

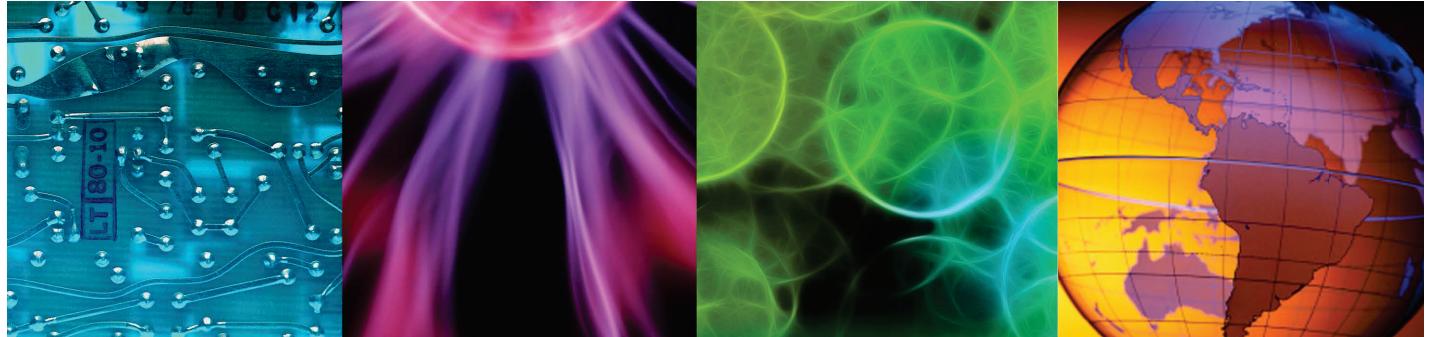


# IBM Training

## IBM Tivoli Monitoring 6.3 Advanced Administration Student Exercises

Course code TM063 ERC 1.0

November 2013



**Cloud & Smarter Infrastructure**

[ibm.com/training](http://ibm.com/training)

Authorized  
**IBM | Training**

All files and material for this course (TM063, IBM Tivoli Monitoring 6.3 Advanced Administration) are IBM copyright property covered by the following copyright notice.

© Copyright IBM Corp. 2013, 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](http://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

<b>About these exercises .....</b>	<b>v</b>
<b>1 Queries exercises .....</b>	<b>1</b>
Exercise 1. Using advanced options .....	1
Exercise 2. Defining data collection rules .....	6
Creating a custom query against an existing data source .....	7
Testing the queries .....	12
Creating a custom query against a new database .....	16
Specifying the databases in the Tivoli Enterprise Portal Server .....	16
Building the custom query against the new data source .....	17
<b>2 Advanced link topics exercises .....</b>	<b>21</b>
Exercise 1. Creating advanced links to filter link target workspace content .....	21
Creating the query .....	22
Adding link symbols to the query .....	22
Creating the target workspace .....	23
Creating the link .....	28
Header and footer link symbols .....	37
Exercise 2. Creating dynamic links to dynamically select a link target .....	43
Exercise 3. Creating a relative link .....	54
<b>3 Monitoring your enterprise using advanced situation techniques exercises .....</b>	<b>62</b>
Exercise 1. Modeling situations .....	63
Using historical monitored data .....	63
Exercise 2. Using embedded situations .....	73
Exercise 3. Creating correlated situations across multiple managed systems .....	79
Exercise 4. Using dynamic thresholds .....	82
<b>4 Agent autonomy exercises .....</b>	<b>89</b>
Exercise 1. Monitoring without a Tivoli Enterprise Monitoring Server .....	89
Creating a private situation by exporting an existing situation to XML .....	89
Creating a private situation manually .....	102
Exercise 2. Using the Event Integration Facility .....	106
Exercise 3. Agent autonomy heartbeats .....	108
Exercise 4. Implementing private historical data collection .....	110
Exercise 5. Using the Agent Service interface .....	113
<b>5 Working with policies exercises .....</b>	<b>119</b>
Exercise 1. Creating a policy .....	119

Including a failure message in the workflow .....	126
Including a success message in the workflow .....	131
Connecting the workflow .....	131
Exercise 2. Modifying policy settings .....	137
Exercise 3. Starting the policies .....	140
Exercise 4. Atomizing policies .....	143
<b>6 Agentless monitoring exercises .....</b>	<b>149</b>
Exercise 1. Agentless monitoring overview .....	150
Exercise 2. Agentless monitoring workspaces .....	157
Exercise 3. Configuring a second system for agentless monitoring .....	159
<b>7 Agent Management Services exercises .....</b>	<b>162</b>
Agent Management Services overview .....	162
Exercise 1. Exploring Agent Management Services workspaces .....	163
Exercise 2. Exploring Agent Management Services features .....	165
Exercise 3. Restarting the Monitoring Agent for Linux OS by using Agent Management Services .....	171
<b>8 Managing situation event integration exercises .....</b>	<b>173</b>
Exercise 1. Configuring IBM Tivoli Monitoring to forward situation events to Netcool/OMNibus .....	174
Exercise 2. Customizing the EIF slot .....	182
<b>9 Command-line interfaces exercises .....</b>	<b>187</b>
Command-line interfaces overview .....	187
Exercise 1. Using the SOAP command interface to work with a situation and a policy .....	188
Exercise 2. Using the SOAP command interface to extract attribute information .....	193
Exercise 3. Exporting and importing infrastructure data .....	197



# About these exercises

The exercise environment consists of four Linux images (VM01, VM02, VM04, and VM05).

- **VM01** hosts these Tivoli® Monitoring components:
  - Hub Tivoli Enterprise Monitoring Server
  - Tivoli Enterprise Portal Server
  - Portal server database using DB2® UDB
  - Tivoli Enterprise Portal (client)
  - Monitoring agent for Linux OS
  - Tivoli Log File Agent
  - IBM® Tivoli Agent Builder agent
  - Warehouse Proxy agent
  - Summarization and Pruning agent
  - Tivoli Data Warehouse database using DB2® UDB
- **VM02** hosts these Tivoli Monitoring components:
  - Remote Tivoli Enterprise Monitoring Server
  - Monitoring agent for Linux OS
  - Tivoli Log File Agent
  - Java Web Start client
- **VM04** hosts these Tivoli Monitoring components:
  - Linux OS Tivoli Enterprise Monitoring Agent
- **VM05** hosts no Tivoli Monitoring components. It is used for an agentless monitoring exercise.

The primary system to work with is VM01. You can use any type of portal client that you want.

All exercises have step-by-step instructions. Some skills that you learn in earlier exercises also apply to later exercises. You will see fewer basic details in those later exercises.

You perform only numbered steps and lettered steps (for example, 1. Instruction and a.Substep). All other instructions provide background information, describe what you should achieve in an exercise, or explain the result of a step or steps.





# 1 Queries exercises

These exercises cover how to create and manage queries. Queries are used to retrieve data for views that are contained in workspaces. Typically, a product-provided query can be used for a view. If a product-provided query does not exist for the data that you want to retrieve, you can create the query by using the Query editor.

Queries can retrieve data from Tivoli® Enterprise Monitoring agents. This data can be the amount of free disk space, a list of processes running on a computer, or current network statistics.

Queries can also retrieve data from a database on the Tivoli Enterprise Portal Server. This database can be an ODBC database, a JDBC database, or a DB2® database.

## Exercise 1. Using advanced options

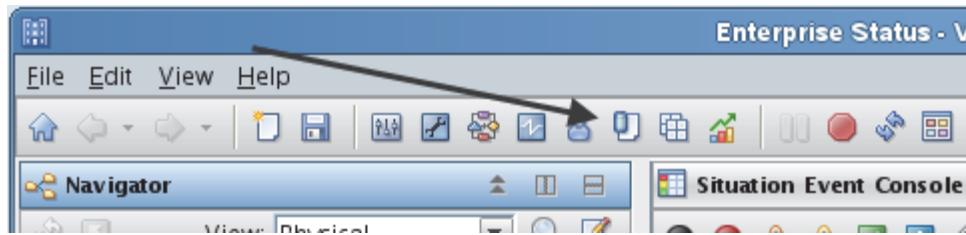
In this exercise, you copy and modify a product-provided query to show only the data that you want. Assume that you want to create a workspace that shows the top three processes consuming shared memory on the local Linux system.

1. Ensure that you have virtual machines **VM01**, **VM02**, and **VM04** started and communicating with each other.
2. On VM01, log in as **root** with a password of **object00**.
3. Double-click the **Tivoli Enterprise Portal Java Webstart Client** icon on the desktop.



4. Log in to the Tivoli Enterprise Portal by using a user ID of **sysadmin** with no password.

5. Click the **Queries**  to open the Query editor from the toolbar.



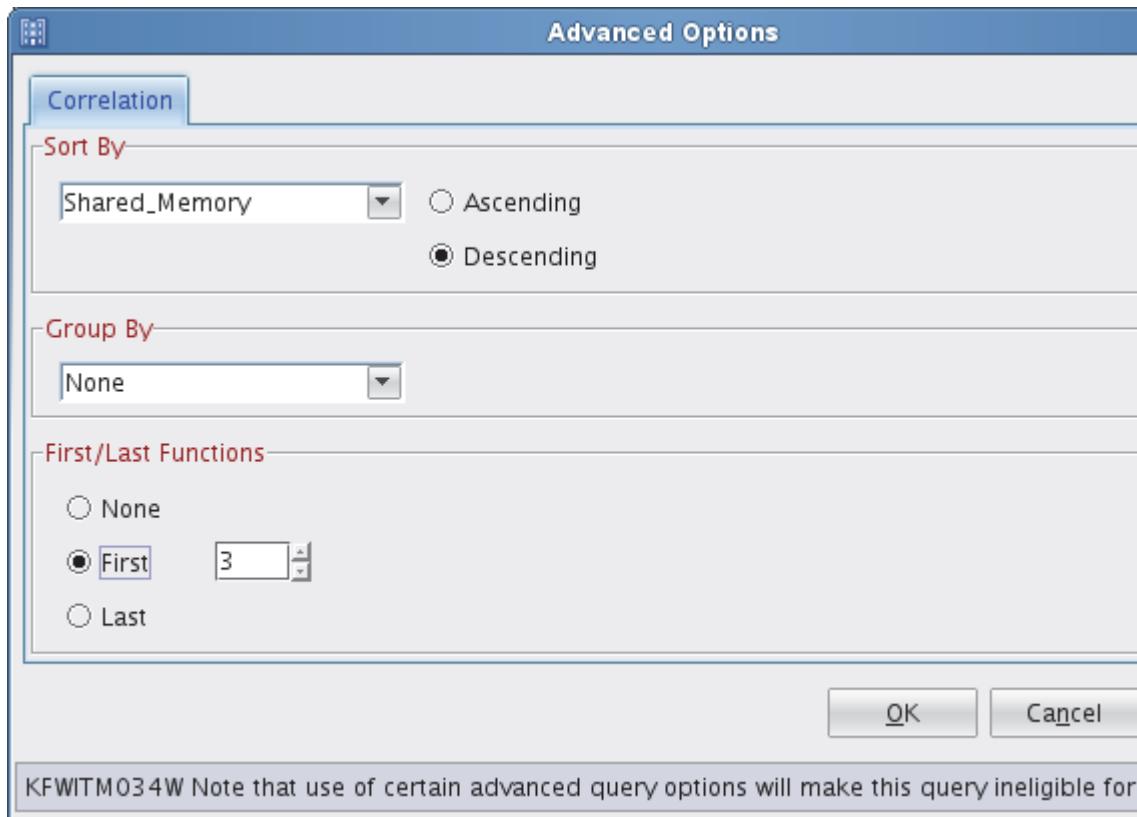
6. Expand the folder **Linux OS > Linux Process**.
7. Select and then right-click the query **Process**, and select **Create Another Query**.
8. Give the new query a name **Top3SharedMemoryPages** and click **OK**.
9. Select **Advanced**.

