Sev	Sev	Ack	Node	Alert Group	Summary
All	126	✖	✖	No	host1	TopNodes	ALERT: 1
Critical	1	✖	⚠	Yes	Sydney	Systems	Machine
Major	12	⚠	⚠	No	link4	Link	Link Dow
Minor	5	⚠	⚠	No	Tokyo	Stats	Diskspa
Warning	47	ℹ	⚠	No	host1.tivoli.edu	ProbeStatus	ALERT: 1
Indeterminate	53	⌚	⚠	No	Beijing	Stats	Diskspa
Clear	8	✓	⚠	No	link3	Link	Link Dow
			⚠	No	link2	Link	Link Dow

b. Stop the primary ObjectServer as follows:

```
nco_pa_stop -server HOST1_PA -password 'object00' -process
MasterObjectServer
```

c. Verify the availability of the primary ObjectServer:

```
nco_ping NYC_AGG_P
NCO_PING: Server unavailable.
```

- d. Observe the Event Viewer window:

The screenshot shows the Event Viewer window with a list of events. The toolbar at the top includes a refresh icon (blue arrow), a search icon, and several status indicators. The main area displays a table of events with columns for Group, Cou, Sev, Sev, Ack, Node, and Alert Group. The 'All' group has 126 events, mostly marked as 'No' for Acknowledged. Other groups like 'Critical', 'Major', 'Minor', 'Warning', 'Indeterminate', and 'Clear' also have entries. The 'Alert Group' column shows categories such as TopNodes, Systems, Stats, ProbeStatus, Stats, Systems, and Link.

Group	Cou	Sev	Sev	Ack	Node	Alert Group
All	126	(red X)	(red X)	No	host1	TopNodes
Critical	1	(red X)	(orange exclamation)	Yes	Sydney	Systems
Major	10	(orange exclamation)	(orange exclamation)	No	London	Systems
Minor	5	(orange exclamation)	(orange exclamation)	No	Tokyo	Stats
Warning	41	(blue info)	(orange exclamation)	No	host1.tivoli.edu	ProbeStatus
Indeterminate	53	(blue diamond)	(orange exclamation)	No	Beijing	Stats
Clear	16	(green checkmark)	(green checkmark)	No	Washington	Systems
			(orange exclamation)	No	link6	Link

Click the blue arrows a couple of time to refresh the contents of the window. It is not apparent that the primary ObjectServer is down. This behavior is normal for the Web GUI user interface. It is not apparent that the user is connected to the primary or backup ObjectServer. However, the fact that the user sees events and that the primary ObjectServer is down, verifies that Web GUI is configured correctly for ObjectServer high availability.

- e. Restart the primary ObjectServer:

```
nco_pa_start -server HOST1_PA -password 'object00' -process  
MasterObjectServer
```

- f. Verify the availability of the primary ObjectServer:

```
nco_ping NYC_AGG_P  
NCO_PING: Server available.
```

The Web GUI component does not fail over, and fail back between the primary and backup ObjectServers like the native desktop. Instead, the Web GUI component connects to both ObjectServers simultaneously at start. The Web GUI component accesses only the primary ObjectServer whenever it is available. If the primary ObjectServer is not available, then the Web GUI component accesses the backup ObjectServer.

- g. Log out of Dashboard Application Services Hub.

- h. Close the Firefox browser.

Exercise 6 Enabling triggers for high availability

Many experienced users configure triggers that modify or delete events so they are only active on the primary ObjectServer. The best way to ensure that these triggers are only active on the primary ObjectServer is to place them in the **primary_only** trigger group. This trigger group exists on both the primary, and backup ObjectServers. However, when the primary ObjectServer is active, the trigger group is disabled on the backup ObjectServer. All triggers that belong to that trigger group on the backup ObjectServer cannot run. If the primary ObjectServer fails, a trigger in the *backup* ObjectServer detects the failure and enables the **primary_only** trigger group. Enabling the trigger group allows the triggers in that group to run. When the primary ObjectServer recovers, a different trigger in the *backup* ObjectServer disables the **primary_only** trigger group. These triggers exist in both the primary, and backup ObjectServers. They are disabled in both ObjectServers by default. To enable the behavior that is just described, you must enable the triggers in the *backup* ObjectServer.

1. Enable the triggers as follows.

- a. Start the administrator utility:

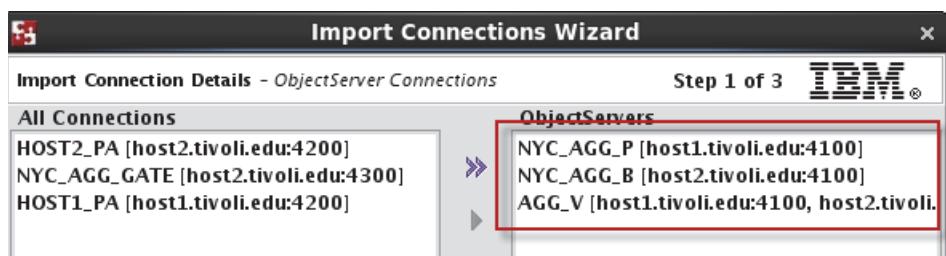
```
nco_config &
```

- b. Click **Yes** to import the connection details.



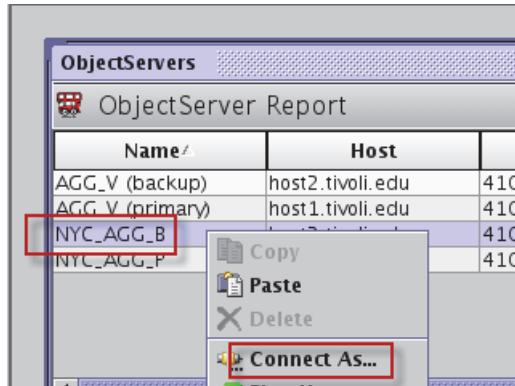
- c. Click **Next**.

- d. Click **Finish**.



Observe that the list of ObjectServers contains the primary (NYC_AGG_P), the backup (NYC_AGG_B), and the virtual (AGG_V).

- e. Connect to the NYC_AGG_B ObjectServer as the **root** user.

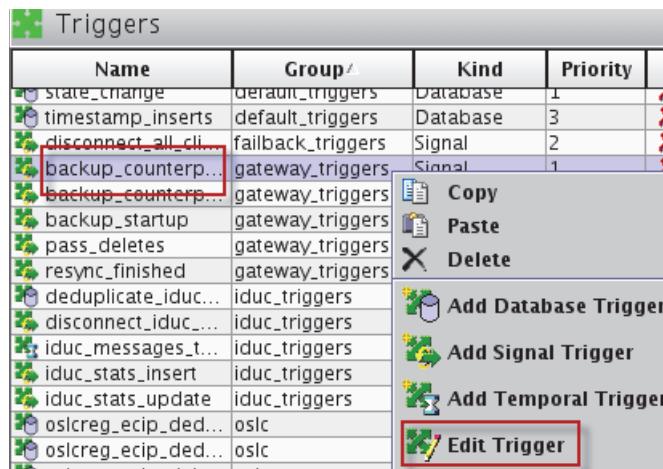


- f. Expand **Automation** and click **Triggers**.
g. Click the **Group** column name to sort the contents by that column.
h. Scroll down in the list and locate the triggers in the *gateway_triggers* group.

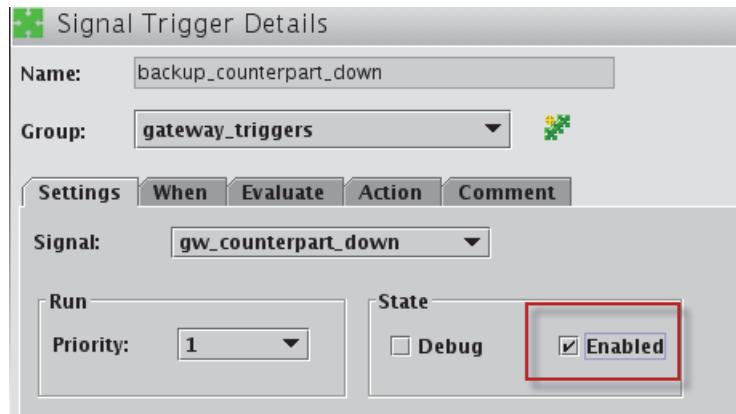
Name	Group	Kind	Priority	Debug	Enabled
state_change	default_triggers	Database	1	<input checked="" type="checkbox"/> raise	<input checked="" type="checkbox"/> true
timestamp_inserts	default_triggers	Database	3	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> true
disconnect_all_cli	failback_triggers	Signal	2	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> false
backup_counterp...	gateway_triggers	Signal	1	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> false
backup_counterp...	gateway_triggers	Signal	1	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> false
backup_startup	gateway_triggers	Signal	1	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> false
pass_deletes	gateway_triggers	Signal	1	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> false
resync_finished	gateway_triggers	Signal	1	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> true
deduplicate_iduc...	iduc_triggers	Database	1	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> true
disconnect_iduc...	iduc_triggers	Signal	1	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> true
iduc_messages_t...	iduc_triggers	Temporal	1	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> true

Five triggers are listed for the *gateway_triggers* group. Four of these triggers are currently disabled. Modify all four triggers, and change them to enabled.

- i. Click the first trigger to select it, right-click, and select **Edit Trigger**.



- j. Check the box that is labeled **Enabled** and click **OK** to save the changes.



That trigger is now enabled. Repeat these steps to enable the remaining three triggers.

Name	Group	Kind	Priority	Debug	Enabled	Gr
state_change	default_triggers	Database	1	✗ raise	✓ true	✓
timestamp_inserts	default_triggers	Database	3	✗ false	✓ true	✓
disconnect_all_cli	failback_triggers	Signal	2	✗ false	✗ false	✓
backup_counterpart...	gateway_triggers	Signal	1	✗ false	✓ true	✓
backup_counterpart...	gateway_triggers	Signal	1	✗ false	✓ true	✓
backup_startup	gateway_triggers	Signal	1	✗ false	✓ true	✓
pass_deletes	gateway_triggers	Signal	1	✗ false	✓ true	✓
resync_finished	gateway_triggers	Signal	1	✗ false	✓ true	✓
deduplicate_iduc...	iduc_triggers	Database	1	✗ raise	✓ true	✓
disconnect_iduc...	iduc_triggers	Signal	1	✗ false	✓ true	✓

All the triggers in the gateway_trigger group are now enabled.



Note: The primary_only trigger is an excellent example of why the ObjectServer gateway does not automatically replicate trigger modifications. You must enable these triggers on the backup ObjectServer to implement the feature that enables and disables the primary_only trigger group. You must *not* enable this behavior on the primary ObjectServer.

The ObjectServer gateway controls the triggers that enable and disable the primary_only trigger group. It is the gateway that detects if the primary ObjectServer fails and when it recovers. The gateway generates *signals* that are based on certain conditions. These signals cause the triggers to run.

2. Verify gateway triggers.

Observe the current state of the primary_only trigger group on the NYC_AGG_B ObjectServer.

a. Click **Trigger Groups**.

Configuration of NYC_AGG_B on host2.tivoli.edu:4100		
	Name	Enable
User	AdvCorr	✓ true
Menu	audit_config	✗ false
Automation	automatic_backup_system	✓ true
	compatibility_triggers	✓ true
	connection_watch	✓ true
	default_triggers	✓ true
	fallback_triggers	✓ true
	gateway_triggers	✓ true
	iduc_triggers	✓ true
	oslc	✓ true
Trigger Groups	primary_only	✓ true
	profiler_triggers	✓ true

The primary_only trigger group is enabled. This condition is not the correct state for this trigger group on the backup ObjectServer when the primary ObjectServer is active.

b. Stop the ObjectServer gateway:

```
ps -ef | grep nco_g
```

```
netcool 3934 3905 0 12:50 ? 00:00:02
/opt/IBM/tivoli/netcool/omnibus/platform/linux2x86/bin64/nco_g_objserv_b
-propsfile /opt/IBM/tivoli/netcool/omnibus/etc/NYC_AGG_GATE.props
```

```
kill 3934
```

Process Activity automatically restarts the gateway.

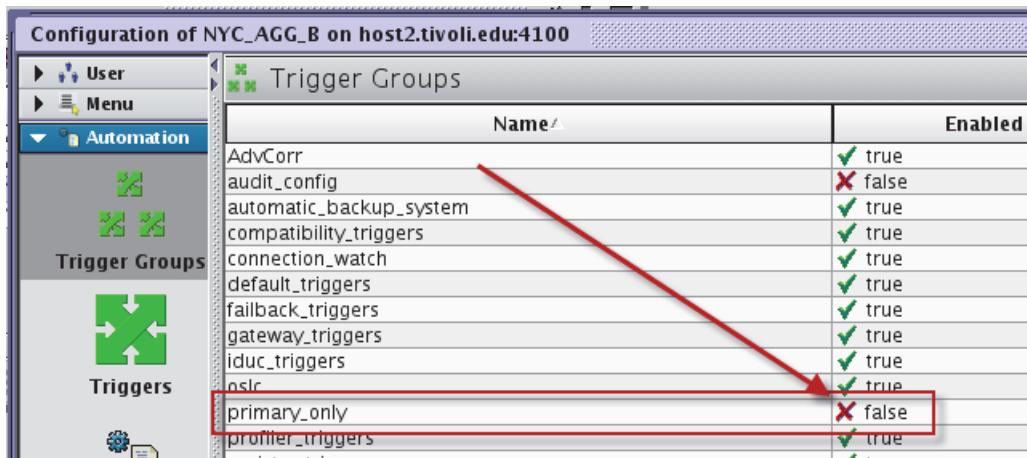
c. Verify that the ObjectServer gateway is restarted.

```
ps -ef | grep nco_g
```

```
netcool 5762 3905 0 13:21 ? 00:00:00
/opt/IBM/tivoli/netcool/omnibus/platform/linux2x86/bin64/nco_g_objserv_b
-propsfile /opt/IBM/tivoli/netcool/omnibus/etc/NYC_AGG_GATE.props
```

d. Observe the current state of the primary_only trigger group on the AGG_B ObjectServer:

Click **Triggers** and click **Trigger Groups**. This action refreshes the **Trigger Groups** pane.



When the gateway restarts, it detects the presence of the primary ObjectServer and generates a specific signal on the backup ObjectServer. That signal causes a trigger to run, and the trigger disables the `primary_only` trigger group.

Leave the administrator utility open. You use it again shortly.

Exercise 7 Configuring probes for ObjectServer high availability

In a previous exercise, you configured the Syslog, and SNMP probes to connect to the `NYC_AGG_P` ObjectServer. You must modify their configurations to connect to the *virtual* ObjectServer. When the probes are configured to use the *virtual* ObjectServer, they fail over automatically to the backup if the primary fails. You have two options as to what causes the probes to *fail back* when the primary ObjectServer recovers.

The first option is to configure the probes to fail back based on time. For this option, you must configure two property settings in each probe:

```
NetworkTimeout      : 10
PollServer          : 10
```

These settings cause the probes, when connected to the backup ObjectServer, to periodically test for the availability of the primary ObjectServer. If the primary ObjectServer is available, the probes automatically fail back to the primary. The values are specified in seconds.

The second option is to configure the probes to not fail back on their own. For this option, you leave the probe property settings at their default values:

```
NetworkTimeout      : 0
PollServer          : 0
```

These settings disable fail back by the probe. Instead, you enable a trigger on the backup ObjectServer. This trigger disconnects the probes after the gateway completes resynchronization. For this exercise, you use this option.