10. In the **Sort By** field, select **Shared\_Memory** and **Descending**.



**Note:** You can enter the first character of a field in the list to select it quickly. In this case, you enter an **S** several times.

11. In the **First/Last Functions** field, select **First**, and set the number to **3**.



# 1 Queries exercises

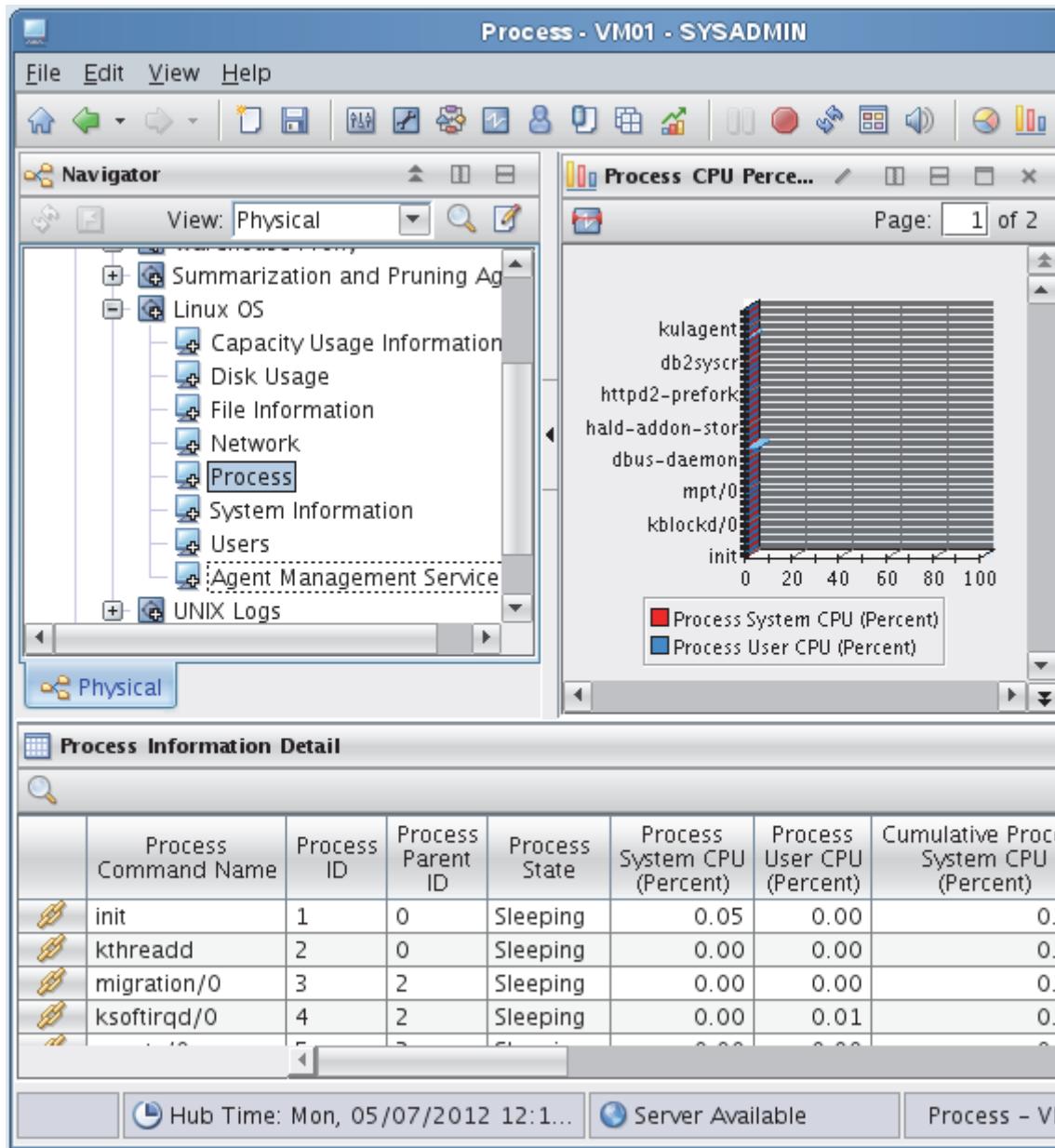
## Exercise 1. Using advanced options

12. Click **OK**.

You created a query that returns the three processes that are using the largest number of shared memory pages.

13. Click **OK** to close the Query editor.

14. Open the default Process workspace for the VM01 Linux system.



15. Split the Process Information Detail view horizontally.

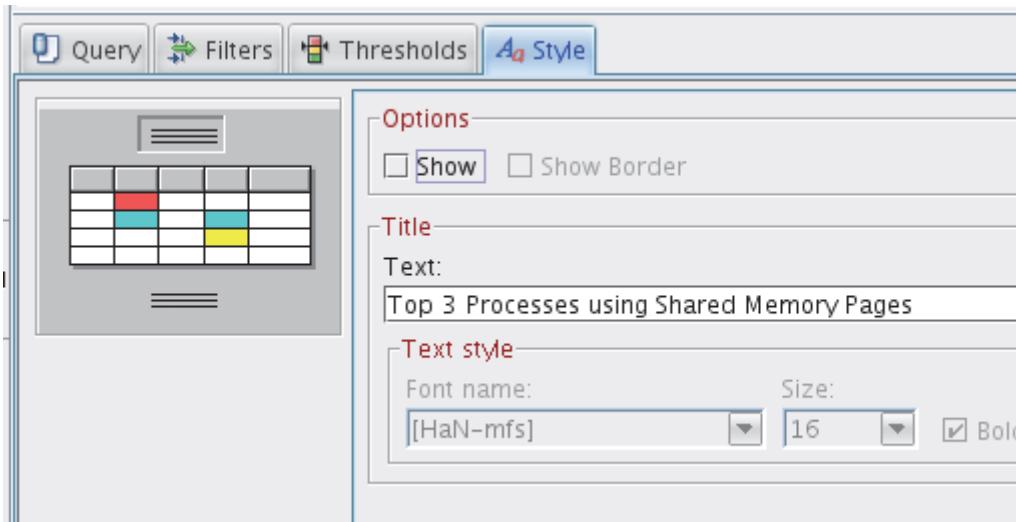
16. Click **Edit Properties** in the lower view and assign the newly created query called **Top3SharedMemoryPages** to this view.

The screenshot shows the 'Specification' tab of a query properties dialog. On the left, a tree view lists various Linux monitoring metrics. A red arrow points from the 'Top3SharedMemoryPages' item under 'Linux Process' to the 'Specification' table on the right. The table has four rows:

	System Name	Process ID
1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	$= = \$NODE\$$	$= = \$PID\$$
3		
4		

17. Click **OK**.

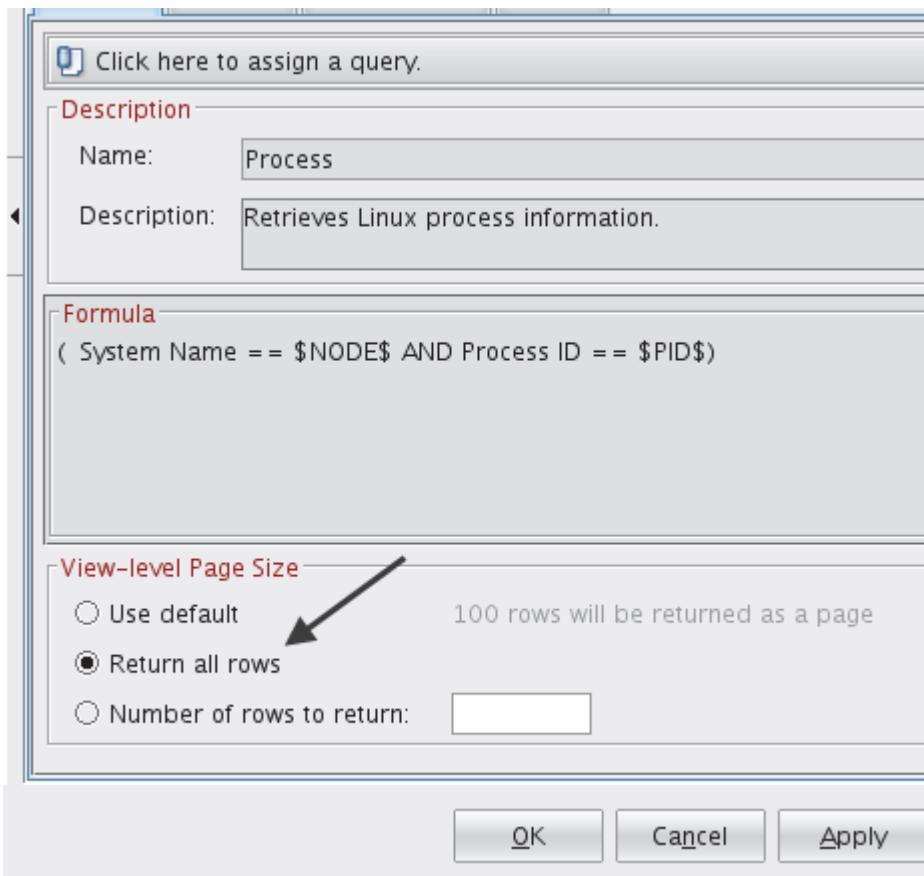
18. Click the **Style** notebook page, type a new title **Top 3 Processes using Shared Memory Pages**, and click **OK**.



19. Drag the attribute **Shared Memory (Pages)** in both of the lower views to be between **Process ID** and **Process Parent ID**.

20. Right-click a row in the Process Information Detail view and select **Properties**.

21. Change the View-level Page Size to **Return all rows**.



22. Test your new query by sorting Shared Memory (Pages) in the Process Information Detail view. In the view, sort the column **Shared Memory (Pages)** in descending order by selecting the attribute header two times. Inspect the workspace to see whether the data is consistent with what you expect. Remove any unwanted columns from the view by using the **Filter** tab.

Process Command Name	Process ID	Shared Memory (Pages)	Process Parent ID	Proc Sta
db2sysc	24966	34224	24964	Sleep
java	6905	16799	1	Sleep
KfwServices	5874	9094	1	Sleep
java	32156	8574	1	Sleep
db2fmp	27645	7713	24964	Sleep

System Name	Process ID	Shared Memory (Pages)	Process Parent ID	Process Command Name
VM01:LZ	24966	34224	24964	db2sysc
VM01:LZ	6905	16799	1	java
VM01:LZ	5874	9094	1	KfwServices

23. Save the workspace by using a name of **My\_Process\_Overview**.

## Exercise 2. Defining data collection rules

During this exercise, you create custom queries to include data from JDBC and DB2 data sources. As a first example, you write custom queries against the portal server database, which is stored in DB2 Universal Database.



**Note:** Queries are not stored as part of a workspace and are not specific to your user ID. Changes that you make to existing queries can have an impact on the configurations of other users. To prevent you from modifying product-provided queries, they are protected.