1. Return to the administrator utility, and the **Automation** feature.
 - a. Verify that you are connected to the NYC_AGG_B ObjectServer and click **Triggers**.
 - b. Locate the trigger **disconnect_all_clients**.

Name	Group	Kind	Priority	Debug	Enabled
delete_clears	primary_only	Temporal	1	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> true
delete_stats	stats_triggers	Temporal	20	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> true
details_inserts	stats_triggers	Database	20	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> true
disable_inactive_users	security_wa...	Temporal	1	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> false
disable_user	security_wa...	Signal	1	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> true
disconnect_all_clients	failback_trig...	Signal	2	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> false
disconnect_iduc_missed	iduc_triggers	Signal	1	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> true
escalate_off	primary_only	Temporal	1	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> false
expire	primary_only	Temporal	1	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> true
flash_not_ack	primary_only	Temporal	1	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> false
generic_clear	primary_only	Temporal	1	<input checked="" type="checkbox"/> false	<input checked="" type="checkbox"/> true

Observe that the trigger is *disabled*.

- c. Edit the trigger and enable it.

Leave the administrator utility open. You use it again shortly.



Important: The probes are running on the host1 image. You must perform the following probe modification steps on the host1 image.

2. Switch to the **host1** image.
3. Configure the Syslog probe to connect to the *virtual* ObjectServer.

- a. Change to the probes directory:

```
cd $OMNIHOME/probes/linux2x86
```

- b. Modify the property file:

```
gedit syslog.props
```

- c. Change the server property as follows:

```
Server : "AGG_V"
```

- d. Save the changes and exit gedit.

- e. Stop the running probe:

```
pkill nco_p_syslog
```

- f. Verify that the probe is stopped.

```
ps -ef | grep nco_p_syslog
```

- g. Start the probe:

```
nco_p_syslog &
```

- h. Verify that the probe is running.

```
ps -ef | grep nco_p_syslog
```

```
netcool 6775 4727 37 13:43 pts/0 00:00:03  
/opt/IBM/tivoli/netcool/omnibus/probes/linux2x86/nco_p_syslog
```

4. Configure the SNMP probe to connect to the *virtual* ObjectServer.

- a. Modify the property file:

```
gedit mttrapd.props
```

- b. Change the server property as follows:

```
Server : "AGG_V"
```

- c. Save the changes and exit gedit.

- d. Stop the running probe:

```
pkill nco_p_mttrapd
```

- e. Verify that the probe is stopped.

```
ps -ef | grep nco_p_mttrapd
```



Important: The probe takes a minute or two to stop. Make sure that the probe is stopped before restarting it.

- f. Start the probe:

```
nco_p_mttrapd &
```

- g. Verify that the probe is running.

```
ps -ef | grep nco_p_mttrapd
```

```
netcool 7070 4727 42 13:48 pts/0 00:00:02  
/opt/IBM/tivoli/netcool/omnibus/platform/linux2x86/probes64/nco_p_mttrapd
```

The following steps verify that the probes behave as expected during an ObjectServer failure, and recovery.

5. Switch to the **host2** image.

6. Return to the administrator utility.
 - a. Verify that you are connected to the NYC_AGG_B ObjectServer.
 - b. Expand **System**, scroll down, and click **Connections**.

The screenshot shows the 'ObjectServer Connections' table. A red arrow points to the 'System' icon in the left sidebar, and a red box highlights the 'Connections' icon in the same sidebar. The table has columns: Application Name, Host, Login Name, and Description. The data is as follows:

Application Name	Host	Login Name	Description
Administrator	host2	root	8
GATEWAY	host2.tivoli.edu	gateway	failover_gate
GATEWAY	host2.tivoli.edu	gateway	failover_gate
WEBTOP	host2.tivoli.edu	root	2
WEBTOP	host2.tivoli.edu	root	3
WEBTOP	host2.tivoli.edu	root	4
WEBTOP	host2.tivoli.edu	root	5
WEBTOP	host2.tivoli.edu	root	6

This table shows every component that is connected to the ObjectServer. Currently, three components are connected: the gateway, Web GUI, and the Administrator GUI.

7. Stop the *primary* ObjectServer.
 - a. Stop the ObjectServer by running the command-line utility:


```
nco_pa_stop -server HOST1_PA -password object00 -process MasterObjectServer
```
 - b. Verify that the ObjectServer is stopped:


```
nco_pa_status -server HOST1_PA -password object00
```

Service Name	Process Name	Hostname	User	Status	PID
Core 0	MasterObjectServer	host1.tivoli.edunetcool		DEAD	

The ObjectServer is down.

8. Examine the contents of the **Connections** table:

Configuration of NYC_AGG_B on host2.tivoli.edu:4100

Application Name	Host	Login Name	Description	Conn
Administrator	host2	root		8
GATEWAY	host2.tivoli.edu	gateway	failover_gate	1
GATEWAY	host2.tivoli.edu	gateway	failover_gate	7
PROBE	host1.tivoli.edu	probe	mttrapd	9
PROBE	host1.tivoli.edu	probe	syslog	10
WEBTOP	host2.tivoli.edu	root		2
WEBTOP	host2.tivoli.edu	root		3
WEBTOP	host2.tivoli.edu	root		4
WEBTOP	host2.tivoli.edu	root		5
WEBTOP	host2.tivoli.edu	root		6

The view does not refresh automatically. Click some other entry, like **SQL** and then click **Connections**.

The two probes are connected to the backup ObjectServer as a result of the primary ObjectServer going down.

9. Examine the primary_only trigger group.

Configuration of NYC_AGG_B on host2.tivoli.edu:4100

Name	Enabled
AdvCorr	true
audit_config	false
automatic_backup_system	true
compatibility_triggers	true
connection_watch	true
default_triggers	true
fallback_triggers	true
gateway_triggers	true
iduc_triggers	true
oslc	true
primary_only	true
promoter_triggers	true

The primary_only trigger group is now enabled.

10. Restart the primary ObjectServer:

```
nco_pa_start -server HOST1_PA -password object00 -process MasterObjectServer
```

11. Examine the **Connections** table:

The screenshot shows the 'ObjectServer Connections' table in the Administrator GUI. The table has columns: Application Name, Host, Login Name, and Description. There are 8 rows of data. A callout bubble points to the last row with the text 'probes are gone'.

Application Name	Host	Login Name	Description	
Administrator	host2	root		8
GATEWAY	host2.tivoli.edu	gateway	failover_gate	1
GATEWAY	host2.tivoli.edu	gateway	failover_gate	7
WEBTOP	host2.tivoli.edu	root		2
WEBTOP	host2.tivoli.edu			3
WEBTOP	host2.tivoli.edu			4
WEBTOP	host2.tivoli.edu	root		5
WEBTOP	host2.tivoli.edu	root		6

The probe entries are gone from the table, which indicates that the probes failed back to the primary ObjectServer.



Hint: Recall that the controlled failback feature is enabled. The gateway must complete ObjectServer synchronization before the probes are disconnected. The probes entries might remain in the table for a short time after the ObjectServer is restarted.

12. Examine the primary_only trigger group.

The screenshot shows the 'Trigger Groups' table in the Administrator GUI. The table has columns: Name and En. A red arrow points to the 'primary_only' row, which is highlighted with a red box and has a red X in the 'En' column.

Name	En
AdvCorr	true
audit_config	false
automatic_backup_system	true
compatibility_triggers	true
connection_watch	true
default_triggers	true
fallback_triggers	true
gateway_triggers	true
iduc_triggers	true
oslc	true
primary_only	false
profiler_triggers	true
registrv triggers	true

The primary_only trigger group is now disabled.

13. Exit the Administrator GUI.

The following list is a summary of the accomplishments from this unit:

- The backup ObjectServer is created and running.
- The synchronizer gateway is configured, and running.
- Virtual ObjectServer is defined.
- The Web GUI configuration is modified for ObjectServer high availability.
- ObjectServer high availability function is verified.
- Configured both probes for ObjectServer high availability.
- The backup ObjectServer is configured to disconnect clients when the gateway finishes resynchronization.



9 Process control exercises

The exercises in this unit configure Netcool/OMNIbus Process Activity to automatically start all core components: ObjectServers, probes, and gateway.

Exercise 1 Configuring the Pluggable Authentication Module (PAM) on host1

The process activity daemon requires a file that defines the PAM authentication configuration. The contents of the file varies by operating system. The most consistent way to create the required file is to copy an existing file.

1. Switch to the **host1** image.

2. Change to the *root* user:

```
su -  
Password: object00
```

3. Copy the existing file and rename it:

```
cd /etc/pam.d  
cp system-auth netcool
```

```
cp: overwrite `netcool'? y  
Enter y to replace the existing file.
```

It is the *netcool* file that provides the PAM authentication for Netcool® components.

4. Examine the permissions on /etc/shadow:

```
ls -la /etc/shadow  
-r--r--r-- 1 root root 1282 Aug 29 17:40 /etc/shadow
```

The file permissions on this file are changed to allow read access to non-root users. This change is necessary to allow the process activity daemon to run as a non-root user.

5. Install the process activity start file:

```
cd /opt/IBM/tivoli/netcool/omnibus/install/startup  
  
.linux2x86install
```

Enter value for \$NCHOME [/opt/IBM/tivoli/netcool] :

This script copies a startup script into the /etc/init.d directory to enable you to automatically start and stop Netcool/OMNIbus processes.

It does this by:

```
Copying linux2x86/etc/rc.d/init.d/nco to /etc/init.d/nco  
Running "/sbin/chkconfig --add nco"
```

Do you wish to continue (y/n) ? [y]

Name of the Process Agent Daemon [NCO_PA] : **HOST1_PA**

Should HOST1_PA run in secure mode (y/n) ? [y]

Enter value for environment variable NETCOOL_LICENSE_FILE

if required [27000@localhost] :

Scripts installed.



Important: Make sure to specify **HOST1_PA** as the Process Agent Daemon.

This step installs a script **/etc/init.d/nco**. This script starts the process activity daemon at system start.

6. Configure the process activity daemon to run as the **netcool** user.

The process activity daemon typically runs as the **root** user. The process runs as the **root** user because the daemon must be able to start a process as a specific user. If the daemon starts processes as only the **netcool** user, the daemon can run as the **netcool** user. The script that you installed in the previous step runs the process activity daemon as the **root** user. Modify the script as follows to run it as the **netcool** user.

a. Change to the required directory:

```
cd /etc/init.d
```

b. Modify the script:

```
gedit nco
```

Locate the following lines:

```
if [ "$SECURE" = "Y" ] ; then  
${OMNIHOME}/bin/nco_pad -name ${NCO_PA} -authenticate PAM -  
secure > /dev/null 2> /dev/null  
else  
${OMNIHOME}/bin/nco_pad -name ${NCO_PA} -authenticate PAM >  
/dev/null 2> /dev/null  
fi
```

Change as follows:

```
if [ "$SECURE" = "Y" ] ; then
  su - netcool -c "${OMNIHOME}/bin/nco_pad -name ${NCO_PA} -
  authenticate PAM -secure > /dev/null 2> /dev/null"
else
  su - netcool -c "${OMNIHOME}/bin/nco_pad -name ${NCO_PA} -
  authenticate PAM > /dev/null 2> /dev/null"
fi
```

The change is to add **su - netcool -c** to the beginning of the two lines and to enclose each command in double quotations.

- Add the NC_RULES_HOME environment variable

Locate the following lines:

```
NETCOOL_LICENSE_FILE=27000@localhost # Points to flex license server
```

```
export NCHOME OMNIHOME NCO_PA NETCOOL_LICENSE_FILE
```

Change as follows:

```
NETCOOL_LICENSE_FILE=27000@localhost # Points to flex license server
NC_RULES_HOME=/opt/IBM/tivoli/netcool/rules
export NCHOME OMNIHOME NCO_PA NETCOOL_LICENSE_FILE NC_RULES_HOME
```

```
NETCOOL_LICENSE_FILE=27000@localhost # Points to flex license server
```

```
NC_RULES_HOME=
```

```
export NCHOME OMNIHOME NCO_PA NETCOOL_LICENSE_FILE
```

- Save the changes and exit gedit.

Process Activity is running on host1 and controlling the primary ObjectServer. If the changes are correct, you can use the script to stop and start process activity.

- Stop Process Activity as the **root** user.

```
/etc/init.d/nco stop
```

- Start Process Activity as the **root** user.

```
/etc/init.d/nco start
```

- Enable the auto-start.

```
cd /etc/init.d
chkconfig nco on
```

- Verify the auto-start.

```
chkconfig | grep nco
```

```
[root@host1 init.d]# chkconfig | grep nco
nco          0:off  1:off  2:on   3:on   4:on   5:on   6:off
```

11. Exit out of the **root** user back to the **netcool** user:

```
exit
```



Important: Make sure that you are the **netcool** user before proceeding.

In a subsequent exercise, you configure process activity to manage the Syslog and SNMP probes. Currently, the probes are running on host1. Stop the probes before proceeding.

12. Stop the Syslog probe.

```
pkkill nco_p_syslog
```

13. Stop the SNMP probe.

```
pkkill nco_p_mttrapd
```

Exercise 2 Configuring the Pluggable Authentication Module (PAM) on host2

The same steps as the previous exercise must also be performed on the host2 image.

1. Switch to the **host2** image.

2. Change to the **root** user:

```
su -  
Password: object00
```

3. Copy the existing file and rename it:

```
cd /etc/pam.d
```

```
cp system-auth netcool
```

```
cp: overwrite `netcool'? y
```

Enter **y** to replace the existing file.

It is the **netcool** file that provides the PAM authentication for Netcool components.

4. Examine the permissions on /etc/shadow:

```
ls -la /etc/shadow
```

```
-r--r--r-- 1 root root 1282 Aug 29 17:40 /etc/shadow
```

The file permissions on this file are changed to allow read access to non-root users. This change is necessary to allow the process activity daemon to run as a non-root user.

5. Install the process activity start file:

```
cd /opt/IBM/tivoli/netcool/omnibus/install/startup
```

```
./linux2x86install
```

Enter value for \$NCHOME [/opt/IBM/tivoli/netcool] :

This script copies a startup script into the /etc/init.d directory to enable you to automatically start and stop Netcool/OMNIbus processes.

It does this by:

Copying linux2x86/etc/rc.d/init.d/nco to /etc/init.d/nco

Running "/sbin/chkconfig --add nco"

Do you wish to continue (y/n) ? [y]

Name of the Process Agent Daemon [NCO_PA] : **HOST2_PA**

Should HOST1_PA run in secure mode (y/n) ? [y]

Enter value for environment variable NETCOOL_LICENSE_FILE

if required [27000@localhost] :

Scripts installed.



Important: Make sure to specify **HOST2_PA** as the Process Agent Daemon.

This step installs a script **/etc/init.d/nco**. It is this script that starts the process activity daemon at system start.

6. Configure the process activity daemon to run as the **netcool** user.

a. Change to the required directory:

```
cd /etc/init.d
```

b. Modify the script:

```
gedit nco
```

Locate the following lines:

```
if [ "$SECURE" = "Y" ] ; then
${OMNIHOME}/bin/nco_pad -name ${NCO_PA} -authenticate PAM -
secure > /dev/null 2> /dev/null
else
${OMNIHOME}/bin/nco_pad -name ${NCO_PA} -authenticate PAM >
/dev/null 2> /dev/null
fi
```

Change as follows:

```
if [ "$SECURE" = "Y" ]; then
  su - netcool -c "${OMNIHOME}/bin/nco_pad -name ${NCO_PA} -
  authenticate PAM -secure > /dev/null 2> /dev/null"
else
  su - netcool -c "${OMNIHOME}/bin/nco_pad -name ${NCO_PA} -
  authenticate PAM > /dev/null 2> /dev/null"
fi
```

The change is to add **su - netcool -c** to the beginning of the two lines, and to enclose each command in double quotations.

- Add the NC_RULES_HOME environment variable

Locate the following lines:

```
NETCOOL_LICENSE_FILE=27000@localhost # Points to flex license server
```

```
export NCHOME OMNIHOME NCO_PA NETCOOL_LICENSE_FILE
```

Change as follows:

```
NETCOOL_LICENSE_FILE=27000@localhost # Points to flex license server
NC_RULES_HOME=/opt/IBM/tivoli/netcool/rules
export NCHOME OMNIHOME NCO_PA NETCOOL_LICENSE_FILE NC_RULES_HOME
```

```
NETCOOL_LICENSE_FILE=27000@localhost # Points to flex license server
```

```
NC_RULES_HOME=
```

```
export NCHOME OMNIHOME NCO_PA NETCOOL_LICENSE_FILE
```

- Save the changes and exit gedit.

Process Activity is running on host2 and controlling the backup ObjectServer and gateway. If the changes are correct, you can use that script to stop and start process activity.

- Stop Process Activity as the **root** user.

```
/etc/init.d/nco stop
```

- Start Process Activity as the **root** user.

```
/etc/init.d/nco start
```

- Enable the auto-start.

```
cd /etc/init.d
chkconfig nco on
```

- Verify the auto-start.

```
chkconfig | grep nco
```

```
[root@host2 init.d]# chkconfig | grep nco
nco           0:off  1:off  2:on   3:on   4:on   5:on   6:off
```

11. Exit out of the **root** user back to the **netcool** user:

```
exit
```



Important: Make sure that you are the **netcool** user before proceeding.

Exercise 3 Configuring the process activity daemons

When you installed the Netcool/OMNIbus core components on host1 and host2, you ran the Initial Configuration Wizard. The wizard created the process activity configuration files for host1 and host2. On host1, the file contains an entry to start the primary ObjectServer. On host2, the file contains an entry to start the backup ObjectServer and gateway. You must make a few changes to this file on each system.

The first change is to configure user credentials that are used by the process agent. These credentials are required when the agent is running in *secure* mode.

1. Switch to **host1**.
2. Modify the process activity daemon configuration file.

- a. Change to the required directory:

```
cd $OMNIHOME/etc
```

- b. Edit the configuration file.

```
gedit nco_pa.conf
```

- c. Add authentication credentials.

Locate the following lines:

```
#  
# ROUTING TABLE  
#  
# 'user'      -  (optional) only required for secure mode PAD on target host  
#                  'user' must be member of UNIX group 'ncoadmin'  
# 'password'   -  (optional) only required for secure mode PAD on target  
host  
#                  use nco_pa_crypt to encrypt.  
nco_routing  
{  
    host 'host1.tivoli.edu' 'HOST1_PA'
```

Exercise 3 Configuring the process activity daemons

```
host 'host2.tivoli.edu' 'HOST2_PA'  
}
```

Change the two *host* lines as shown:

```
host 'host1.tivoli.edu' 'HOST1_PA' 'netcool' 'EDEAAPAIANFMCHCB'  
host 'host2.tivoli.edu' 'HOST2_PA' 'netcool' 'EDEAAPAIANFMCHCB'
```

The lines that require modification are the user name (netcool), and *encrypted* password (EDEAAPAIANFMCHCB).



Hint: You encrypt the password value by running the utility nco_pa_crypt.

- d. Save the changes and exit gedit.

You must make the same changes on host2.

3. Switch to **host2**.
4. Modify the process activity daemon configuration file.

- a. Change to the required directory:

```
cd $OMNIHOME/etc
```

- b. Edit the configuration file.

```
gedit nco_pa.conf
```

- c. Add authentication credentials.

Locate the following lines:

```
#  
# ROUTING TABLE  
#  
# 'user'      - (optional) only required for secure mode PAD on target host  
#                 'user' must be member of UNIX group 'ncoadmin'  
# 'password'   - (optional) only required for secure mode PAD on target  
host  
#                 use nco_pa_crypt to encrypt.  
nco_routing  
{  
    host 'host1.tivoli.edu' 'HOST1_PA'  
    host 'host2.tivoli.edu' 'HOST2_PA'  
}
```

Change the two *host* lines as shown:

```
host 'host1.tivoli.edu' 'HOST1_PA' 'netcool' 'EDEAAPAIANFMCHCB'  
host 'host2.tivoli.edu' 'HOST2_PA' 'netcool' 'EDEAAPAIANFMCHCB'
```

The lines that require modification are the user name (netcool), and *encrypted* password (EDEAAPAIANFMCHCB).



Hint: You encrypt the password value by running the utility nco_pa_crypt.

- d. Save the changes and exit gedit.

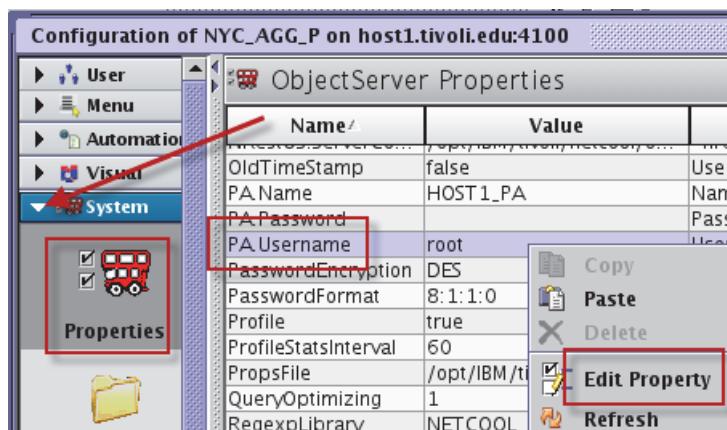


Important: Remain on the **host2** image.

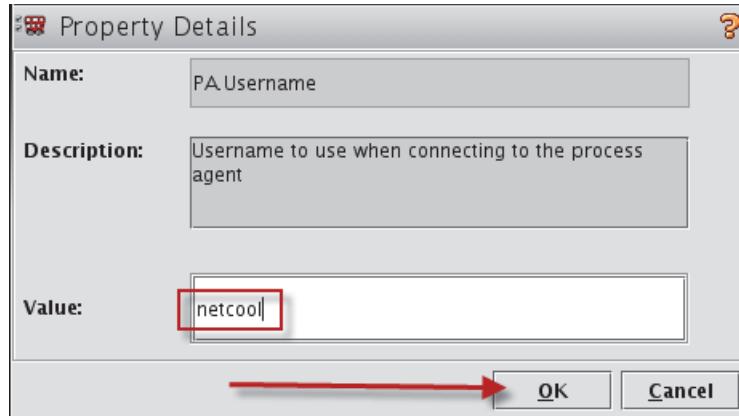
Exercise 4 Modifying the ObjectServer process activity credentials

In the previous steps, you modified the process activity daemon configuration on each server and added a user ID (*netcool*), and encrypted password to the routing section. This modification configures the daemon to require a user ID, and password whenever the ObjectServer connects to it. The following steps configure the user ID, and password values as property settings in each ObjectServer.

1. Modify the primary ObjectServer.
 - a. Start the Netcool/OMNIbus administrator utility:
nco_config &
 - b. Connect to the **NYC_AGG_P** ObjectServer as the *root* user.
 - c. Expand **System** and click **Properties**.
 - d. Click **PA.Username** to select it, right-click and select **Edit Property**.

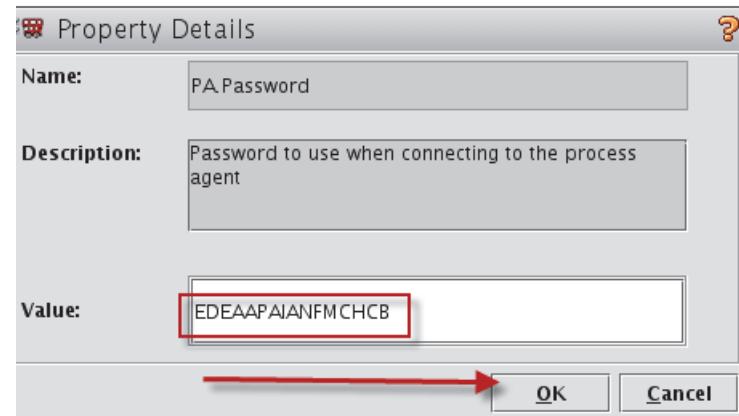


- e. Change **Value** to *netcool* and click **OK**.



- f. Click **PA.Password** to select it, right-click and select **Edit Property**.

- g. Change the Value to **EDEAAPAIAJANFMCHCB** and click **OK**.



The correct values are listed as in the following screen capture.

Name	Value
OldTimeStamp	false
PA.Name	HOST1_PA
PA.Password	EDEAAPAIAJANFMCHCB
PA.Username	netcool
PasswordEncryption	DES
PasswordFormat	8:1:1:0

2. Modify the backup ObjectServer.

Repeat the previous steps to set the same values in the NYC_AGG_B ObjectServer.



Important: Use the instance of the Administrator GUI. Connect to the NYC_AGG_B ObjectServer.

The proceeding steps demonstrate how to modify an ObjectServer configuration remotely. You used the Administrator GUI that is running on the host2 image to modify the NYC_AGG_P ObjectServer that is running on the host1 image.

Leave the Administration GUI open. You use it again shortly.

Exercise 5 Working with the process activity daemon

In several previous exercises, you used commands to cause the process activity daemon to stop processes, start processes and display the status of managed processes. These commands can be used to connect to and control a remote process agent.

You are currently on the host2 image.

1. Display the status of the process agents.

- a. Display the status of the host1 agent as follows:

```
nco_pa_status -server HOST1_PA -password object00
```

[netcool@host2 etc]\$ nco_pa_status -server HOST1_PA -password object00					
Service Name	Process Name	Hostname	User	Status	PID
Core	MasterObjectServer	host1.tivoli.edunetcool		RUNNING	6489

- b. Display the status of the host2 agent as follows:

```
nco_pa_status -server HOST2_PA -password object00
```

[netcool@host2 etc]\$ nco_pa_status -server HOST2_PA -password object00					
Service Name	Process Name	Hostname	User	Status	PID
Core	BackupObjectServer	host2.tivoli.edunetcool		RUNNING	5976
	BackupGateway	host2.tivoli.edunetcool		RUNNING	5977

Just by changing the name of agent on the command line, it is possible to connect to any remote agent. The same technique works for all commands, including these examples:

```
nco_pa_status
nco_pa_stop
nco_pa_shutdown
```

In addition to the command-line interface, the process agent also supports access through a graphical interface. This interface is referred to as Visual PA and is access from the Administrator GUI.

2. Return to the Administrator GUI.

- Locate the Process Agents box.

Name	Host	Port	SSL	Ru...	Pe...	De...	Save ...
HOST1_PA	host1.tivoli.edu	4200	X	-	-	-	X fa...
HOST2_PA	host2.tivoli.edu	4200	X	-	-	-	X fa...

- Click **HOST1_PA**, right-click, and select **Connect As...**.

- Enter *netcool* for the user, *object00* for the password, and click **OK**

Username:	netcool
Password:	*****
<input type="checkbox"/> Always use for this connection <input type="checkbox"/> Use as default	
<input type="button" value="OK"/> <input type="button" value="Cancel"/>	

Important: The user ID is a Linux user that belongs to the *ncoadmin* group.

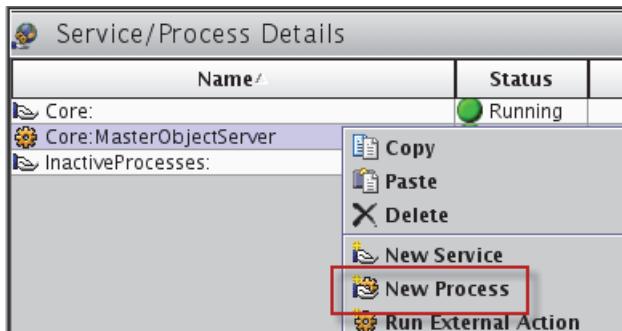
- Observe the list of processes and their status.

Name	Status	Host	ID
Core:	Running		
Core:MasterObjectServer	Running	host1.tivoli.edu	6489
InactiveProcesses:	Stopped		

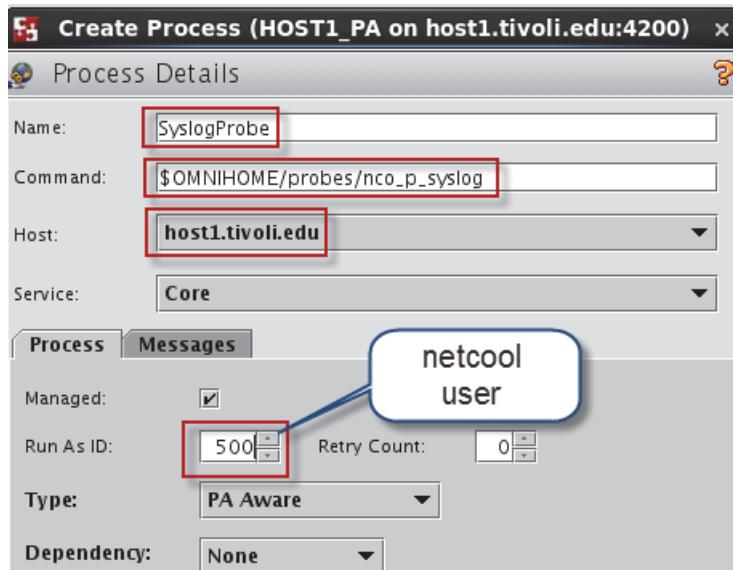
Two probes are installed on the host1 image that are not under control by process activity. You can manually edit the nco_pa.conf file and add entries. Or you can dynamically add them through the Visual PA interface.

7. Add the Syslog probe as follows:

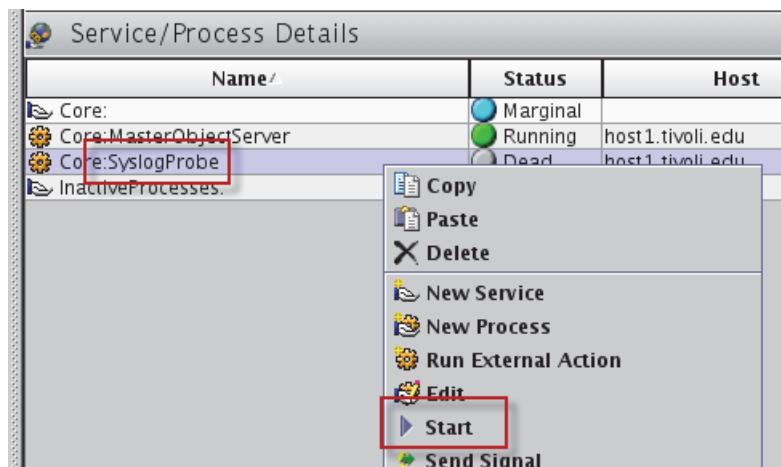
- Right-click and select **New Process**.