## Creating a custom query against an existing data source

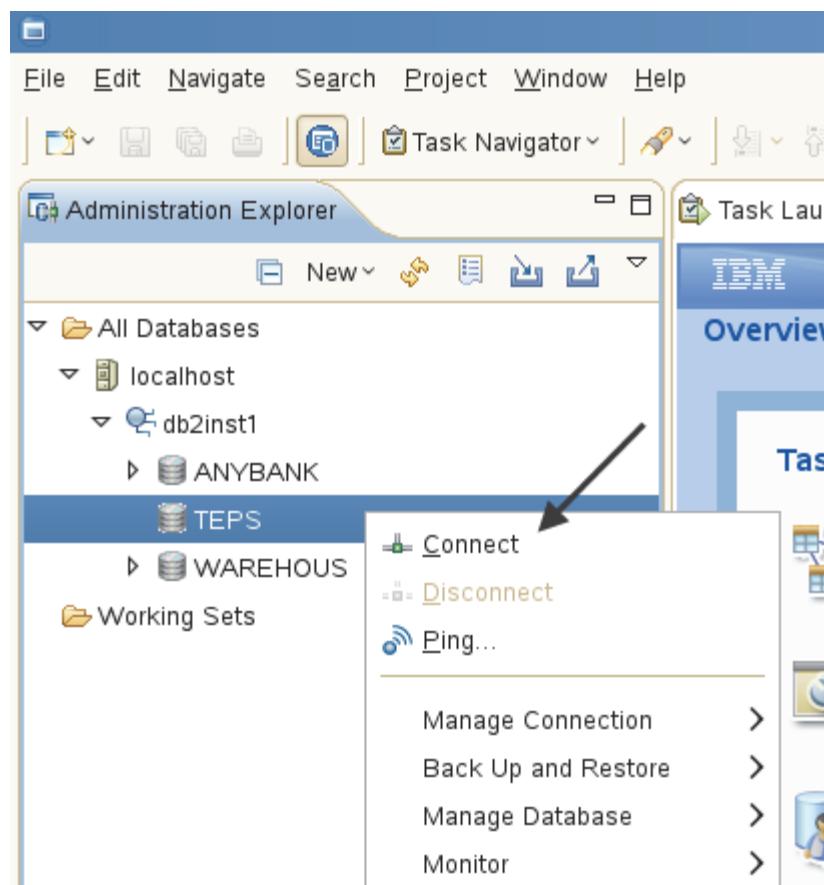
As you learned, configuration data such as users, queries, and Navigators is stored in the Tivoli Enterprise Portal Server database. In this exercise, you create queries that include this data in your workspaces.

Before you can import data, you must know the database layout and table names.

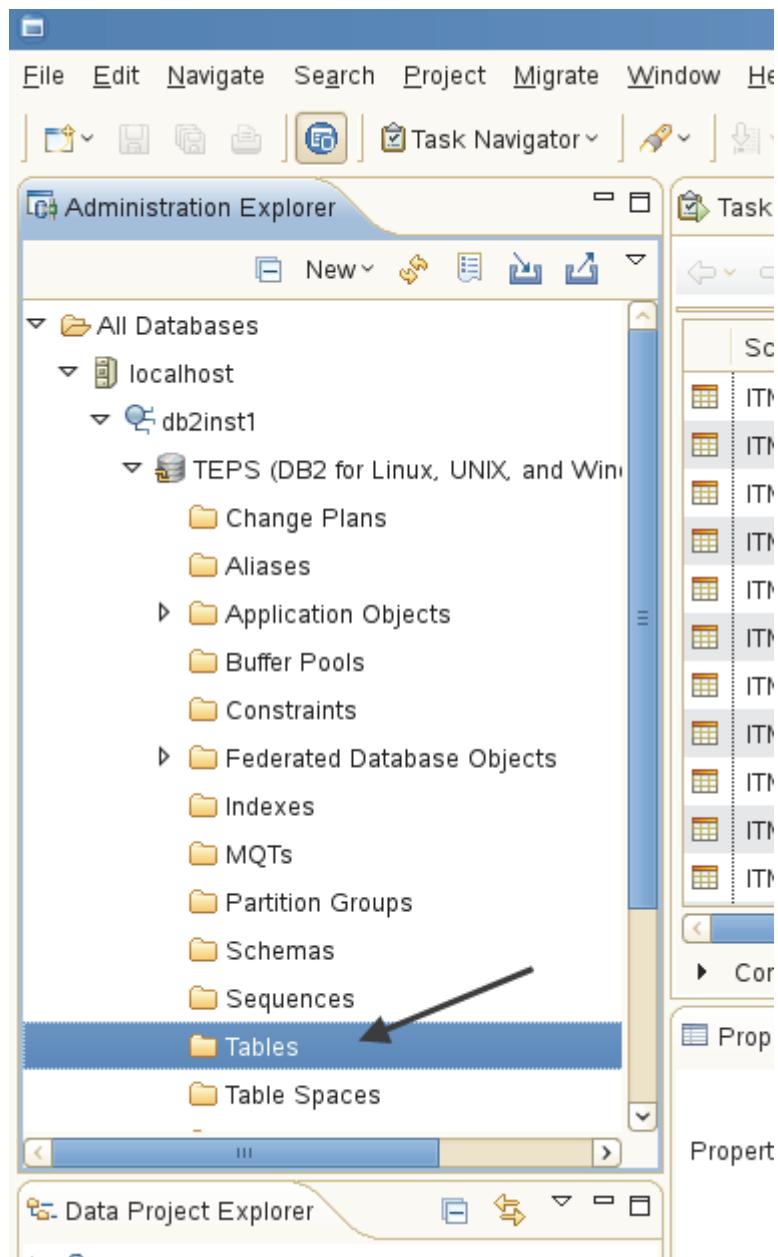
1. Open a terminal window on **VM01** and open the IBM Data Studio administration client by using the following command:

```
/opt/IBM/DSAC3.1.1/dastudio
```

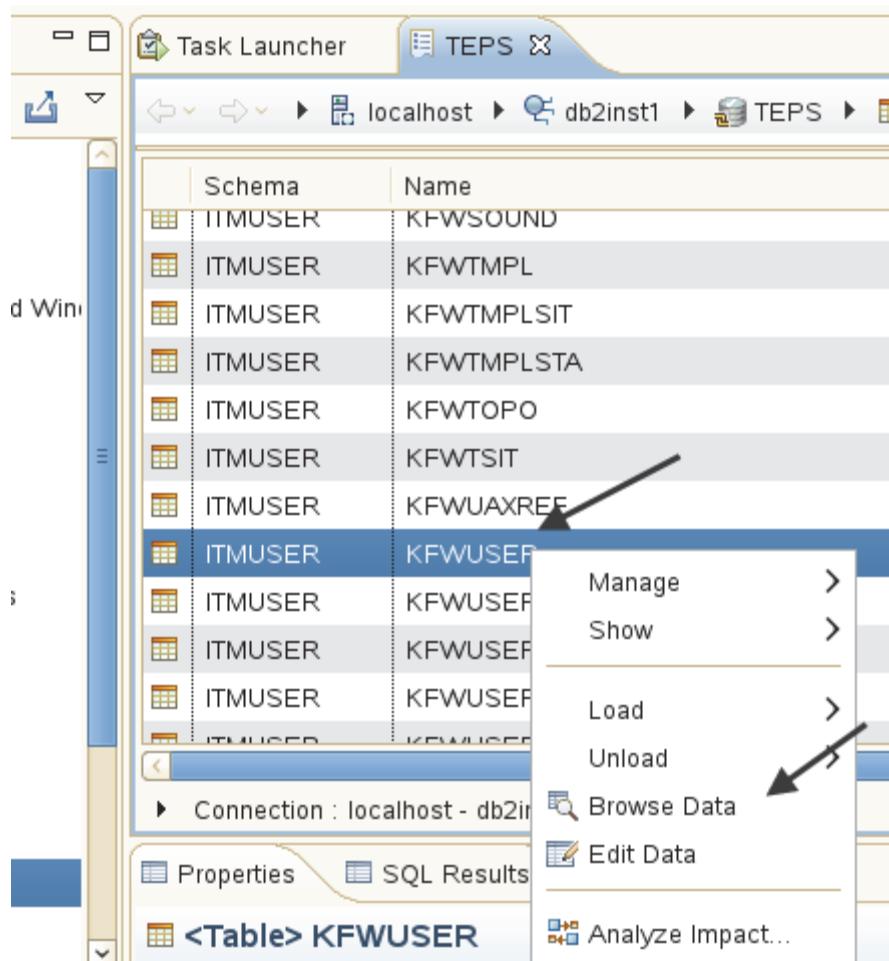
2. Select **All Databases > localhost > db2inst1 > TEPS**. Right-click **TEPS** and select **Connect**.



3. Select Tables.



4. Scroll down, right-click the table **KFWUSER**, and select **Browse Data**.



**Note:** Tables that are used by the Tivoli Enterprise Portal Server have a name that starts with **KFW**.

- Review the content of the **KFWUSER** table. This data is what you display in the Tivoli Enterprise Portal client.

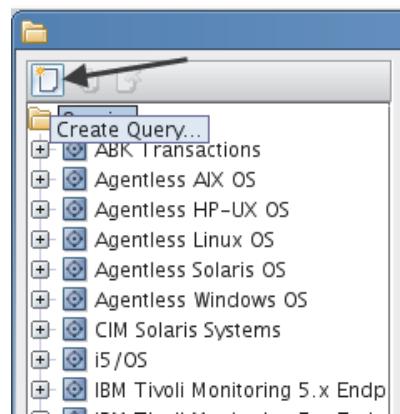
ID [VARCHAR(32)]	NAME [VARCHAR]
SYSADMIN	SYSADMIN
_USER_	Default
*ADMINISTRATOR	Kfw:ADMINISTRAT
*OPERATOR	Kfw:OPERATOR_U
*ITMDPUSER	Kfw:ITMDP_USER

During this part of the exercise, you create a query to include the following information about the Tivoli Enterprise Portal users:

- User IDs that are defined in Tivoli Enterprise Portal
- Users who are currently logged on

This data is stored in tables **KFWLOGIN** and **KFWUSER**.

- Close the IBM DB2 Administration IBM Data Studio client.
- Access your Tivoli Enterprise Portal client and open the Query editor. You can either press Ctrl+Q on your keyboard or select **Queries** from the toolbar.
- Click the **Create Query** icon to create a query.

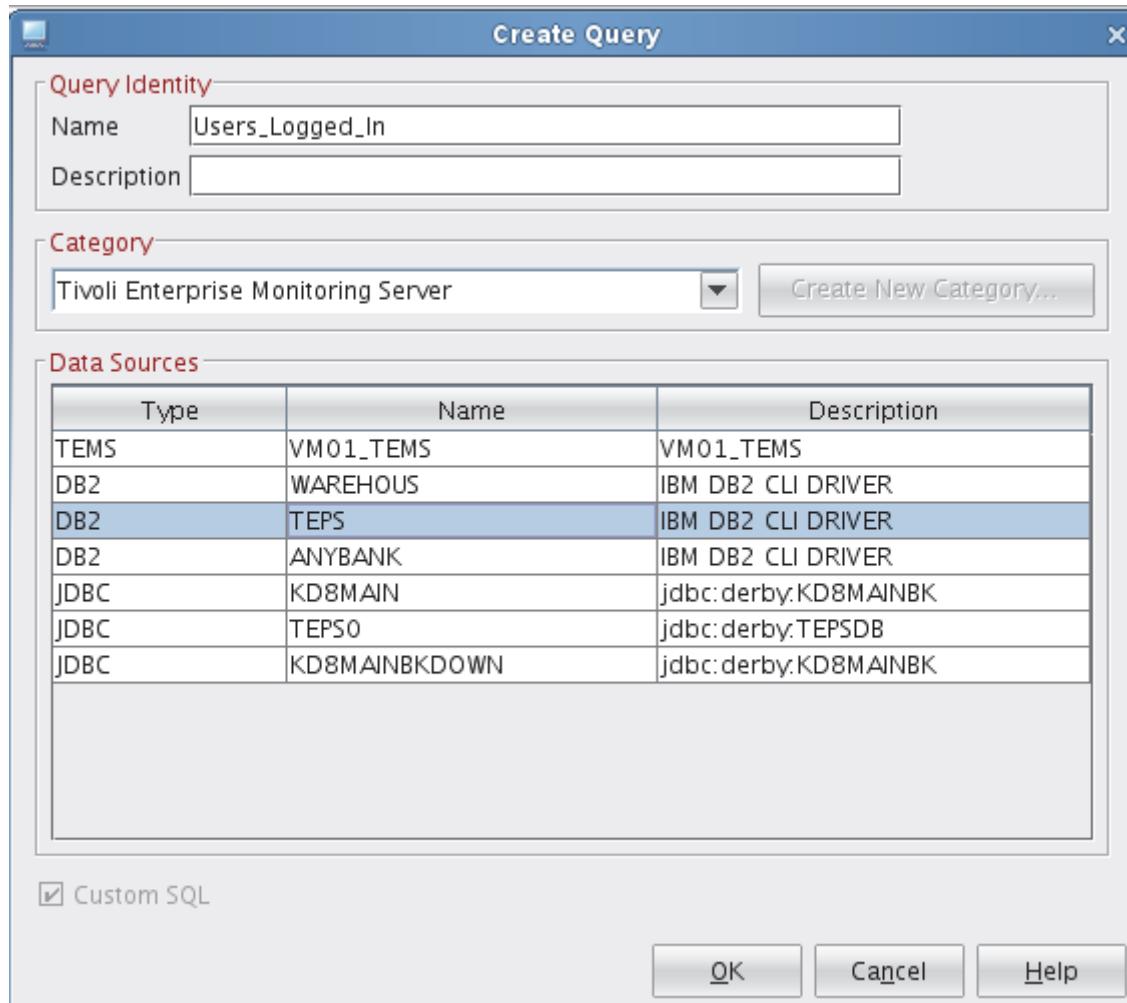


- Name the new query **Users\_Logged\_In**.

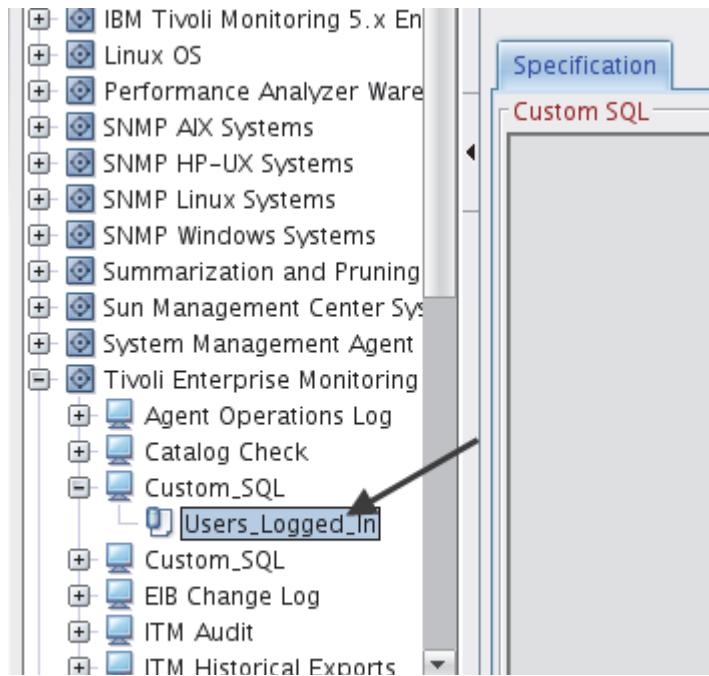
10. Select the category **Tivoli Enterprise Monitoring Server**, which categorizes your new query.

This action associates the query with the Tivoli Enterprise Monitoring Server, although the query runs in the Tivoli Enterprise Portal Server. Categories cannot be customized.

11. Select the **TEPS** data source from the list of existing data sources and click **OK**.



The new query is added under **Tivoli Enterprise Monitoring Server > Custom\_SQL**.



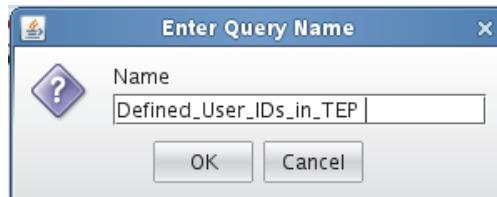
On the right pane of the Query editor, you find the **Specification** tab where you enter a SELECT statement to access the database. In this scenario, you include the entire table from the portal server database.

12. Enter the following SELECT statement:

```
SELECT * FROM KFWLOGIN
```

13. Click **Apply** to save the query, which is now ready to be used in a view.

14. Create a second query called **Defined\_User\_IDs\_in\_TEP**.



15. Select the same data source and category, and enter the following SELECT statement:

```
SELECT * FROM KFWUSER
```

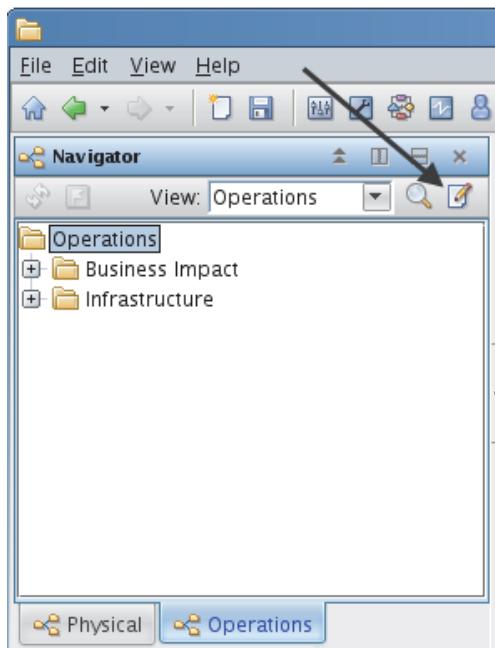
16. Save the query by clicking **OK**.

## Testing the queries

Use the new queries and display the data in tables.

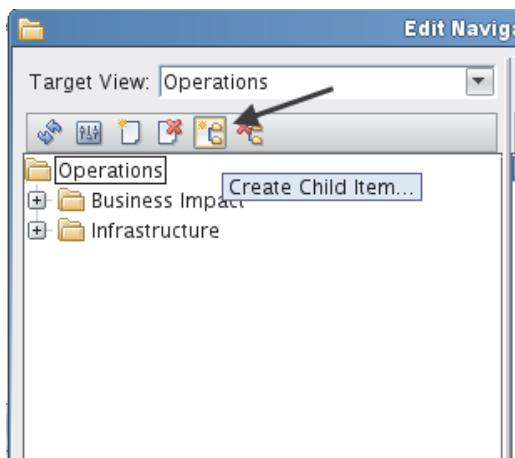
17. Access the **Operations Navigator** item.

18. Click the pencil icon to open the Navigator editor.



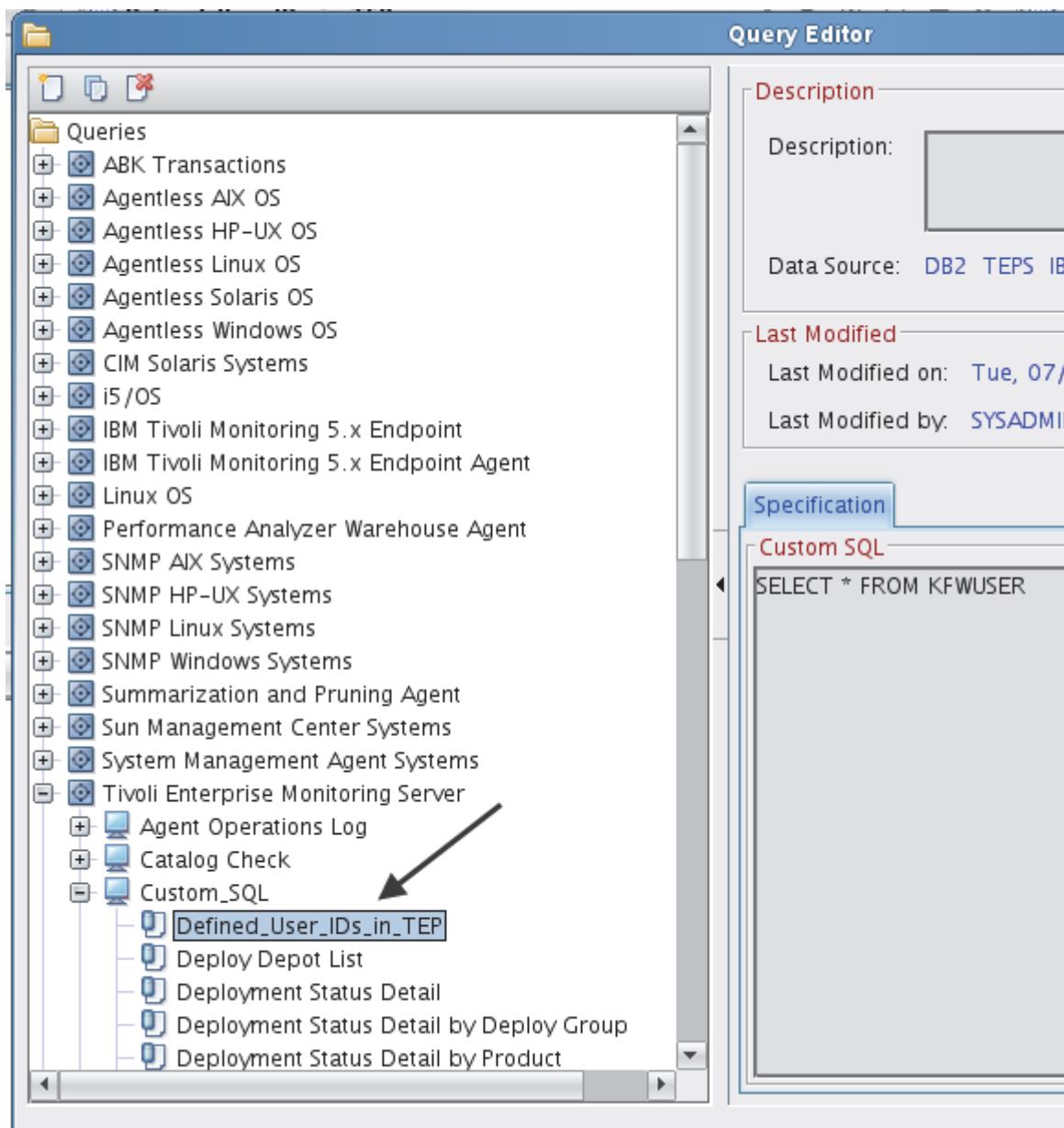
19. Ensure that **Operations** is selected in the Navigator editor.

20. Click the **Create Child Item** icon, which creates a child item under the **Operations** Navigator item.



21. Name this new Navigator item **Reports**, accept the Navigator update, and access this new Navigator item. You do not need to assign a managed system to this Navigator item. The Navigator item holds reports from the queries that you created.
22. Split the top view vertically into two panes and select a table view for both panes. Leave the bottom view as a Notepad. You use it later in the exercise.

23. Use the view properties to assign the **Defined\_User\_IDs\_in\_TEP** query to the left table view.  
Give it a meaningful title.



24. Use the **Filters** tab to eliminate any columns that you do not want to see. These columns are the ones with data that is not displayed in clear text.
25. Assign the **Users\_Logged\_In** query to the table view on the right side. Use the **Filters** tab to eliminate columns with data that is not displayed in clear text.