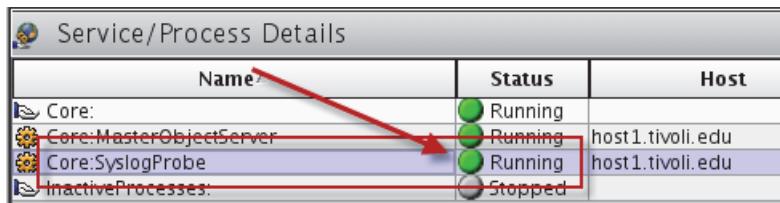
- Complete the entries as shown and click OK.



- Click **SyslogProbe** to select it, right-click and select **Start**.



- d. Verify the status.

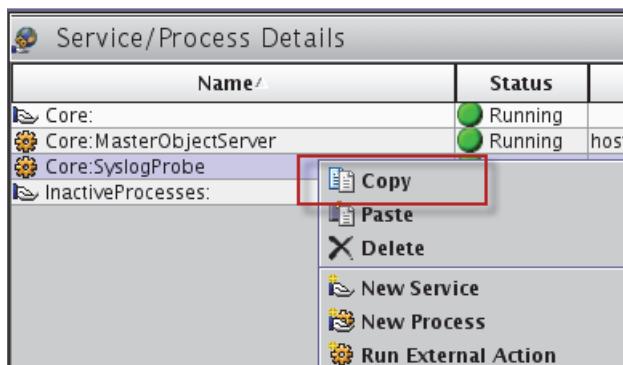


Name	Status	Host
Core:	Running	
Core:MasterObjectServer	Running	host1.tivoli.edu
Core:SyslogProbe	Running	host1.tivoli.edu
InactiveProcesses:	Stopped	

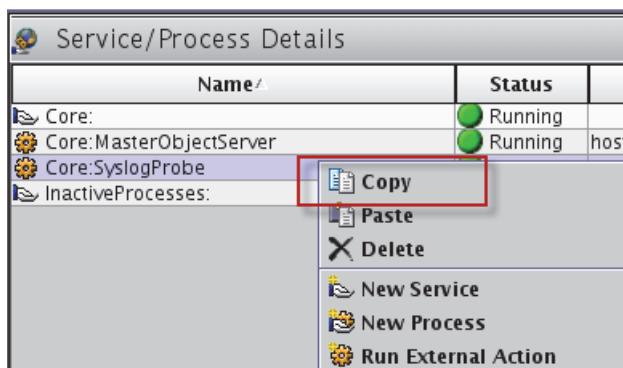
After a short time, the status changes to RUNNING. The Syslog probe is now running on the host1 image, and the process agent is configured to manage that component. You accomplished that from the host2 image.

8. Add the SNMP probe as follows:

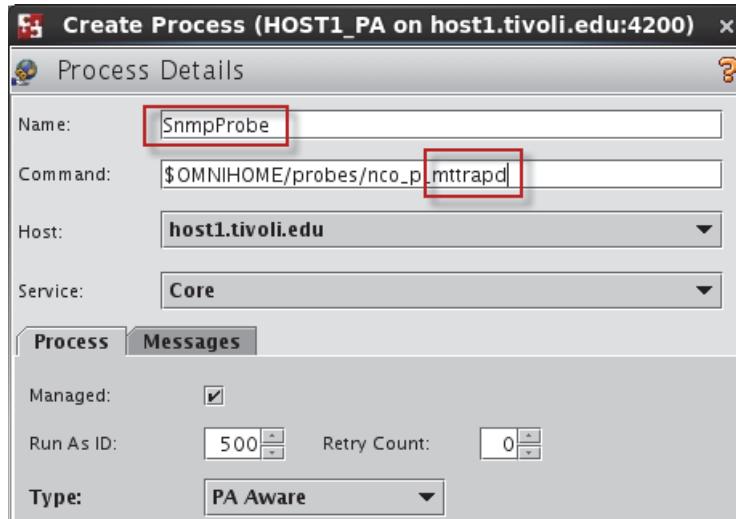
- a. Click SyslogProbe to select it, right-click and select **Copy**.



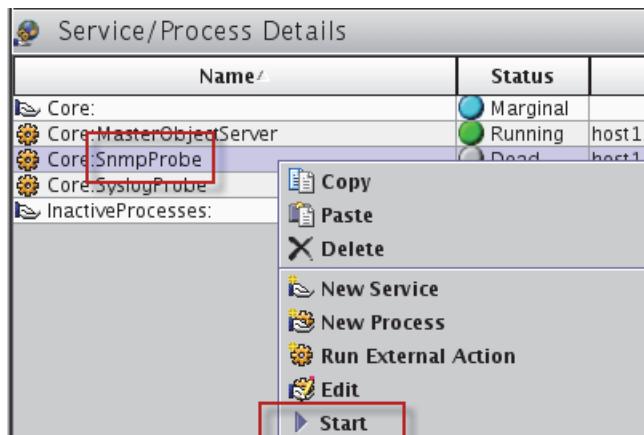
- b. Right-click and select **Paste**.



- c. Change the entries as shown and click **OK**.



- d. Click **SnmpProbe** to select it, right-click and select **Start**.



- e. Verify the status.

Name	Status	Host
Core:	Running	
Core:MasterObjectServer	Running	host1.tivoli.edu
Core:SnmpProbe	Running	host1.tivoli.edu
Core:SyslogProbe	Running	host1.tivoli.edu
InactiveProcesses:	Stopped	

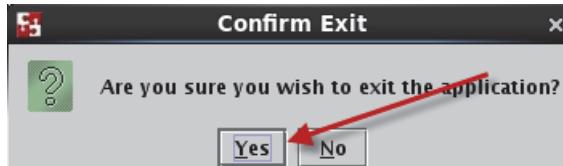
After a short time, the status changes to RUNNING. The SNMP probe is now running on the host1 image, and the process agent is configured to manage that component.

9. Exit the Administrator GUI.

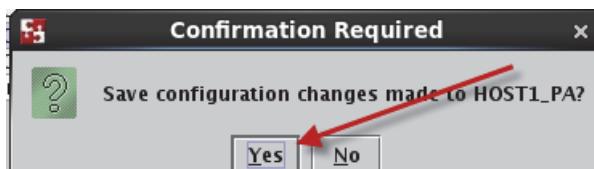
- a. Click **File** and select **Exit**.



- b. Click **Yes** to confirm exit.

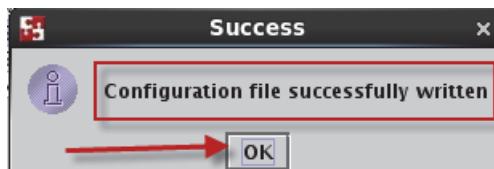


- c. Click **Yes** to confirm saving the changes.



Important: The two windows look similar, but they control different outcomes. The confirmation on the second window causes the changes to be written to the nco_pa.conf file on **host1**.

- d. Click **OK**.



10. Switch to the **host1** image.

11. Examine the process activity configuration file.

```
cd $OMNIHOME/etc
more nco_pa.conf
```

12. Page through the file and verify that the entries for the Syslog and SNMP probes are present.

Exercise 6 Verifying the auto-start configuration

The last step is to restart each server and verify that all components start as expected.

1. Switch to the **host1** image.

2. Change to the **root** user.

```
su -
Password: object00
```

3. Restart the server.

```
init 6
```

While host1 restarts, do the same on host2.

4. Switch to the **host2** image.

5. Change to the *root* user.

```
su -  
Password: object00
```

6. Restart the server.

```
init 6
```

Wait for the **host1** server to restart and display the login page.

7. Log in to **host1** as the **netcool** user.

8. Open a terminal window.

9. Verify the status of the Netcool/OMNIbus components:

```
nco_pa_status -server HOST1_PA -password object00
```

Service Name	Process Name	Hostname	User	Status	PID
Core	MasterObjectServer	host1.tivoli.edunetcool		RUNNING	2054
	SyslogProbe	host1.tivoli.edunetcool		RUNNING	2055
	SnmpProbe	host1.tivoli.edunetcool		RUNNING	2056

Verify that all required processes are running.

Wait for the **host2** server to restart and open the login page.

10. Log in to **host2** as the **netcool** user.

11. Open a terminal window.

12. Verify the status of the Netcool/OMNIbus components:

```
nco_pa_status -server HOST2_PA -password object00
```

Service Name	Process Name	Hostname	User	Status	PID
Core	BackupObjectServer	host2.tivoli.edunetcool		RUNNING	2265
	BackupGateway	host2.tivoli.edunetcool		RUNNING	2266

Verify that all required processes are running.

The summary of accomplishments for this unit includes the following tasks:

- Process activity on both servers is configured to manage all core components.
- Process activity on both servers is configured to run as the **netcool** user.
- Process activity on both servers is configured to start at system restart.
- You verified process status by running the process activity command-line utility.
- You verified process status with Visual PA.



10 Event archiving exercises

The exercises in this unit teach you how to configure event archiving, and to install Tivoli Common Reporting.

Exercise 1 Installing Tivoli Common Reporting

Tivoli® Common Reporting is bundled in several Tivoli products. When bundled with a product, Tivoli Common Reporting installs as a part of the product installation, usually associated with a dashboard component. Tivoli Common Reporting is not bundled with Netcool®/OMNIbus. The installation file must be downloaded, and installed separately.



Important: You install Tivoli Common Reporting on the **host2** image.

1. Expand the installation files.

- a. Change to the required directory:

```
cd /software/tcr
```

- b. Expand the installation file:

```
gunzip ITCR_3.1_FOR_LINUX_ML.tar.gz  
tar -xvf ITCR_3.1_FOR_LINUX_ML.tar
```



Note: It takes approximately 10 minutes to expand the installation files.

2. Create the DB2® report content store database.

- a. Change to the required directory:

```
cd /software/tcr/TCRInstaller/ContentStoreDatabase
```

- b. Generate the SQL files:

```
./TCR_generate_content_store_db2_definition.sh TCR_DB db2inst1
```

The `tcr_create_db2_cs.sql` script was created



Important: TCR_DB is the name of the report content store database. It can be any valid DB2 database name. The name itself is not relevant. The text *db2inst1* identifies the DB2 user name that owns the content store database.

The output from this command is an SQL file. You import that SQL file into a DB2 command line to create the content store database structure.



Important: You must import the SQL file as a DB2 user that has authority to create DB2 databases and tables. For this exercise that user is **db2inst1**.

- c. Switch to the **db2inst1** user.

```
su - db2inst1  
Password: object00
```

- d. Import the SQL file and create the content store:

```
cd /software/tcr/TCRInstaller/ContentStoreDatabase
```

```
db2 -vtf tcr_create_db2_cs.sql  
. . .  
CONNECT RESET  
DB20000I The SQL command completed successfully.
```



Note: You can ignore the error messages.

- e. Verify that the database is created.

```
db2 connect to TCR_DB
```

Database Connection Information

```
Database server      = DB2/LINUXX8664 10.5.3  
SQL authorization ID = DB2INST1  
Local database alias = TCR_DB
```

- f. Exit the **db2inst1** user back to the **netcool** user.

```
exit
```

3. Install Tivoli Common Reporting.



Important: Tivoli Common Reporting v3.1 cannot be installed with the IBM Installation Manager utility. You must use the installation utility that is bundled with Tivoli Common Reporting.

- a. Change to the required directory.

```
cd /software/tcr/TCRInstaller
```

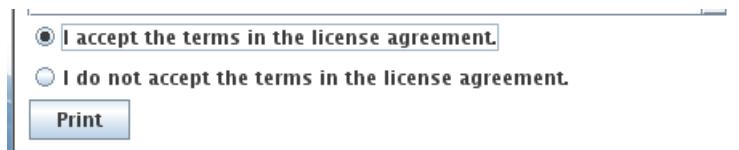
- b. Start the installation utility:

```
./install.sh
```

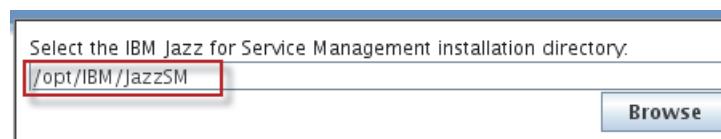
- c. Leave the language set to English and click **OK**.

- d. At the welcome screen, click **Next**.

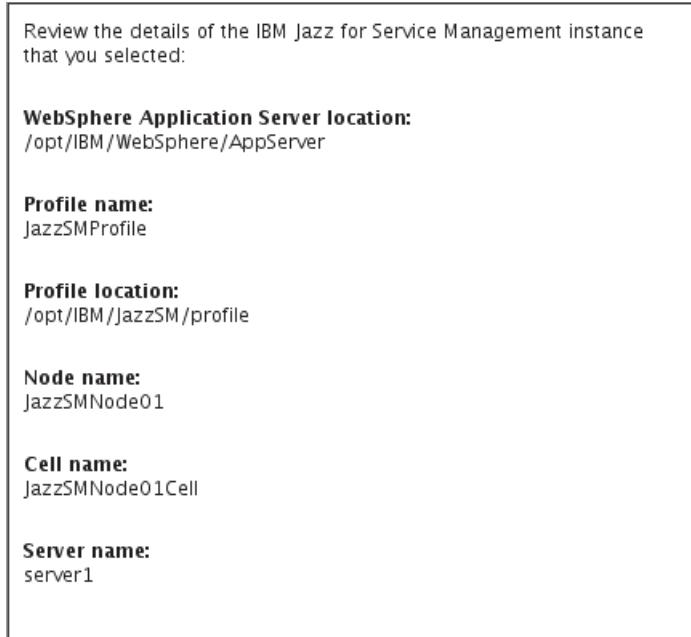
- e. Accept the license agreement and click **Next**.



- f. Leave the default setting for the location of Jazz for Service Management and click **Next**.



- g. Review the details for Jazz for Service Management and click **Next**.



- h. Enter **smadmin** for the user and **object00** for the password, and click **Next**.

Provide the administrative user ID and password used to log into the IBM Jazz for Service Management WebSphere profile.

User ID:
smadmin

Password:
object00

- i. Enter the connection information for the content store database and click **Next**.

Hostname: host2.tivoli.edu

Port: 50000

Database name: TCR_DB

Database owner: db2inst1

Password: object00

Provide the DB2 connection details for IBM Cognos Content database.

Hostname / IP:
host2.tivoli.edu

Port number:
50000

Database name:
TCR_DB

Database owner:
db2inst1

Database owner password:
object00

- j. Review the installation summary and click **Install**.

You are about to install IBM Tivoli Common Reporting 3.1

Installation location:
/opt/IBM/JazzSM/reporting

Content Store database hostname or IP address:
host2.tivoli.edu

Content Store database port:
50000

Content Store database name:
TCR_DB

Owner of the Content Store database:
db2inst1

Required disk space:
5,461 MB

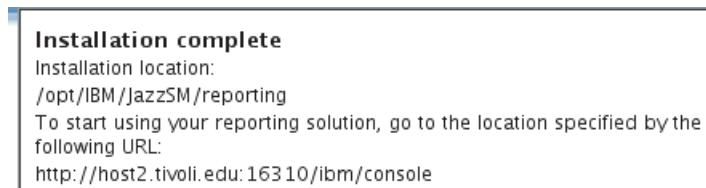
Available disk space:
16,622 MB

Installation log file location:
/home/netcool/TCR31/installMessage00.log



Hint: An installation on most servers runs approximately 45 minutes.