## 1 Queries exercises

### Exercise 2. Defining data collection rules

Ensure that the result looks like the following screen capture.

The screenshot shows the IBM Tivoli Monitoring interface with the title bar "Reports - VM01.tivoli.edu - SYSADMIN". On the left is a "Navigator" pane with a tree view of "Operations", "Business Impact", "Infrastructure", and "Reports". The main area contains two tables. The first table, titled "Defined\_User\_IDs...", has columns "ID" and "NAME" and contains the following data:

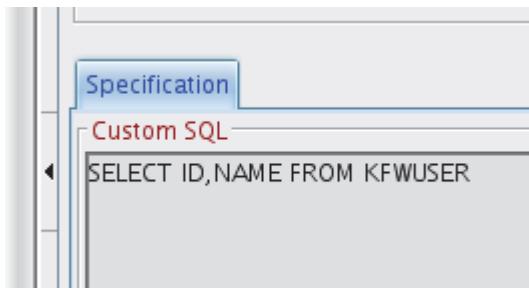
ID	NAME
SYSADMIN	SYSADMIN
_USER_	Default
*ADMINISTRATOR	ADMINISTRATOR
*OPERATOR	OPERATOR
*ITMDPUSER	Kfw:ITMDP_USER_NAME

The second table, titled "Users\_logged\_In", has columns "Time On", "User ID", and "IP Address" and contains one row of data:

Time On	User ID	IP Address
2013-07-02 07:53:50.000000	SYSADMIN	192.168.100.10



**Note:** Using the **Filter** tab is one way to eliminate unwanted columns from a view. This method is called **post-filtering**, and can add to network traffic. It is more efficient to extract only the columns that you want at the source by modifying the SQL statement, as shown in the following example.



This SQL statement returns only the ID and NAME columns from DB2, which eliminates the need to extract and transmit unwanted data.

### Optional step: Modifying the custom queries

26. Using what you learned, go back to your two custom queries and modify them to return only the columns that you want to see in your table views. Remove the asterisk and specify each column name, separated by commas.

## Creating a custom query against a new database

27. If it is not running, open a new terminal window and start your **abk.sh** file from **/labfiles/ABK**.

```
VM01:/labfiles/ABK # ./abk.sh
Welcome to App-O-Matic...
Opening file anybank_db2.any
Opening database AnyBank...OK
```

This script is writing business data into a database called ANYBANK.

During this exercise, you import the information into IBM® Tivoli Monitoring and display it in a table view. To display the data in Tivoli Enterprise Portal, you typically perform several steps:

- Make the database known to the Tivoli Enterprise Portal Server
- Recycle the Tivoli Enterprise Portal Server
- Write a custom query against the data

## Specifying the databases in the Tivoli Enterprise Portal Server

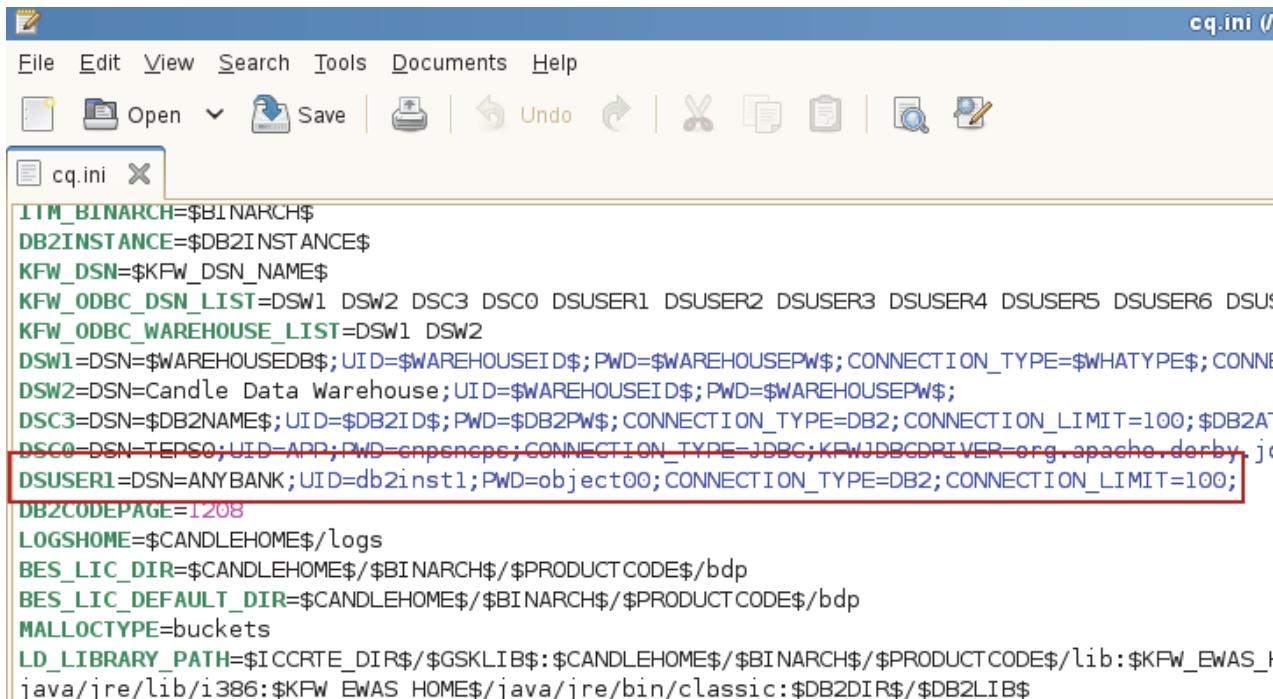
To see the new database in the portal client, it must be in the environment file for the Tivoli Enterprise Portal Server.

28. Open a terminal session and change to the directory **/opt/IBM/ITM/config**.
29. Edit the file **cq.ini** by using gedit.

## 1 Queries exercises

### Exercise 2. Defining data collection rules

30. Review the highlighted **DSUSER1** statement in the **cq.ini** file as shown in the following screen capture.



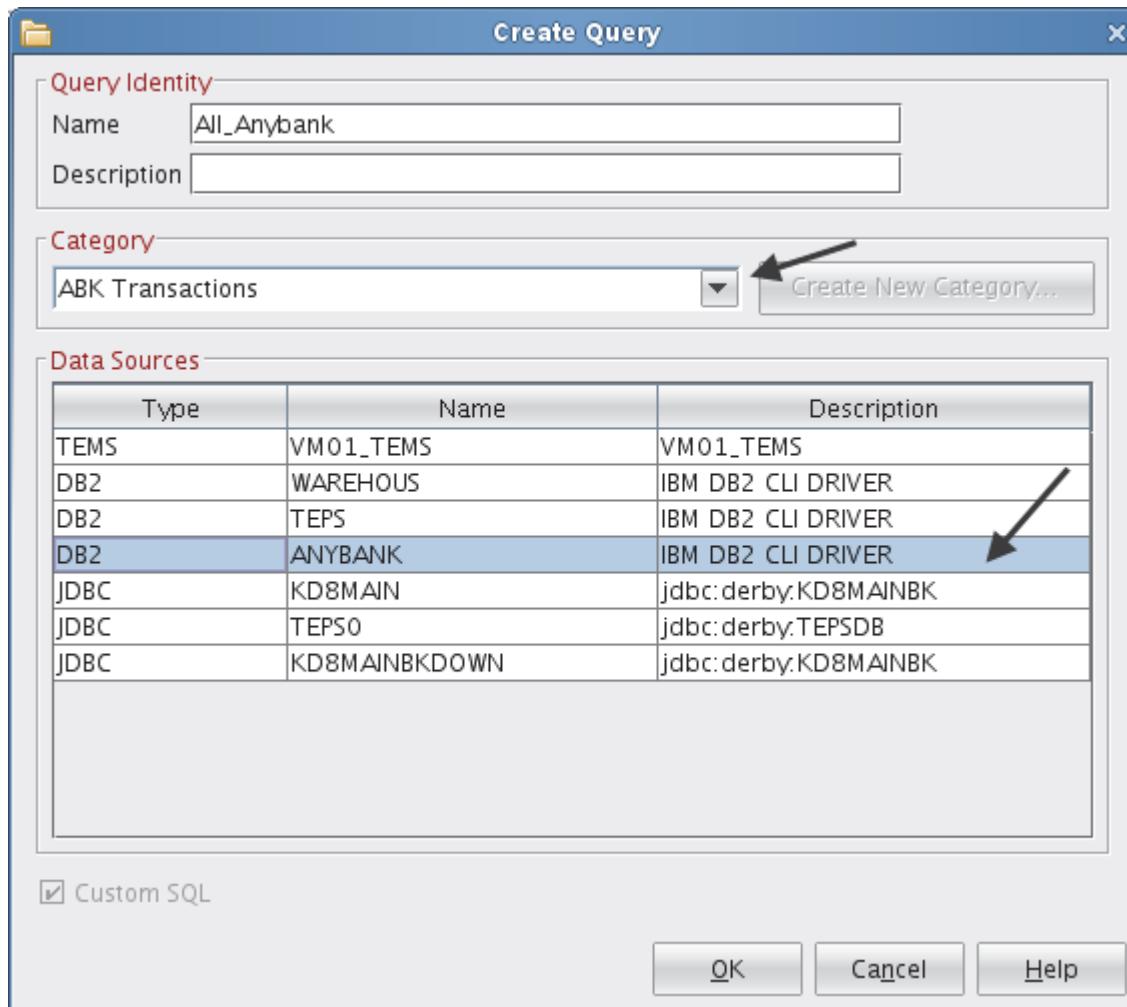
```
File Edit View Search Tools Documents Help
Open Save Undo Cut Copy Paste Find Replace
cq.ini X
ITM_BINARCH=$BINARCH$
DB2INSTANCE=$DB2INSTANCE$
KFW_DSN=$KFW_DSN_NAME$
KFW_ODBC_DSN_LIST=DSW1 DSW2 DSC3 DSC0 DSUSER1 DSUSER2 DSUSER3 DSUSER4 DSUSER5 DSUSER6 DSU:
KFW_ODBC_WAREHOUSE_LIST=DSW1 DSW2
DSW1=DSN=$WAREHOUSEDB$;UID=$WAREHOUSEID$;PWD=$WAREHOUSEPW$;CONNECTION_TYPE=$WHATYPE$;CONN
DSW2=DSN=Candle Data Warehouse;UID=$WAREHOUSEID$;PWD=$WAREHOUSEPW$;
DSC3=DSN=$DB2NAME$;UID=$DB2ID$;PWD=$DB2PW$;CONNECTION_TYPE=DB2;CONNECTION_LIMIT=100;$DB2A
DSC0=DSN=TEPSO;UID=APP;PWD=cnpncps;CONNECTION_TYPE=JDBC;KFWJDBCDRIVER=org.apache.derby.jdbc
DSUSER1=DSN=ANYBANK;UID=db2inst1;PWD=object00;CONNECTION_TYPE=DB2;CONNECTION_LIMIT=100;
DB2CODEPAGE=1208
LOGSHOME=$CANDLEHOME$/logs
BES_LIC_DIR=$CANDLEHOME$/BINARCH$/PRODUCTCODE$/bdp
BES_LIC_DEFAULT_DIR=$CANDLEHOME$/BINARCH$/PRODUCTCODE$/bdp
MALLOCTYPE=buckets
LD_LIBRARY_PATH=$ICCRTE_DIRS:$GSKLIB$:$CANDLEHOME$/BINARCH$/PRODUCTCODE$/lib:$KFW_EWAS_HOME$/
java/jre/lib/i386:$KFW_EWAS_HOME$/java/jre/bin/classic:$DB2DIR$/DB2LIB$
```

31. Close gedit and do not save the changes.

## Building the custom query against the new data source

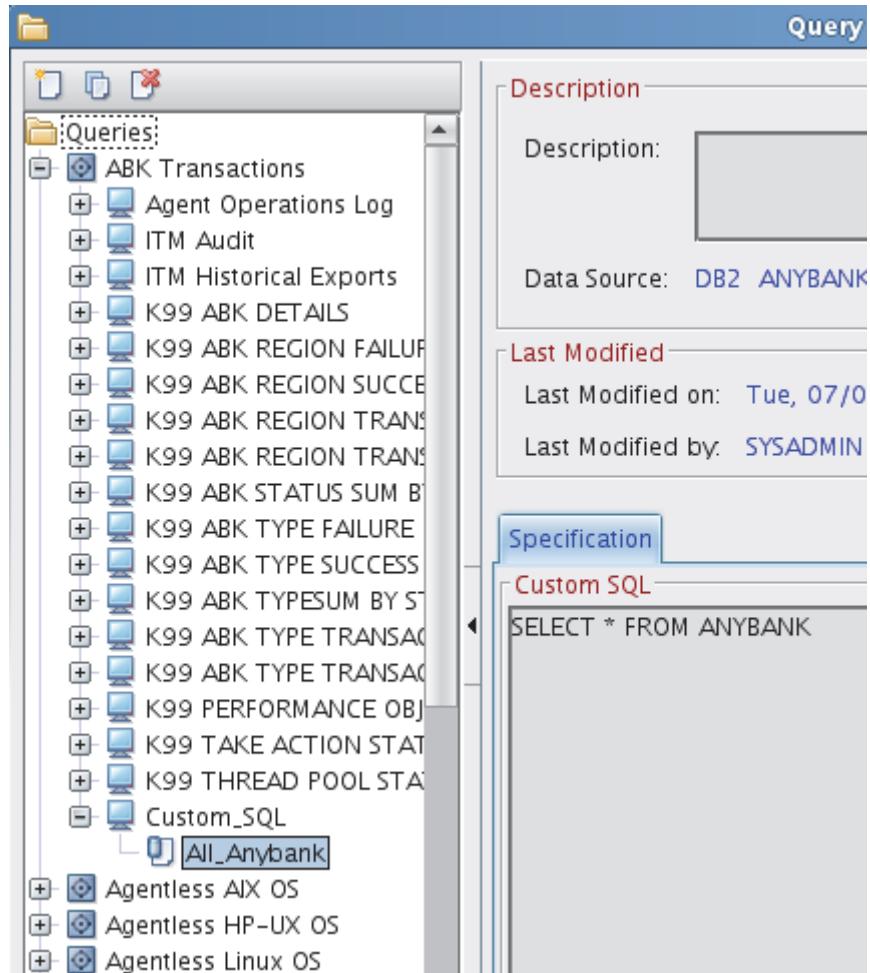
32. To build a query against the data, access the Query editor by pressing Ctrl+Q or by selecting the toolbar **Queries** icon.
33. Create a query from the Query editor.
34. Enter the name of the new query as **All\_Anybank**.

35. Select the category **ABK\_Transactions** and the data source **ANYBANK**.



The query is under a folder called **Custom SQL** in the category that you specified.

36. To specify your SELECT statement, you must know the table name of your data. You can verify that you know your data by using the IBM Data Studio administration client to open the actual ANYBANK database. In this case, the table is named ANYBANK.



37. Test your new query by including a new table view at the bottom of the **Operations > Reports** workspace. Rename the view header **All Anybank Data**.

All Anybank data				
REGION	TYPE	TRANSACTIONCOUNT	OK	FAILED
North	ATM	9827	9164	663
North	Online	15019	14863	156
North	Branch	7249	6832	417
North	CheckCard	11539	11124	415
North	Phone	8921	8506	415
South	ATM	7391	7057	334
South	Online	13094	12331	763
South	Branch	6290	6067	223
South	CheckCard	17652	16460	1192
South	Phone	6688	6424	264
West	ATM	14891	14421	470
West	Online	8427	7773	654
West	Branch	6345	6040	305
West	CheckCard	7334	7258	76
West	Phone	11309	10183	1126
East	ATM	12364	11246	1118
East	Online	13091	12820	271
East	Branch	6036	5479	557

38. Save the **Reports** workspace.



## 2 Advanced link topics exercises

In the prerequisite class, you learned how to create simple links. In this exercise, you learn how to create advanced links by using dynamic and relative links.

### Exercise 1. Creating advanced links to filter link target workspace content

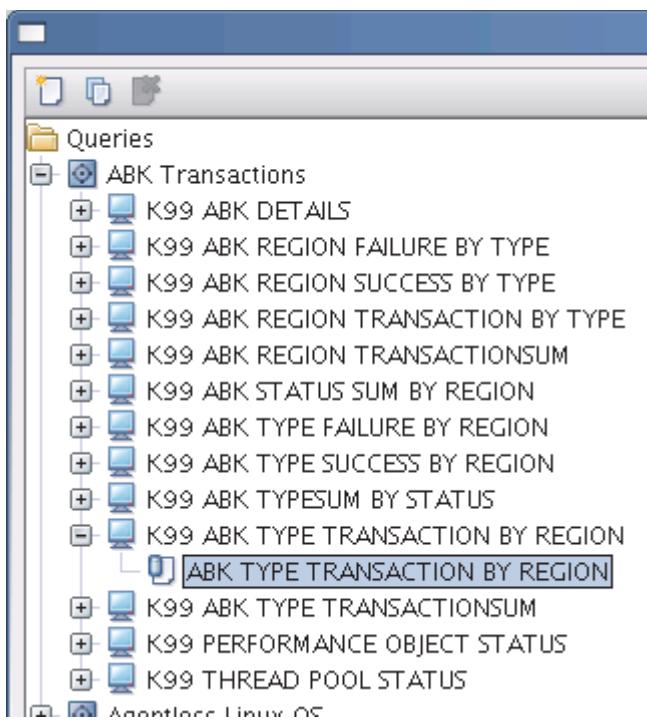
In this exercise, you have five regions and you want to view specific information for only the selected region. To do this, you create one workspace and build queries to filter the workspaces by region, instead of building five workspaces (one for each region).

You use the following steps to create an advanced link that includes link symbols:

- Create a query for the target workspace views.
- Include link symbols in the new query to dynamically define the query filter.
- Create the target workspace and assign the new query to the views that display region-specific data.
- Open the Link wizard from the source workspace to create a link.
- Define the link symbols that are used by the query. Then, when the linked workspace is opened, the value for the filter is dynamically assigned, depending on the data in the source view. For example, if you click a link next to a record for the West region, it can use the region attribute (West) to filter the results.

## Creating the query

1. Access the Query editor from the portal toolbar by clicking **Queries**  or by pressing Ctrl+Q on your keyboard.
2. Select **Queries > ABK Transactions > K99 ABK TYPE TRANSACTION BY REGION > ABK TYPE TRANSACTION BY REGION.**



3. Click the **Create Another Query** icon to make a copy.
4. Enter **ABK\_Region\_Details** as the query name to distinguish it from the original query.

## Adding link symbols to the query

Link symbols are variables that are substituted with current portal values when the link is opened. In this scenario, you want to pass the region name to the new query. This name is used to filter the information that is retrieved from the database by the specific region. With this procedure, you can create one query instead of five different queries with five different filters.

When you create the query, it is important to include link symbols to allow data from the source workspace to be passed to the query. Do not remove the **\$NODE\$** link symbol, which is already listed in the query. This symbol is used to pass the managed system name to the query. Without it, the query does not know what data source to access when it is issued.

5. Add a link symbol for the region by clicking in the first empty **Region** table cell and typing **\$ABKRegion\$**.

	Node	Timestamp	Region
1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	$\nu == \$NODE$$		$\nu == \$ABKRegion$$
3			
4			

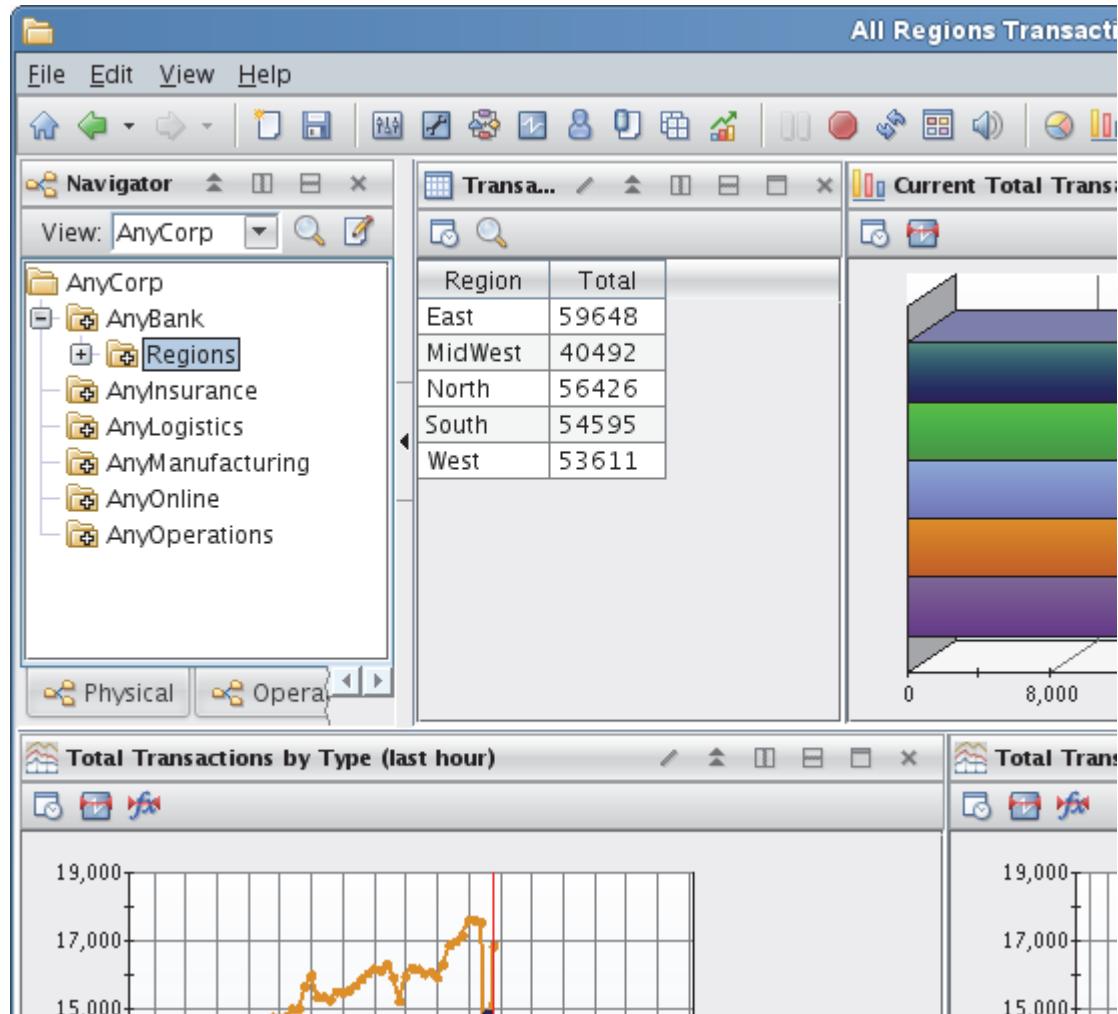
6. Click **OK** to save your changes and close the Query editor.