- k. Verify that the installation is successful and click **Done**.



Hint: You can find the installation log file here:

/home/netcool/TCRInstaller-00.log

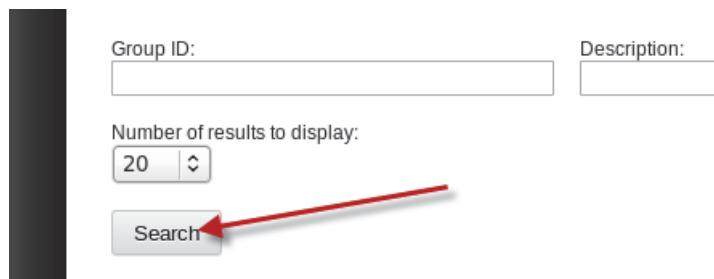
4. Update groups to allow access to Tivoli Common Reporting.

Access to Tivoli Common Reporting requires a specific Dashboard Application Services Hub role. The installation process adds that role to the **smadmin** user. Add the role to other user groups.

- Start a Firefox browser.
- Log in as user **smadmin**.
- Click the icon and select **Group Roles**.



- Click **Search**.



- Click **Netcool_Admin**.

Group Name	Roles
Netcool_Admin	ncw_gauges_editor, ncw_admin, ncw_dashboard_editor, iscadmins, netcool_rw
Netcool_User	ncw_user, ncw_gauges_viewer, netcool_rw

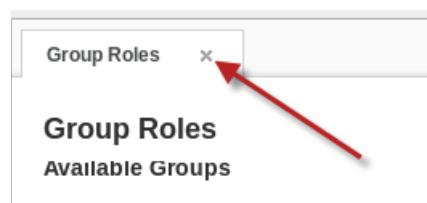
- f. Scroll to the bottom of the page, select **tcrPortalOperator**, and click **Save**.



- g. Repeat the previous steps to add **tcrPortalOperator** to the **Netcool_User** group.

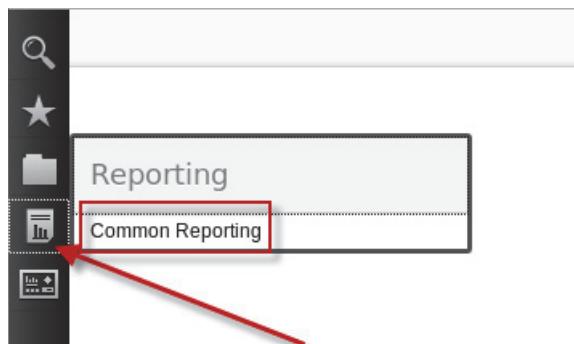
Group Name	Roles
Netcool_Admin	ncw_gauges_editor, ncw_admin, tcrPortalOperator, ncw_dashboard_editor, iscadmins, netcool_rw
Netcool_User	ncw_user, tcrPortalOperator, ncw_gauges_viewer, netcool_rw

- h. Click the X to close the Group Roles page.



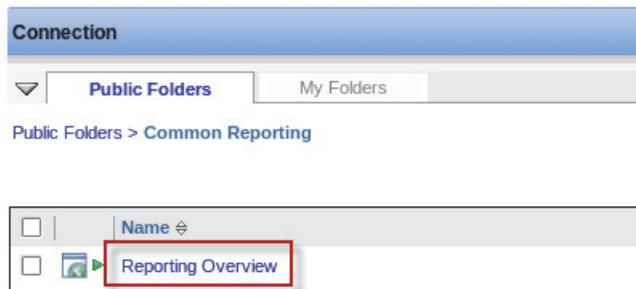
5. Verify basic Tivoli Common Reporting function.

- a. Click the icon and select **Common Reporting**.



- b. Click the report package **Common Reporting**.

- c. Click **Reporting Overview** to run the report.



- d. Leave the default settings on the prompt page and click **Finish**.

The completed report opens.



This report lists all of the report templates that currently exist in the Tivoli Common Reporting report store database. Currently, there is only one, which is this report. This report verifies that Tivoli Common Reporting is able to generate a basic report.

6. Modify environment settings for the **netcool** user.

The Cognos reporting engine for Tivoli Common Reporting requires access to various DB2 library files. This change is necessary only if you are creating reports from a DB2 data source. The Cognos reporting engine is started when Dashboard Application Services Hub starts. The engine runs as the same user that starts Dashboard Application Services Hub. For the class room environment that is the *netcool* user. The simplest way to make the library files available to the Cognos reporting engine is to modify the *netcool* user environment.

```
cd /home/netcool
```

```
gedit .bashrc
```

Scroll down in the file and remove the comment character from the following line:

```
#source /home/db2inst1/sql1ib/db2profile
```

The modified line appears as follows:

```
source /home/db2inst1/sql1ib/db2profile
```

Save the file and exit gedit.



Note: The file **/home/db2inst1/sql1ib/db2profile** contains a definition for the **LD_LIBRARY_PATH** environment variable. This variable definition is what implements the required environment.

7. Log out of Dashboard Application Services Hub.
8. Close the Firefox browser.
9. Restart Dashboard Application Services Hub.
 - a. Change to the **root** user.

```
su -  
Password: object00
```

- b. Stop the server.

```
/etc/init.d/jazz stop
```

- c. Start the server.

```
/etc/init.d/jazz start
```

You restart Dashboard Application Services Hub to incorporate the environment setting changes.

Exercise 2 Creating the event archive database structure

Netcool/OMNIbus archives events to a customer provided database. For this workshop, you use DB2. The DB2 software is already installed on the system. You use a supplied SQL file to create the database structure that archives the event records. The SQL is bundled in a package that contains the gateway configuration files. That package must be installed. In addition, the gateway that is

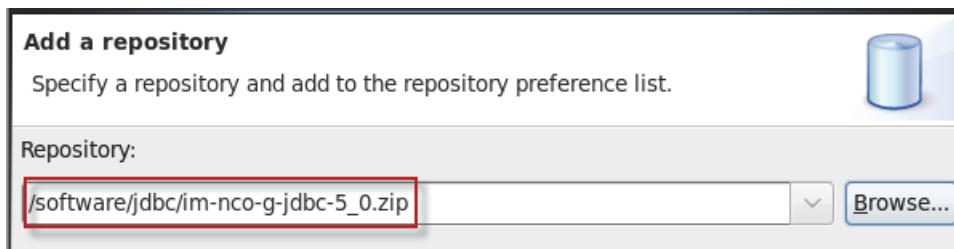
used to archive the event records must also be installed. They can both be installed at the same time.

1. Install the gateway components.

- a. Start IBM Installation Manager:

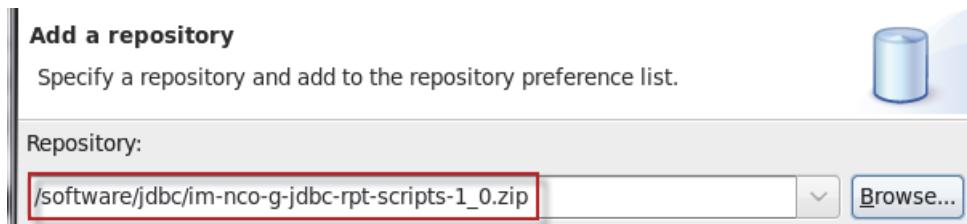
```
cd /home/netcool/IBM/InstallationManager/eclipse  
./IBMMIM
```

- b. Add the gateway installation compressed file as a repository.



Note: It is not necessary to expand the compressed file.

- c. Add the gateway scripts installation compressed file as a repository.



- d. Verify that the other repositories are not selected and click **OK**.

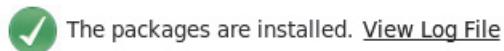
Repositories:	
Location	Connection
<input type="checkbox"/> /tmp/jazz_install/JazzSMRepository/disk1/diskTag.inf	
<input type="checkbox"/> /tmp/jazz_install/WASRepository/disk1/diskTag.inf	
<input type="checkbox"/> /software/omnibus/OMNIBusRepository/repository.config	
<input checked="" type="checkbox"/> /software/jdbc/im-nco-g-jdbc-5_0.zip	
<input checked="" type="checkbox"/> /software/jdbc/im-nco-g-jdbc-rpt-scripts-1_0.zip	

- e. Click **Install**.

- f. Select the two packages to install and click **Next**.

Installation Packages	Status
<input checked="" type="checkbox"/> Netcool/OMNIbus Gateway nco-g-jdbc <input checked="" type="checkbox"/> Version 1.5.0.0	Will be installed
<input checked="" type="checkbox"/> Netcool/OMNIbus Gateway nco-g-jdbc-reporting-scripts <input checked="" type="checkbox"/> Version 1.1.0.0	Will be installed

- g. Accept the license agreement and click **Next**.
- h. Leave the option selected to use the existing package group and click **Next**.
- i. Leave the features selected and click **Next**.
- j. Review the installation summary and click **Install**.
- k. Verify that the installation is successful and click **Finish**.



The following packages were installed:

▽	IBM Tivoli Netcool OMNIbus
	Netcool/OMNIbus Gateway nco-g-jdbc 1.5.0.0
	Netcool/OMNIbus Gateway nco-g-jdbc-reporting-scripts 1.1.0.0

- I. Click **File** and select **Exit** to close IBM Installation Manager.
2. Create the DB2 structure.
DB2 is running as the **db2inst1** user. You must use this user to create the database structure.
 - a. Change to the **db2inst1** user:

```
su - db2inst1  
Password: object00
```
 - b. Change to the required directory:

```
cd /opt/IBM/tivoli/netcool/omnibus/gates/reporting/db2
```

 **Hint:** The reporting directory is created when the gateway package is installed.

- c. Import the SQL file.

There are three SQL files in this directory: **db2.drop_reporting.sql**, **db2.reporting.sql**, and **db2.reporting.old.sql**. You can use the file that contains the name *drop* to remove the DB2 database structure if necessary. The file that contains the name *old* includes the SQL commands to create the REPORTER database. The other file, without drop in the name, does not contain commands to create the database. Many companies have specific standards with regards to how databases are defined. For those customers, the commands in the *old* file are most likely not adequate. In that case, they can create the database that is based on their standards. They would then use **db2.reporting.sql** to add the table structure to the database. For this exercise, you use **db2.reporting.old.sql**.

```
db2 -td@ -vf db2.reporting.old.sql
```

 **Note:** This command runs for several minutes.

```
COMMIT WORK
```

DB20000I The SQL command completed successfully.

3. Verify the DB2 structure.

The SQL file creates a database (REPORTER), and numerous tables.

a. Connect to the REPORTER database:

```
db2 connect to reporter
```

Database Connection Information

```
Database server      = DB2/LINUXX8664 10.5.3
SQL authorization ID = DB2INST1
Local database alias = REPORTER
```



Hint: DB2 is not case-sensitive. You can use upper or lowercase characters for any DB2 object.

b. Verify the table structure:

```
db2 list tables
```

Table/View	Schema	Type	Creation time
REPORTER_CLASSES	DB2INST1	T	2014-10-01-17.54.12.151628
REPORTER_CONVERSIONS	DB2INST1	T	2014-10-01-17.54.12.268367
REPORTER_DETAILS	DB2INST1	T	2014-10-01-17.54.09.595508
REPORTER_GROUPS	DB2INST1	T	2014-10-01-17.54.11.970445
REPORTER_JOURNAL	DB2INST1	T	2014-10-01-17.54.10.911528
REPORTER_MEMBERS	DB2INST1	T	2014-10-01-17.54.12.073987
REPORTER_NAMES	DB2INST1	T	2014-10-01-17.54.11.828164
REPORTER_STATUS	DB2INST1	T	2014-10-01-17.54.11.121974
REP_AUDIT	DB2INST1	V	2014-10-01-17.54.13.372712
REP_AUDIT_ACK	DB2INST1	T	2014-10-01-17.54.11.630816
REP_AUDIT_OWNERGID	DB2INST1	T	2014-10-01-17.54.11.415751
REP_AUDIT_OWNERUID	DB2INST1	T	2014-10-01-17.54.11.339441
REP_AUDIT_SEVERITY	DB2INST1	T	2014-10-01-17.54.11.499096
REP_REFERENCE_DATE	DB2INST1	V	2014-10-01-17.54.13.125819
REP_SEVERITY_TYPES	DB2INST1	T	2014-10-01-17.54.12.414987
REP_TIME_PERIODS	DB2INST1	T	2014-10-01-17.54.12.629815
STATUS_VW	DB2INST1	V	2014-10-01-17.54.13.319413

17 record(s) selected.

Verify that 17 tables and views are created.

c. Exit out of the db2inst1 user back to the netcool user.

```
exit
```



Important: Make sure that you are the netcool user before proceeding.

Exercise 3 Configuring the database gateway

You use the JDBC gateway to archive Netcool/OMNIbus event records to the DB2 REPORTER database. You installed the gateway in the previous exercise.

1. Add the gateway to the Netcool/OMNIbus communications file.

The gateway must have a name. For this exercise, use JDBC_GATE. That name must be added to the Netcool/OMNIbus communications file, the *interfaces* file.

- a. Run the **Server Editor** utility:

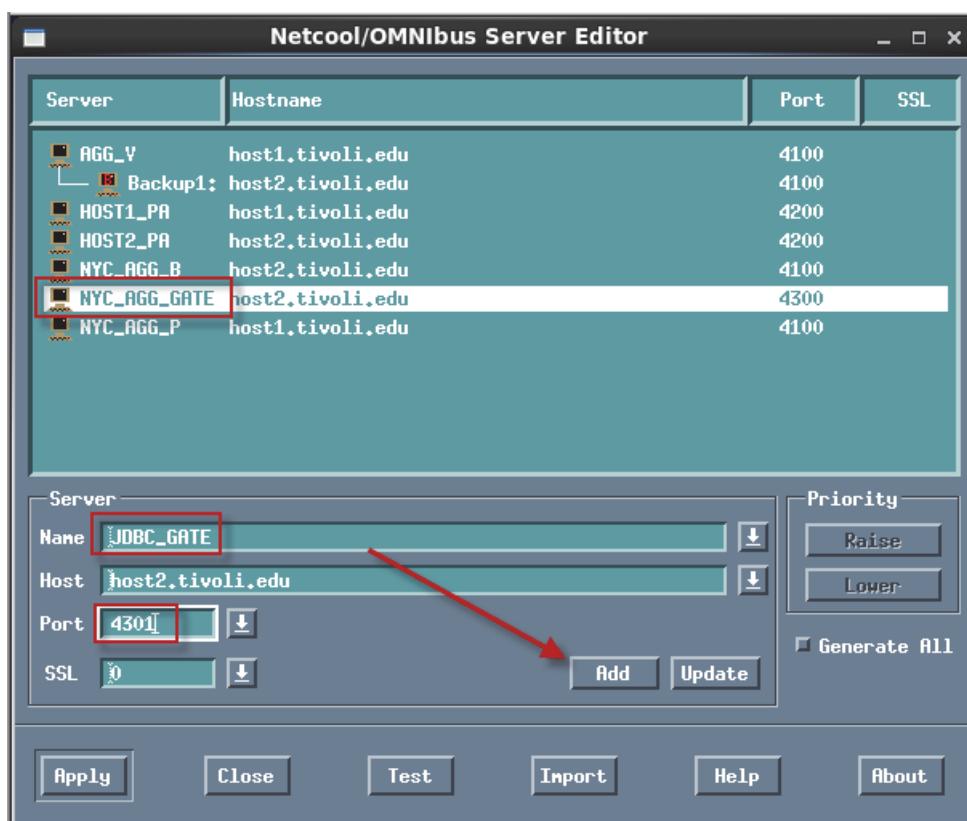
```
nco_xigen &
```

- b. Click the entry **NYC_AGG_GATE** to select it.

- c. Change the Name to **JDBC_GATE**.

- d. Change the Port to **4301**

- e. Click **Add**.



Important: Make sure you click **Add** because you want to create a new entry. If you click **Update**, you *change* the entry for NYC_AGG_GATE to JDBC_GATE.

- f. Click **Apply** and click **Close**.

Server	Hostname	Port
AGG_V	host1.tivoli.edu	4100
Backup1:	host2.tivoli.edu	4100
HOST1_PA	host1.tivoli.edu	4200
HOST2_PA	host2.tivoli.edu	4200
JDBC_GATE	host2.tivoli.edu	4301
NTL_HGG_B	host2.tivoli.edu	4100
NYC_AGG_GATE	host2.tivoli.edu	4300
NYC_AGG_P	host1.tivoli.edu	4100

Verify that the entry for JDBC_GATE is listed.

2. Configure the gateway.

The gateway is configured with several text files. The installation process creates these files in a specific directory. Copy the default files from that location to **\$OMNIHOME/etc**, and rename the files to include the gateway name, JDBC_GATE.

- a. Change to the required directory:

```
cd $OMNIHOME/gates/jdbc
```

- b. Copy and rename the files:

```
cp reporting.jdbc.map $OMNIHOME/etc/JDBC_GATE.map
cp reporting.G_JDBC.props $OMNIHOME/etc/JDBC_GATE.props
cp jdbc.rdrwtr.tblrep.def $OMNIHOME/etc/JDBC_GATE.rdrwtr.tblrep.def
cp jdbc.startup.cmd $OMNIHOME/etc/JDBC_GATE.startup.cmd
```

- c. Verify that the files are correctly renamed:

```
cd $OMNIHOME/etc
ls -1 JDBC_GATE.*
```

```
JDBC_GATE.map
JDBC_GATE.props
JDBC_GATE.rdrwtr.tblrep.def
JDBC_GATE.startup.cmd
```

- d. Examine the map file:

```
more JDBC_GATE.map
```

This file defines how the gateway maps a column in an ObjectServer table to the corresponding column in a table in the REPORTER database. The column names on the left refer to columns in a DB2 table. The column names on the right are the column names in the ObjectServer table. This file is typically modified when you add custom columns to the ObjectServer event record (alerts.status). No modifications are required for this exercise.



Note: You did add two custom columns to the ObjectServers. However, these columns do not contain data of any significance to historical reporting.

e. Modify the property file:

You must modify the property file to define things like: ObjectServer name and DB2 database user and password.



Hint: When modifying the file, be sure to place all changes on the end of the file.

```
gedit JDBC_GATE.props
```

Scroll to the bottom of the file. There are numerous properties values already defined. Some of the property values must be modified, and more lines must be added.

Modify the following *existing* lines as shown:

```
# JDBC Connection properties
Gate.Jdbc.Driver: 'com.ibm.db2.jcc.DB2Driver' # STRING (JDBC Driver)
Gate.Jdbc.Url: 'jdbc:db2://host2.tivoli.edu:50000/reporter' # STRING (JDBC connection URL)
Gate.Jdbc.Username: 'db2inst1' # STRING (JDBC username)
Gate.Jdbc.Password: 'object00' # STRING (JDBC password)
Gate.Jdbc.ReconnectTimeout: 30 # INTEGER (JDBC database reconnection timeout)
Gate.Jdbc.InitializationString: '' # STRING (JDBC connection initialization string)
```

Comment out the following two existing lines:

```
# ObjectServer Connection properties
#Gate.RdrWtr.Username: 'root' # STRING ([RdrWtr] Name of the user to connect as.)
#Gate.RdrWtr.Password: '' # STRING ([RdrWtr] Password of the user to connect as.)
```

An ObjectServer user name and password is required only if the ObjectServer is running in secure mode.

Add the following lines:

```
# New lines
# Log file name
MessageLog          : '$OMNIHOME/log/JDBC_GATE.log'
# Gateway name
Name                : 'JDBC_GATE'
# Property file name
```

```

PropsFile           : '$OMNIHOME/etc/JDBC_GATE.props'
# Map file name
Gate.MapFile       : '$OMNIHOME/etc/JDBC_GATE.map'
# Name of ObjectServer - this is the virtual ObjectServer
Gate.RdrWtr.Server : 'AGG_V'
# Table replication file name
Gate.RdrWtr.TblReplicateDefFile :
'$OMNIHOME/etc/JDBC_GATE.rdrwtr.tblrep.def'
# Startup command file name
Gate.StartupCmdFile : '$OMNIHOME/etc/JDBC_GATE.startup.cmd'
# Description name - this value appears in the list of ObjectServer
connections
Gate.RdrWtr.Description : 'JDBC Gateway'

```



Hint: Each of the new property statements is located in the upper part of the file. You can copy the property value from the top of the file and paste the line. Remove the comment character, and modify the value.

- f. Save the changes and exit the gedit utility.
- g. Modify the startup command file.

This file provides the option to configure the gateway to run one or more commands when the gateway starts. There are several tables in the REPORTER database that are *relatively* static. These tables are created from ObjectServer tables that contain user names, group names, conversions, and classes. There are commands in the default startup file to transfer these tables. The commands are commented out. Remove the comment character to configure the gateway to transfer copies of these tables at start.

```
gedit JDBC_GATE.startup.cmd
```

Remove the comment character (#) from the beginning of each TRANSFER command as follows:

```

TRANSFER FROM 'alerts.conversions' TO 'REPORTER_CONVERSIONS' DELETE USING
TRANSFER_MAP ConversionsMap;
TRANSFER FROM 'alerts.objclass' TO 'REPORTER_CLASSES' DELETE USING
TRANSFER_MAP ObjectClassesMap;
TRANSFER FROM 'master.groups' TO 'REPORTER_GROUPS' DELETE USING TRANSFER_MAP
GroupsMap;
TRANSFER FROM 'master.members' TO 'REPORTER_MEMBERS' DELETE USING
TRANSFER_MAP MembersMap;
TRANSFER FROM 'master.names' TO 'REPORTER_NAMES' DELETE USING TRANSFER_MAP
NamesMap;

```

- h. Save the changes and exit the gedit utility.
- i. Modify the table replication file.

When an ObjectServer table changes, that is, inserted, updated, or deleted, the gateway *replicates* the change to the corresponding DB2 table. The table replication file defines which ObjectServer tables that the gateway replicates. There are two: alerts.status, and alerts.journals.

In previous units, you configured various ObjectServer components to implement a technique for deleting event records. The final change that is required to complete this implementation involves a change to the gateway configuration. By adding a command to the table replication file, you configure the gateway to modify a column in the ObjectServer event record. The column is modified *after* the gateway writes the record to DB2. This column, Archived_Flag, is referenced in an ObjectServer trigger that is created previously. The trigger removes any event record where Delete_Flag=1, set by a custom tool, and Archived_Flag=1, which is set by the gateway.

```
gedit JDBC_GATE.rdrwtr.tblrep.def
```

Modify the file as shown:

```
REPLICATE ALL FROM TABLE 'alerts.status'  
    USING MAP 'StatusMap'  
    AFTER IDUC DO 'Archived_Flag=1';
```

Remove the semicolon from the end of the line that contains USING MAP. Add the line that contains AFTER IDUC DO.

- j. Save the changes and exit the gedit utility.
3. Install the DB2 JDBC driver files.

```
cd /opt/ibm/db2/v10.5/java  
cp db2jcc.jar $OMNIHOME/gates/java  
cp db2jcc_license_cu.jar $OMNIHOME/gates/java
```

4. Start the gateway.

```
nco_g_jdbc -name JDBC_GATE &
```

Wait a short time, and verify that the gateway is running. If the gateway fails, examine the log file for issues:

```
more $OMNIHOME/log/JDBC_GATE.log
```



Hint: One of the primary reasons for the gateway to fail to start is an issue with the DB2 connection information. If the gateway fails to start, examine the gateway property file, and verify the host name, port number, user name, and password.

5. Verify gateway operation.

If the gateway is functioning correctly, the REPORTER database contains data.

- a. Change to the **db2inst1** user:

```
su - db2inst1
```

- b. Connect to the REPORTER database:

```
db2 connect to reporter
```

- c. Examine the event archive table:

```
db2 select node from reporter_status
```

```
NODE
```

```
-----  
host2.tivoli.edu  
host2.tivoli.edu  
host2.tivoli.edu  
host1  
host1  
host2.tivoli.edu  
host1  
host1  
host1  
host1
```

The values that are shown for NODE indicate that the gateway is archiving event records to DB2.

- d. Examine the alternative tables.

Verify that the following commands return data:

```
db2 select name from reporter_classes  
db2 select column_name from reporter_conversions  
db2 select name from reporter_names  
db2 select name from reporter_groups  
db2 select owneruid from reporter_members
```

These tables are all populated when the gateway starts. Data in these tables indicates that the gateway startup command file is correct.

- e. Exit out of the **db2inst1** user back to the **netcool** user.

```
exit
```

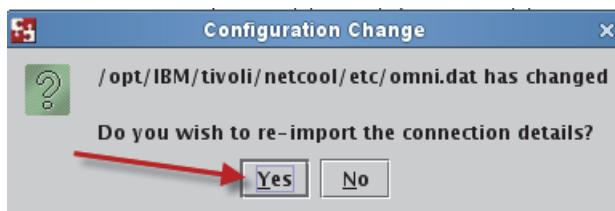
6. Add the gateway to process activity.

- a. Stop the running event gateway:

```
pkill nco_g_jdbc
```

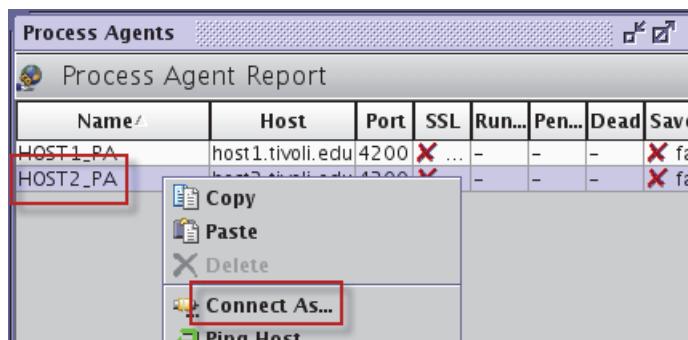
- b. Start the Netcool/OMNIbus administrator utility.

nco_config &

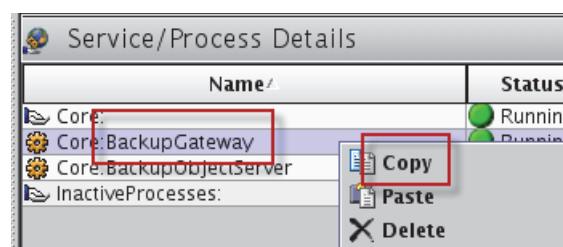


Previously you modified the Netcool/OMNIbus communications file and added an entry for the JDBC gateway. The administrator utility uses that communication file to locate the ObjectServers. The utility detects that the file is changed and wants to import the updated file.

- c. Click **Yes**.
- d. Click **Next**.
- e. Click **Finish**.
- f. Connect to the HOST2_PA process agent as the **netcool** user with password **object00**.

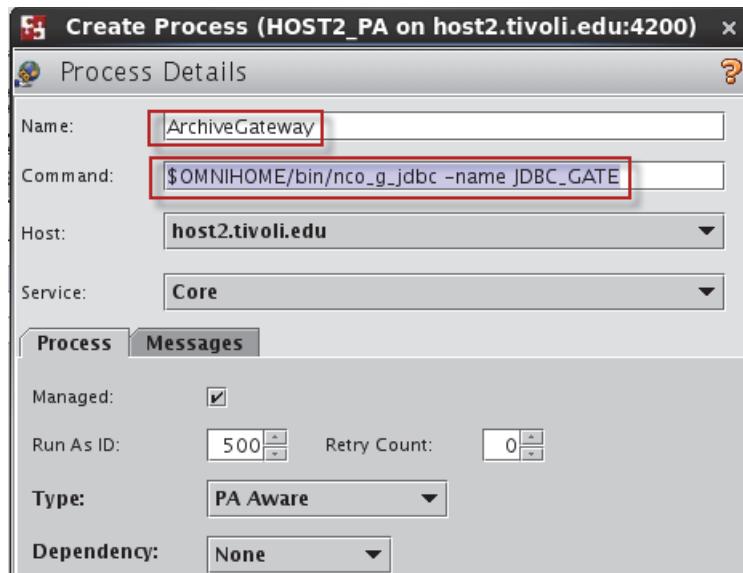


- g. Click **BackupGateway** to select it, right-click, and select **Copy**.



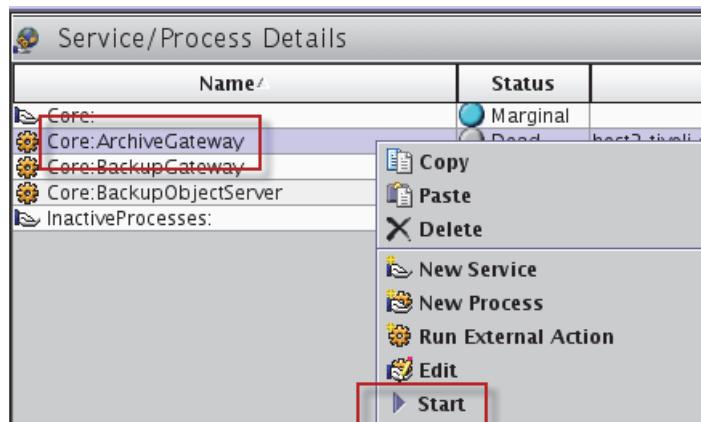
- h. Right-click and select **Paste**.
- i. Enter **ArchiveGateway** for the Name.
- j. Change the Command to **\$OMNIHOME/bin/nco_g_jdbc -name JDBC_GATE**

- k. Click OK.

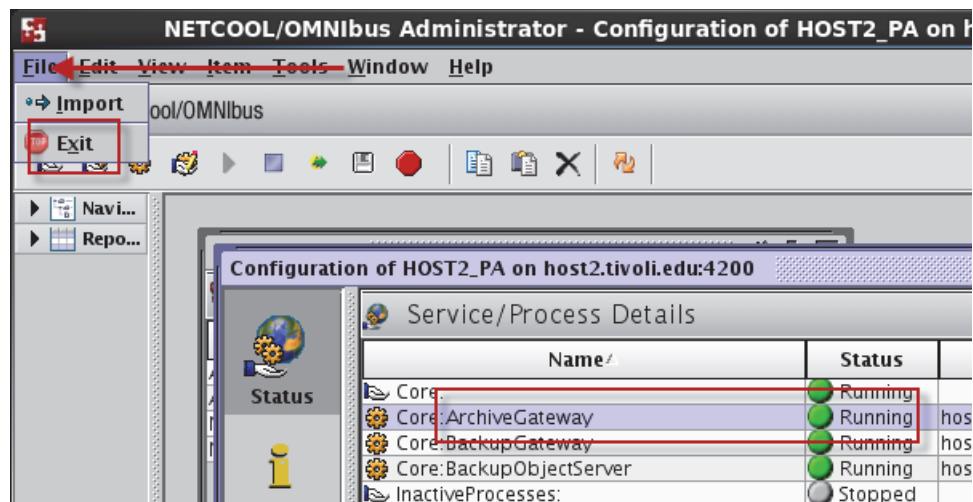


The entry ArchiveGateway appears with a status of Dead.