## Creating the target workspace

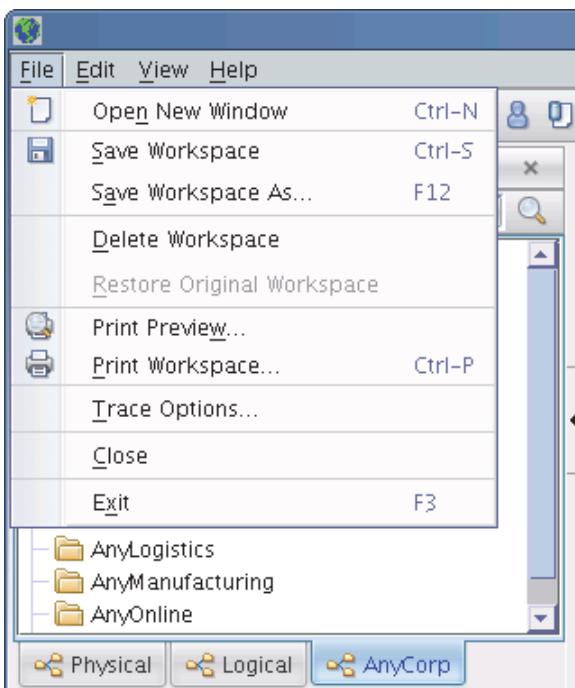
Use the query in your target workspace to filter the workspace data.

7. Verify that the current Navigator view is **AnyCorp**. If you cannot see the Navigator, click the double-box icon on the view title to restore the Navigator. Select the **AnyCorp** workspace.

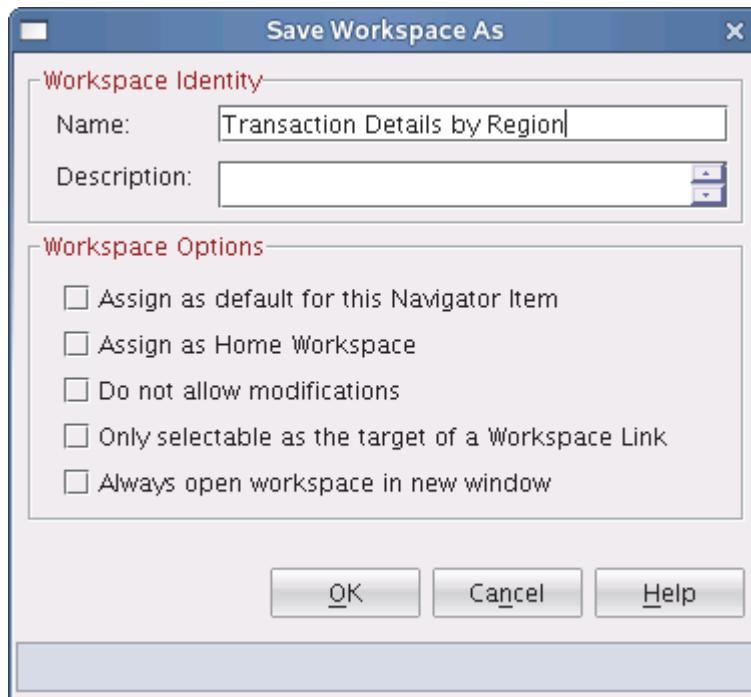
8. In the Navigator view, expand the **AnyBank** item. If you cannot see the Navigator, click the double-box icon on the view title to restore the Navigator. Select **Regions**.



9. Select **File > Save Workspace As** (or press F12) to make a copy of the existing All Regions Transaction Overview workspace.



10. Enter the name **Transaction Details by Region** and click **OK**.



11. From the workspace, delete the bar chart view in the upper right by clicking the X in that view.  
From the bottom views, delete one of the plot views.

12. The top view is a table view. Right-click inside the view and select **Properties**. For the query, assign **ABK\_Region\_Details**. Click **OK**.

	fx Node	fx Region
1	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
2	= \$NODE\$	... \$ABKRegion\$
3		
4		

13. On the **Filters** tab, select only the **Region**, **ATM**, **Branch**, **CheckCard**, **Online**, and **Phone** columns. Click **Test** to ensure that the data looks correct. Click **OK** to save the changes.

	Node	Region	Timestamp	Total	ATM	Branch	CheckCard	Online	Phone
1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2					<input checked="" type="checkbox"/>				
3						<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
4					<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

14. Define the lower view as a bar chart view by using the **ABK\_Region\_Details** query.

- On the **Filters** tab, select only the **ATM**, **Branch**, **CheckCard**, **Online**, and **Phone** columns.

The screenshot shows the 'Filters' tab of a query editor. At the top, there are three tabs: 'Query' (selected), 'Filters' (highlighted in blue), and 'Style'. Below the tabs is a grid titled 'fx' with a red border. The grid has a header row with columns labeled 'Timestamp', 'Total', 'ATM', 'Branch', 'CheckCard', 'Online', and 'Phone'. There are four rows numbered 1 through 4. In each row, the 'ATM', 'Branch', 'CheckCard', 'Online', and 'Phone' columns contain checked checkboxes, while 'Timestamp' and 'Total' are empty. The 'fx' grid is surrounded by scroll bars on the right and bottom.

	Timestamp	Total	ATM	Branch	CheckCard	Online	Phone
1	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>				
2							
3							
4							

- On the **Style** tab, enter **Transaction Totals By Region and Type** as the title.
- On the **Style** tab, click **Category Axis** below the **Plot Area**.
- Click the **Category Axis** tab.
- Change the attribute from **Default Category Label** to **Region**. Click **OK** to save it.

15. Verify that your workspace is like the following screen capture.

Region	ATM	Branch	CheckCard	Online	Phone
East	8092	9645	12904	7793	11472
MidWest	11937	13399	7169	13234	12972
North	5681	6743	12952	9348	6117
South	14246	5812	14239	13597	14004
West	13996	10879	12536	11812	6879

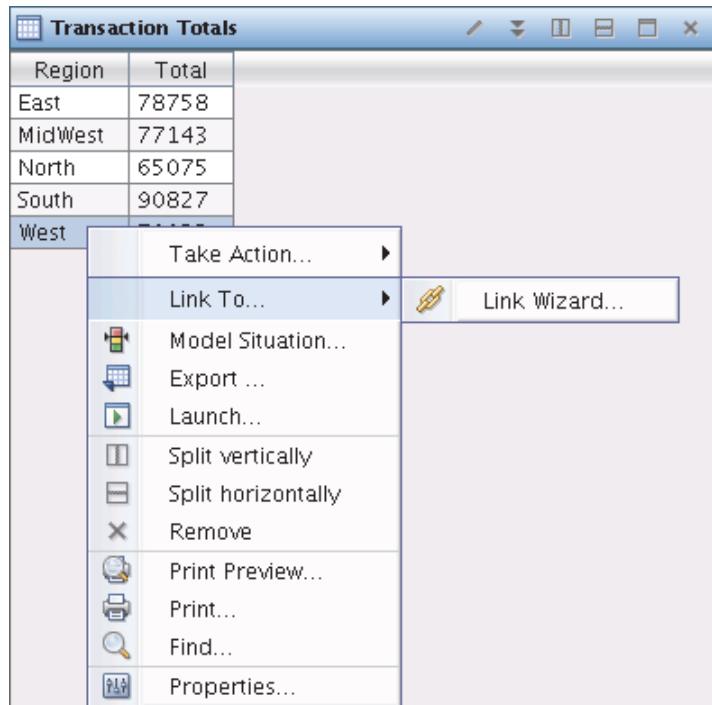
**Note:** You do not need to rename your view headers because you configure them to be automatically substituted during the link process.

16. Save the workspace.

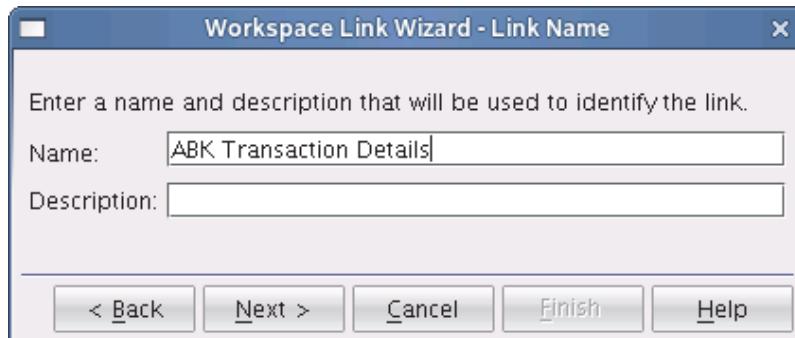
## Creating the link

17. Right-click the **Regions** Navigator item and select **Workspace > All Regions Transaction Overview**. This workspace is the source workspace for your link.

18. Right-click a row of data within the **Transaction Totals** table view and select **Link to > Link Wizard**.

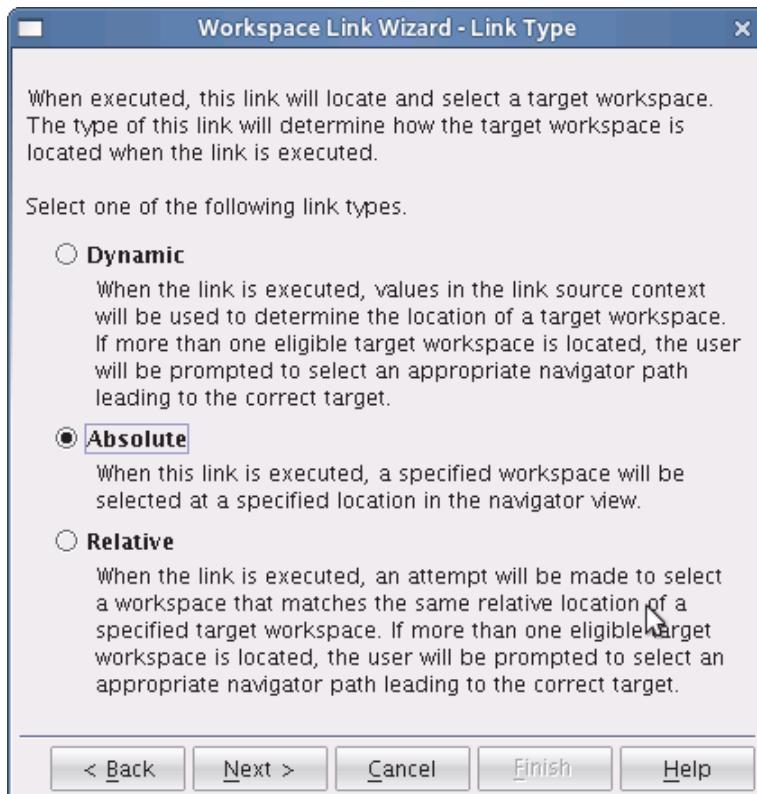


19. Create a new **Absolute** link and name it **ABK Transaction Details**.

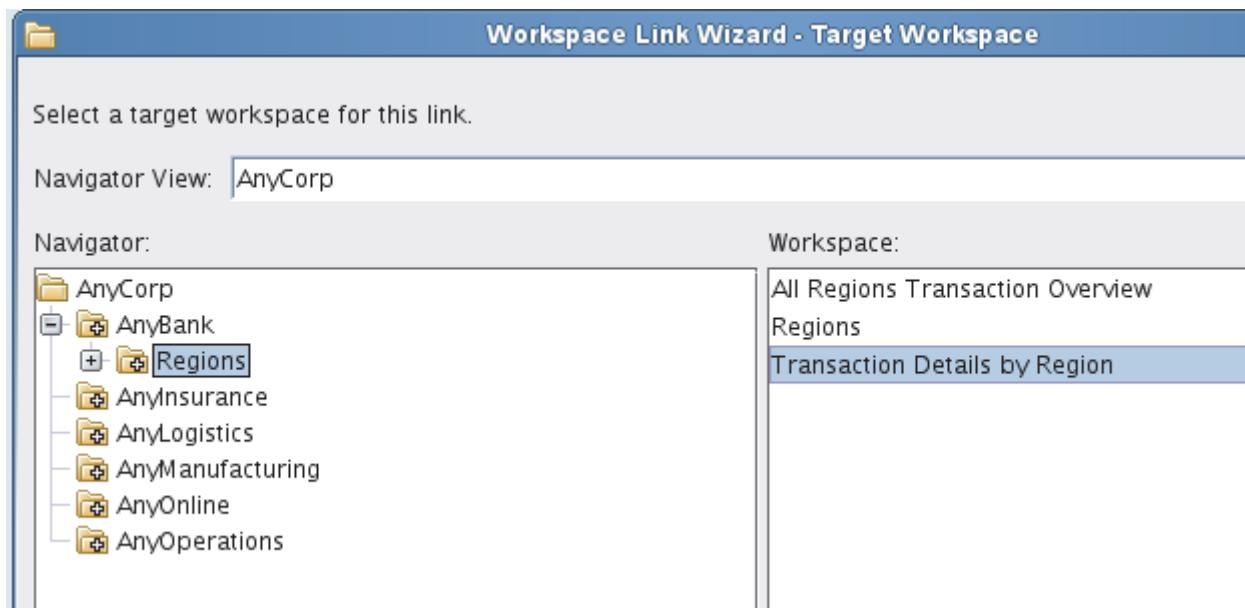




**Note:** You want to select an **Absolute** link because you have one link target. There is no need to set the link target as dynamic. The dynamic part of the link is the filtering of the query with the link symbol.