- l. Click the entry to select it, right-click and select Start.



- m. Click File and Exit.

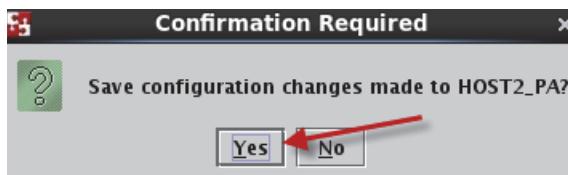


The JDBC gateway is now running.

n. Click **Yes**.



o. Click **Yes**.



The modifications that are made to the process activity configuration are automatically added to **\$OMNIHOME/etc/nco_pa.conf**.

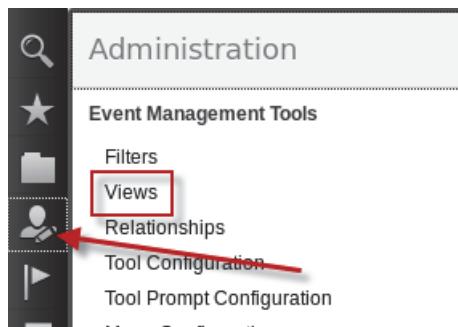
7. Modify the event view.

In a previous step, you verified that the customized delete tool updated the Delete_Flag column correctly. You also verified that any event with Delete_Flag=1 was hidden from the **whill** user. In the following steps, you verify that the custom trigger physically removes the event.

a. Open a Firefox browser.

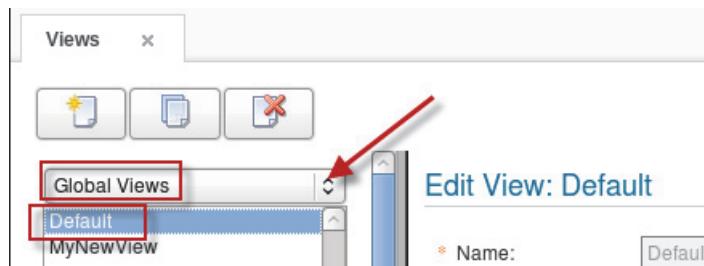
b. Log in to Dashboard Application Services Hub as the **ncoadmin** user.

c. Click the icon and select **Views**.



To facilitate verification of the event deletion solution, you modify an event view, and add two column names.

d. Change the selection to **Global Views** and click **Default**.



The view definition opens.

- e. Click **Archived_Flag** to select it and click the right-arrow to add the column to the view.

Edit View: Default

* Name: Default

Data Source: Click to show

Display Columns Sort Columns Group Columns Relationships

Available fields:

- AggregationFirst
- AlertKey
- Archived_Flag**
- BSM_Identity
- CauseType
- ...

Event list view:

- Severity [locked]
- Acknowledged [locked]
- Node
- AlertGroup
- Summary

The column is added to the bottom of the list.

- f. Click the column to select it and click the double up arrow to move the column to the top of the list.

Relationships

Event list view:

- Node
- AlertGroup
- Summary
- LastOccurrence
- Tally
- Type
- ExpireTime
- Agent
- Manager
- Archived_Flag**

- g. Click **Delete_Flag** to select it and click the right arrow to add it to the list.

Display Columns Sort Columns Group Columns Relationships

Available fields:

- CorrScore
- Customer
- Delete_Flag**
- DisplayFirst
- EventId
- ExtendedAttr
- FirstOccurrence
- Flash

Event list view:

- Severity [locked]
- Acknowledged [locked]
- Archived_Flag**
- Node
- AlertGroup
- Summary
- LastOccurrence
- Tally

- h. Move it to the top of the view.

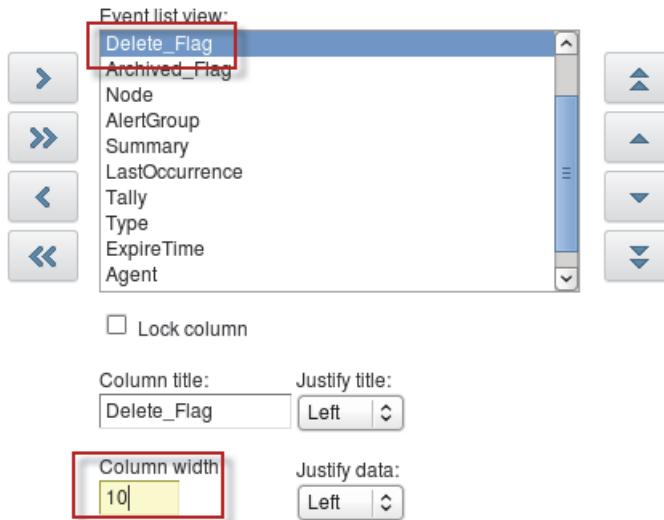
Group Columns Relationships

Event list view:

- Severity [locked]
- Acknowledged [locked]
- Delete_Flag**
- Archived_Flag**
- Node

The two columns appear at the top of this list, which causes them to appear on the far left of the user event display.

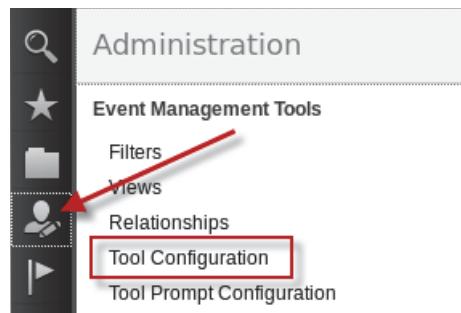
- i. Select the Delete_Flag column, scroll down, and change the column width to 10.



- j. Repeat this step to change the column width for the Archived_Flag column.
 - k. Click **Save and Close**.
8. Modify the NoDelete tool.

To verify that the event delete solution works correctly, you need to be able to view an event, mark the event for deletion, and then verify that it is removed from the ObjectServer. If you use the **whill** user, after the event is marked for deletion, it disappears from the event list and you cannot verify that it is physically removed. If you use the **ncoadmin** user, the custom NoDelete tool is not available in the event list menu, which is by design. To use the **ncoadmin** user to verify the solution, you must modify the NoDelete tool, and make it available to the user.

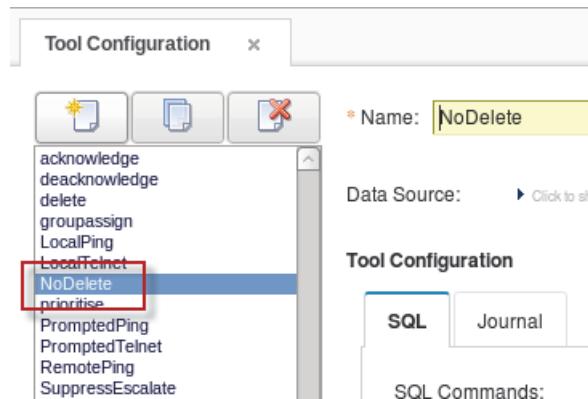
- a. Click the icon and select **Tool Configuration**.



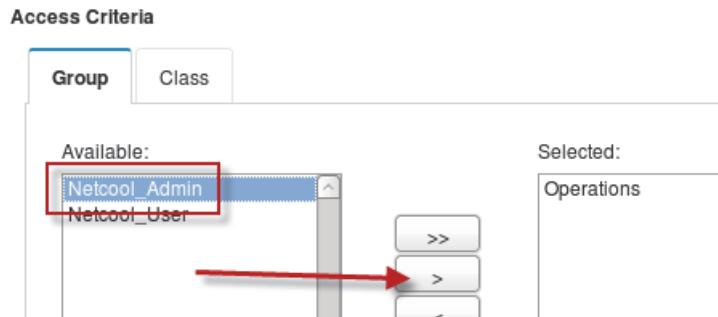
- b. Expand **Event Management Tools**.
- c. Click **Tool Configuration**.

The list of available tools opens.

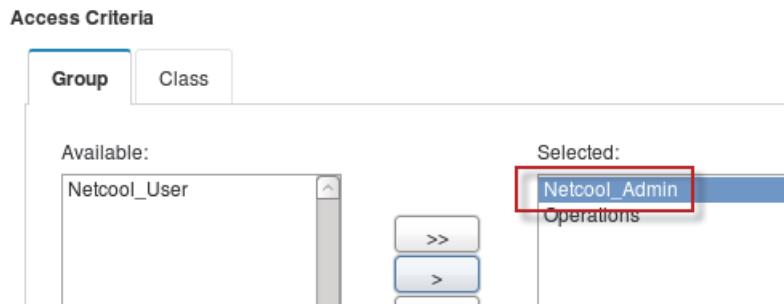
- d. Click **NoDelete**.



- e. Scroll to the bottom of the page and locate the Access Criteria section.
- f. Click **Netcool_Admin** to select it and click the right arrow to add it to the selected column.



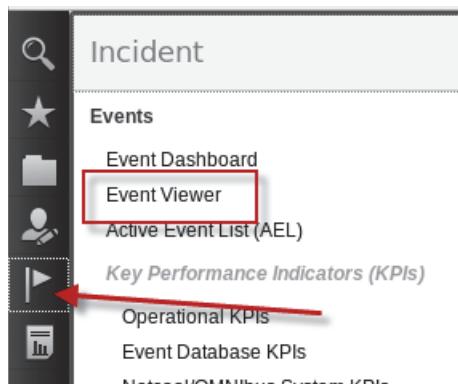
Users that belong to the Netcool_Admin group can now see the NoDelete tool.



- g. Click **Save**.

9. Verify event delete solution.

- a. Click the icon and select **Event Viewer**.



The event list opens, and the two columns: Delete_Flag, and Archived_Flag, are shown in the view. In the previous screen capture, all events are flagged as archived, and no events are flagged for deletion.



Hint: The JDBC Gateway modifies the Archived_Flag column *after* the record is written to DB2.

Group	Cou	Sev	Sev	Ack	Delete_Flag	Archived_Flag	Node
All	903	✖	✖	No	No	Yes	host1
Critical	1	✖	⚠	No	No	Yes	link6
Major	7	⚠	⚠	No	No	Yes	London
Minor	4	⚠	⚠	No	No	Yes	Tokyo
Warning	40	ⓘ	⚠	No	No	Yes	Washington

- b. Click an event in the list to select it and right-click.

Two tools are labeled **Delete**. The first tool is the original tool that removes the event record. The second tool is the custom NoDelete tool that marks the event for deletion.

Sev	Ack	Delete_Flag	Archived_Flag	Node
✖	No	No	Yes	host1
⚠	No	No	Yes	link6
⚠	No	No	Yes	London
⚠	No	No	Yes	Tokyo
⚠	No	No	Yes	host1.tivoli.edu
⚠	No	No	Yes	link1
⚠	No	No	Yes	Moscow
⚠	No	No	Yes	host1.tivoli.edu

- c. Click the second **Delete** tool.

Sev	Ack	Delete_Flag	Archived_Flag	Node
✖	No	No	Yes	host1
⚠	No	Yes	Yes	link6
⚠	No	No	Yes	London
⚠	No	No	Yes	Tokyo

The custom delete tool sets `Delete_Flag=1` (Yes). With both columns set to 1, the event is now a candidate for physical deletion. The `DeleteArchivedEvent` trigger deletes the event. This trigger is a temporal trigger with a frequency of 60 seconds.

Sev	Ack	Delete_Flag	Archived_Flag	Node
✖	No	No	Yes	host1
⚠	No	No	Yes	host1.tivoli.edu
⚠	No	No	Yes	London
⚠	No	No	Yes	host1.tivoli.edu
⚠	No	No	Yes	link1
⚠	No	No	Yes	Moscow

After the trigger runs, the event is physically removed from the ObjectServer.



Note: When the event is removed from the primary ObjectServer by the trigger, the `NYC_AGG_GATE` component removes the event from the backup ObjectServer.

10. Modify the NoDelete tool and remove the Netcool_Admin group from the Access Criteria.

Leave the browser session as is. You use it again shortly.

Exercise 4 Installing the Netcool/OMNibus report package

A set of Tivoli Common Reporting reports are bundled with Netcool/OMNibus. The report package must be imported into Tivoli Common Reporting.

1. Copy the report package.

The report package file must be placed in a specific directory.

a. Change to the source directory:

```
cd $OMNIHOME/extensions/tcr_event_reports
```

b. Copy the file to the target directory:

```
cp Netcool_OMNIbus.zip /opt/IBM/JazzSM/reporting/cognos/deployment
```

2. Import the package.

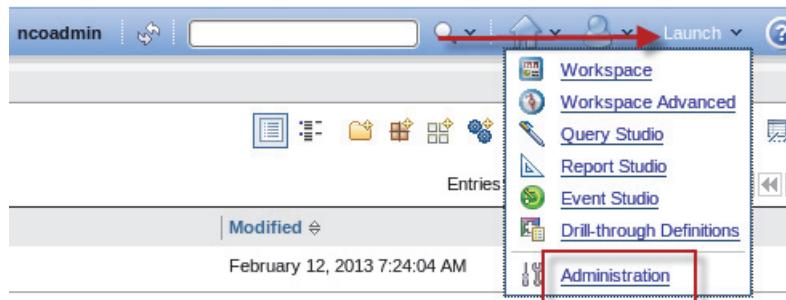
a. Return to the browser.

b. Click the icon and select Common Reporting.

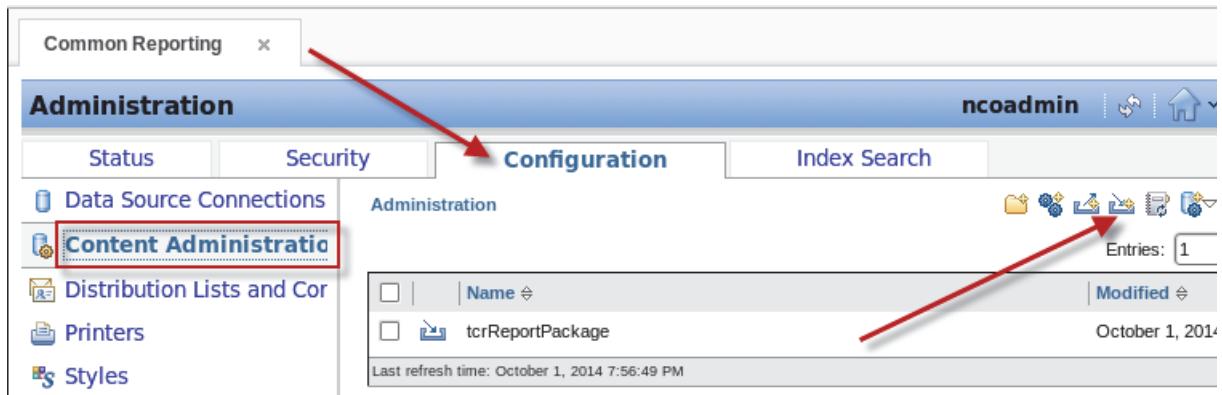


Important: In a previous exercise, you added the tcrPortalOperator to the Netcool_Admin group. The **ncoadmin** user is a member of that group. That role grants access to this feature.

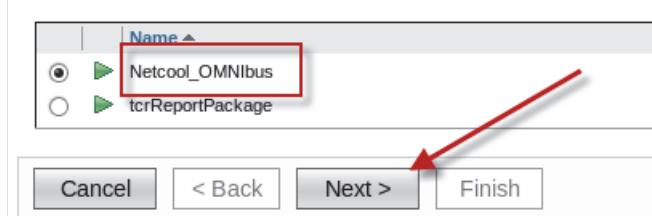
- c. Click **Launch** and select **Administration**.



- d. Click the **Configuration** tab.
e. Click **Content Administration**.
f. Click the indicated icon to start a **New Import**.

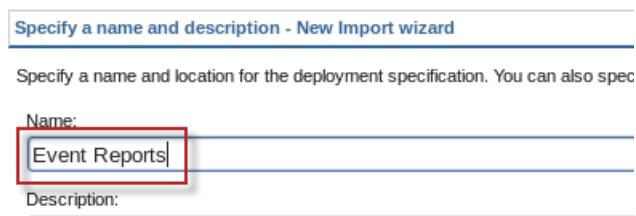


- g. Verify that the Netcool_OMNIbus package is selected and click **Next**.

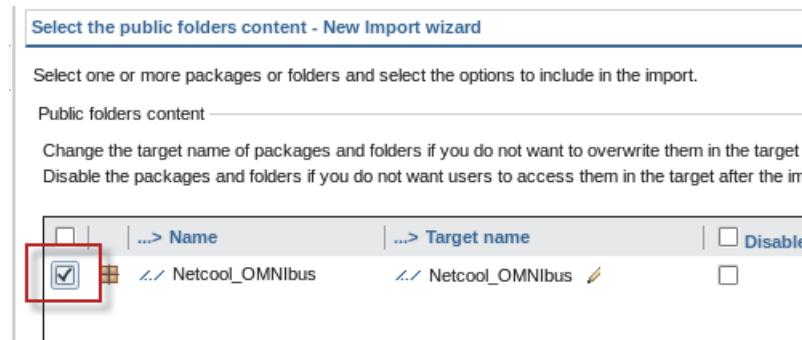


Important: If the Netcool_OMNIbus package is not listed, it means that it was not copied to the correct location.

- h. Enter a name, and click **Next**.



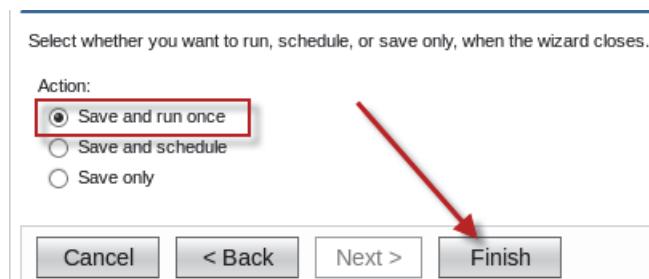
- i. Click the box to select the Netcool_OMNIbus package and click **Next**.



- j. Scroll to the bottom of the page, and click **Next**.

- k. Scroll to the bottom of the page, and click **Next**.

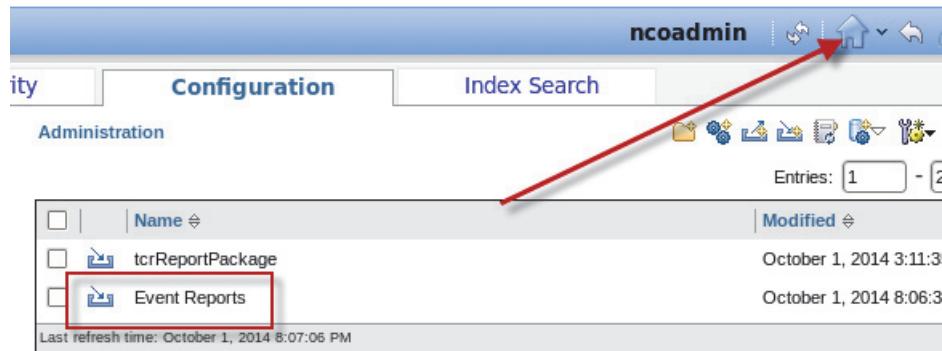
- l. Leave the option set as shown and click **Finish**.



- m. Scroll to the bottom of the page, and click **Run**.

- n. Click **OK**.

- o. Click the icon to return to the home page.



The import process is complete.

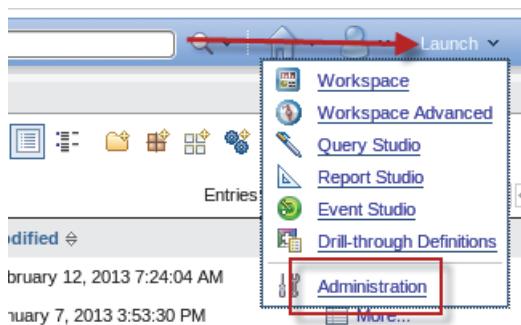


The Netcool/OMNibus report package is now available.

3. Create a data source.

A Tivoli Common Reporting data source defines the location of the database that reports use.

a. Click **Launch** and select **Administration**.



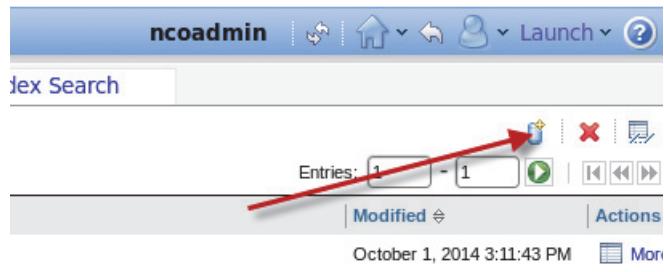
b. Click the **Configuration** tab.

c. Click **Data Source Connections**.



The available data sources are listed. The entry that is shown defines the location of the Tivoli Common Reporting report store database. The overview report run previously uses this data source.

- d. Click the indicated icon to create a new data source.



- e. Enter **Reporter** for the name and click **Next**.

Specify a name and description - New Data Source wizard

Specify a name and location for this entry. You can also specify a d

Name: **Reporter**

Description:



Important: The name *must* be Reporter because this value is defined in the report templates.

- f. Select IBM DB2 for the database type.
- g. Remove the check mark to configure a JDBC connection and click **Next**.

Specify the parameters for the connection of this new data source. The name c

Type: **IBM DB2**

Isolation level:

Use the default object gateway

Specify a value: **Cursor stability**

Configure JDBC connection



Note: In a production environment, the database might be on a remote server. In that case, you can define a JDBC connection.

- h. Enter REPORTER for the DB2 database name.

Specify the IBM DB2 connection string - New Data Source wizard

Edit the parameters to build a DB2 connection string.

DB2 database name: **REPORTER**

DB2 connect string:



Important: This value must be REPORTER because it is the database name that is cataloged in DB2.

- i. Scroll down to the bottom of the page.
- j. Select the **Password** check box.
- k. Enter *db2inst1* for the user ID and *object00* for the password.
- l. Click the line that is labeled **Test the connection**.

An external namespace:
VMMPProvider (Active) ▾

Signons

Password

Create a signon that the Everyone group can use:

User ID:

Password:

Confirm password:

Testing
[Test the connection...](#)

- m. Click **Test**.

Test the connection - New Data Source wizard

Test the parameters that make up the database connection.

Connection string:
^User ID:^?Password:;LOCAL;D2;DSN=REPORTER;UID=%s;PWD=%s;@ASYNC=0@0/0@COLSEQ=

[Test](#)

- n. Verify that the test is successful and click **Close**.

...> Name	Type / Query Mode	Status	Message
http://host2.tivoli.edu:16310/tarf	IBM DB2 / Compatible	Succeeded	

[Close](#)



Important: If the test fails with QE-DEF-0285, it typically means that either the user ID or password is incorrect, or the **netcool** user environment variables are not correct.

- o. Scroll to the bottom of the page, and click **Close**.

Test the connection using:
User ID:
db2inst1
Password:

Close

- p. Scroll to the bottom of the page, and click **Finish**.

Testing
Test the connection...

Cancel < Back Next > Finish

The **Reporter** data source is shown in the list of available data sources.

- q. Click the icon to return to the report packages.

Directory > Cognos

	Name	Modified
<input type="checkbox"/>	Reporter	October 2, 2014
<input type="checkbox"/>	servletInventory	October 1, 2014

4. Verify the Netcool/OMNibus reports.

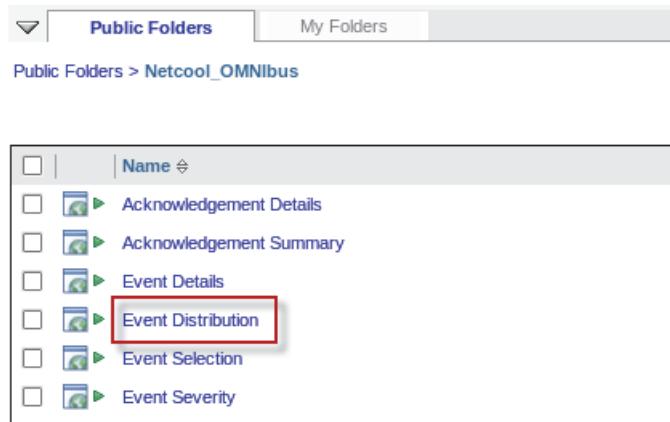
- a. Click **Netcool_OMNIbus** to view the report templates.

Public Folders

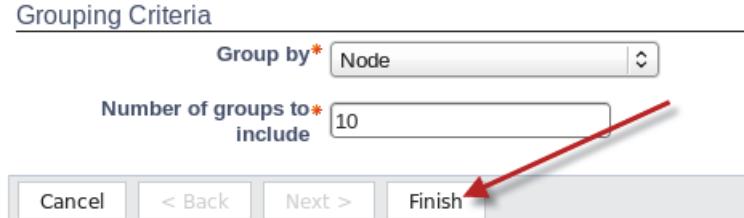
	Name
<input type="checkbox"/>	Common Reporting
<input type="checkbox"/>	Netcool_OMNIbus

The list of reports opens.

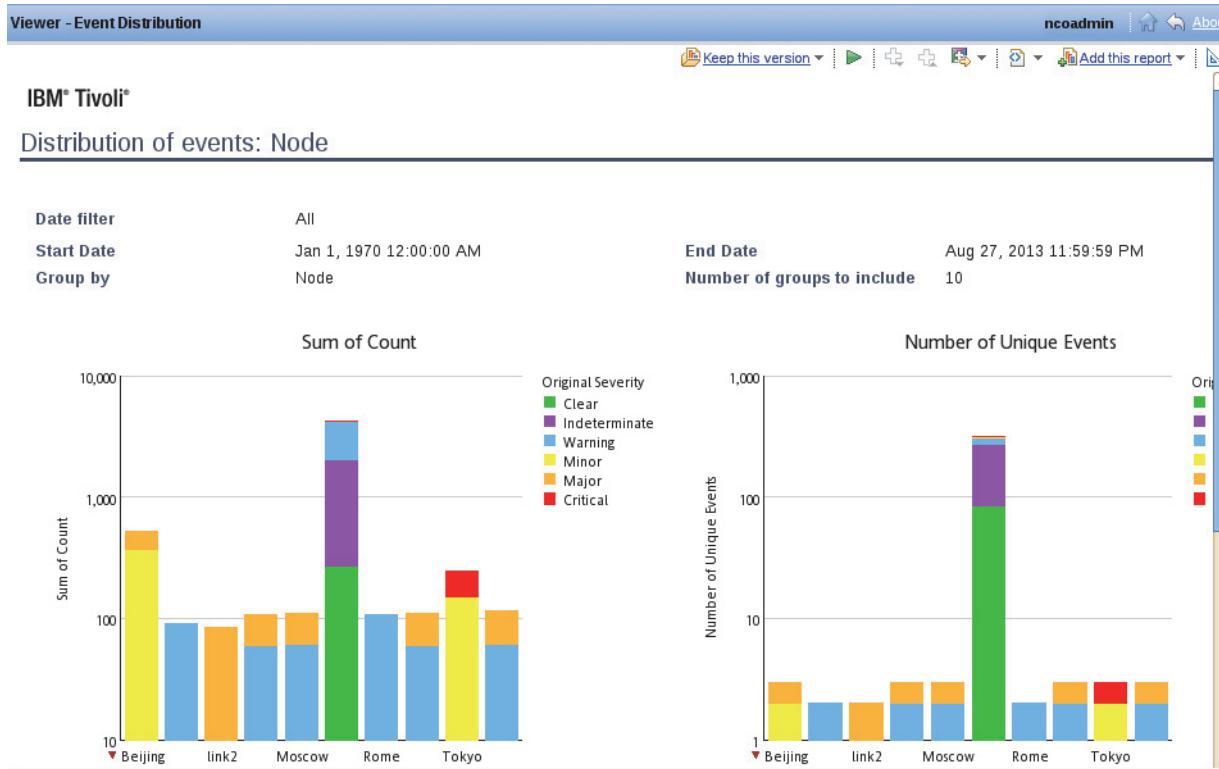
b. Click Event Distribution.



- c. Leave all the default values on the prompt page, scroll to the bottom of the page, and click **Finish**.



The report is generated.



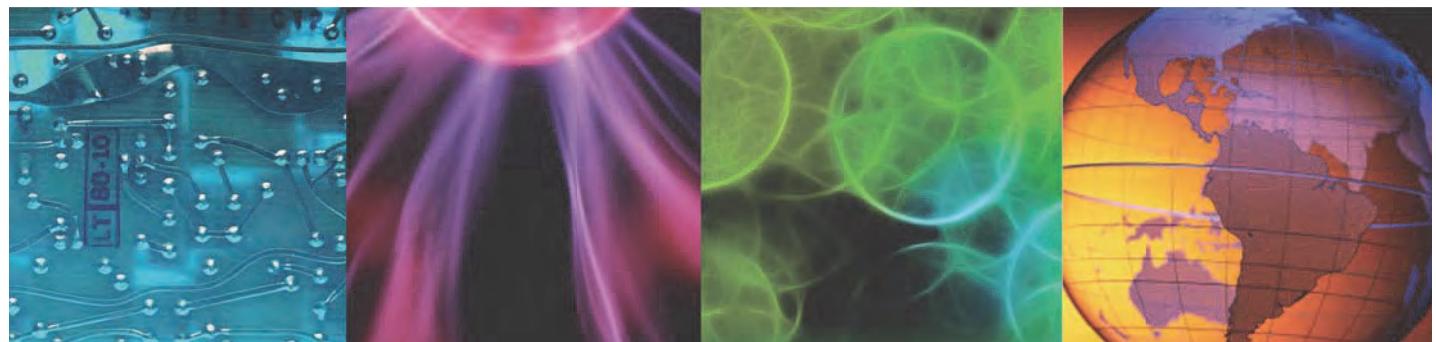
The event distribution report opens.

The exercises for this unit are complete. Feel free to explore other reports as time allows.

The summary of accomplishments for this unit includes these tasks:

- Tivoli Common Reporting is installed.
- Report archive database structure is defined.
- The JDBC Gateway is installed, configured, and running.
- Process Activity is configured to automatically start the JDBC Gateway.
- The Netcool/OMNibus report package is installed.
- The reports are verified.

TN025 1.0



ibm.com/training

Authorized
IBM | Training



Printed in Ireland