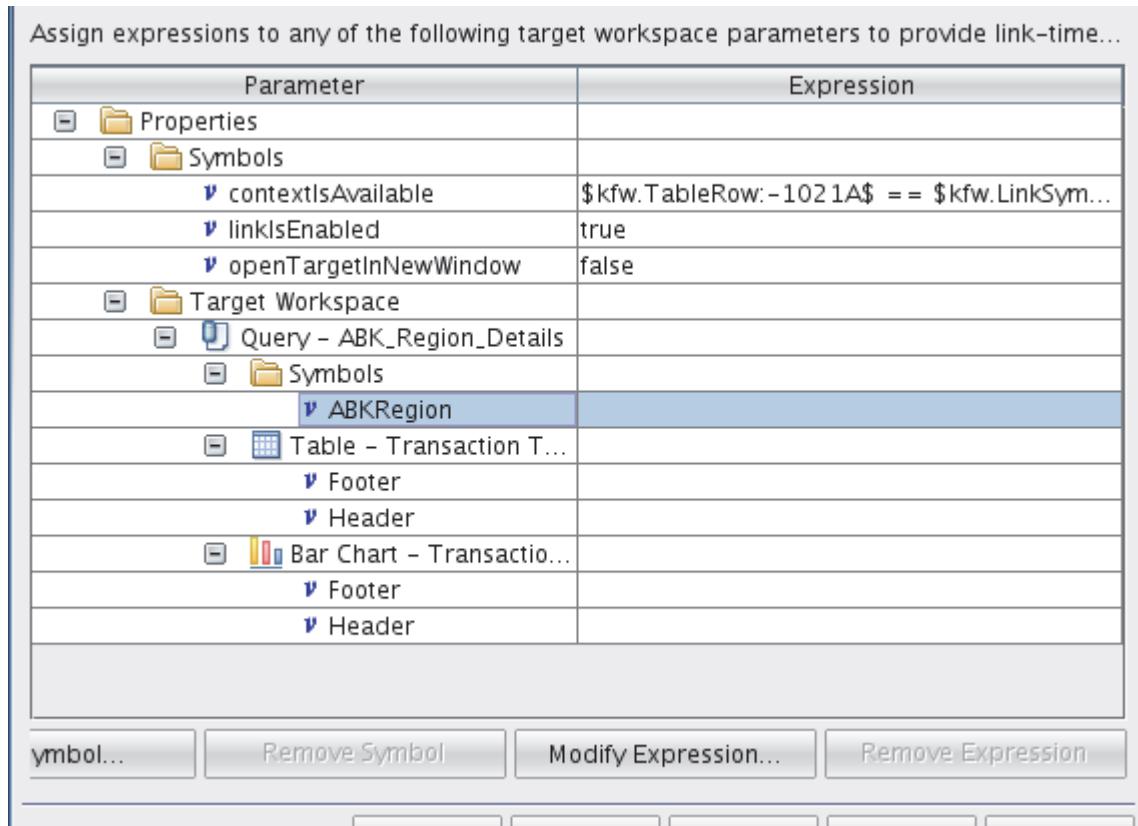


20. Select the workspace that you created from **AnyCorp > AnyBank > Regions > Transaction Details by Region** and click **Next** to modify the link parameters.



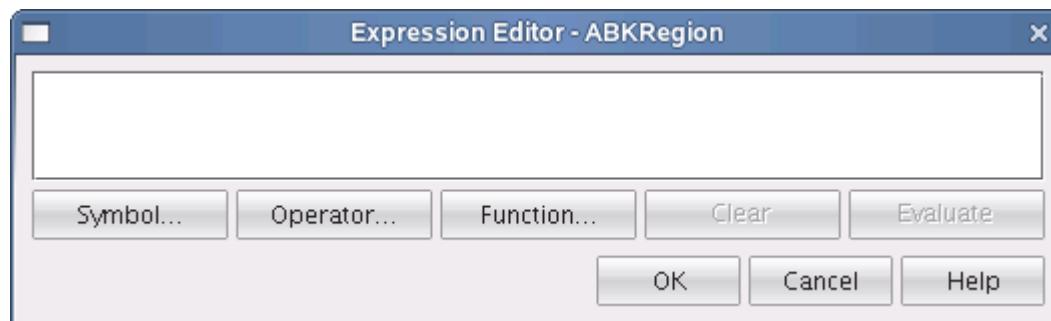
**Note:** If you do not see **AnyCorp** Navigators listed, you must select the **Physical** Navigator view and then select the **AnyCorp** Navigator view.

21. Notice that **Workspace Link Wizard - Parameters** includes parameter entries for the primary components of the target workspace, the query, and both views. Also, notice the **ABKRegion** link symbol that you created is listed under **Query - ABK\_Region\_Details** and **Symbols**.



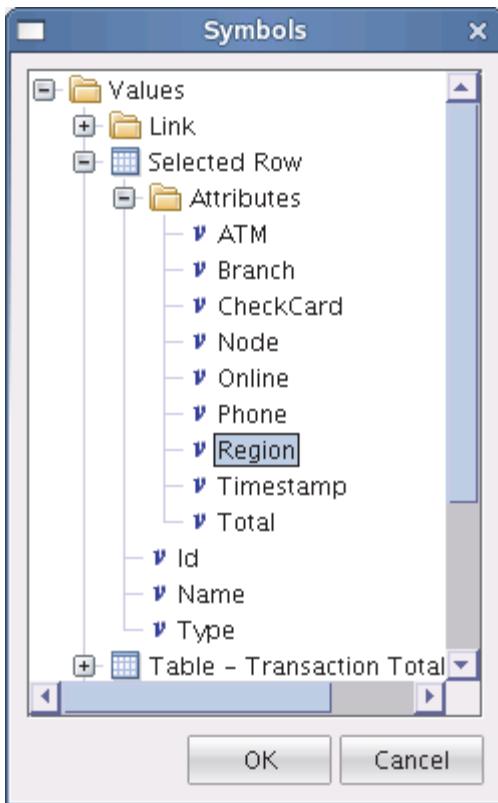
You can dynamically define the value to use for the query filter.

22. Create an expression that pulls the region attribute from the table in the source workspace. Select the **ABKRegion** link symbol in the parameter column under the **ABK\_REGION\_DETAILS** query.
23. Click **Modify Expression** to access the Expression editor and assign values and expressions to your link symbol.



24. Click **Symbol** to select the symbol for your expression.

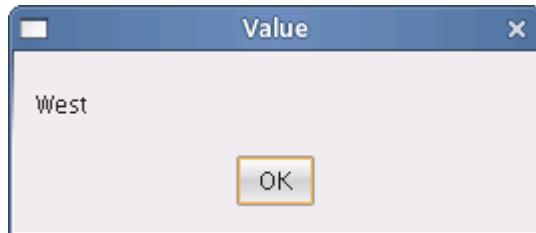
25. Select the **Region** attribute and click **OK** to save the symbol.



26. Click **Evaluate** to verify the expression. This evaluation shows the region that you selected in the table when you started the link creation.



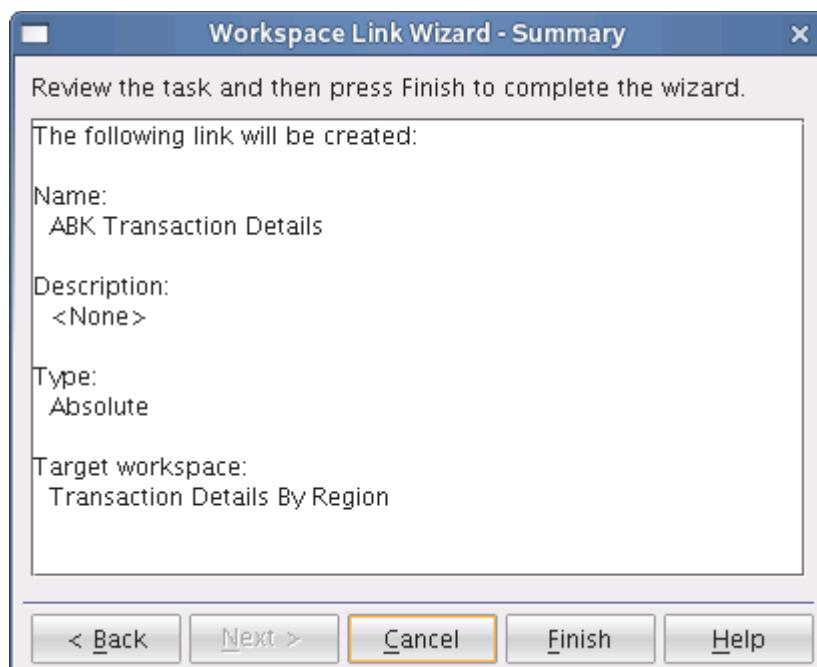
27. Click **OK** to exit the Value window. Click **OK** to exit the Expression editor.



The expression in the parameters menu is populated.



28. Click **Next** to review your link summary.



29. Click **Finish** to complete the link setup.

The table looks like the following example.

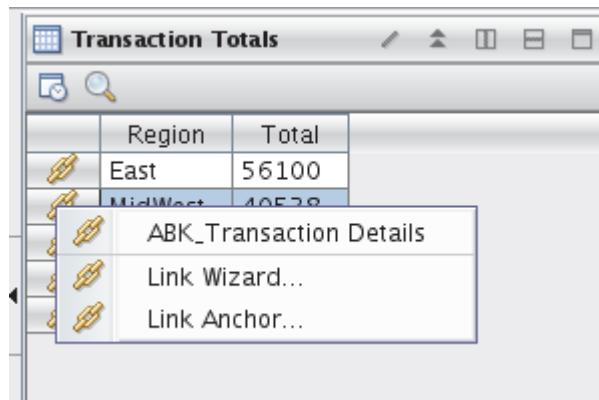
	Region	Total
	East	54129
	MidWest	43508
	North	58551
	South	56743
	West	56361

## 2 Advanced link topics exercises

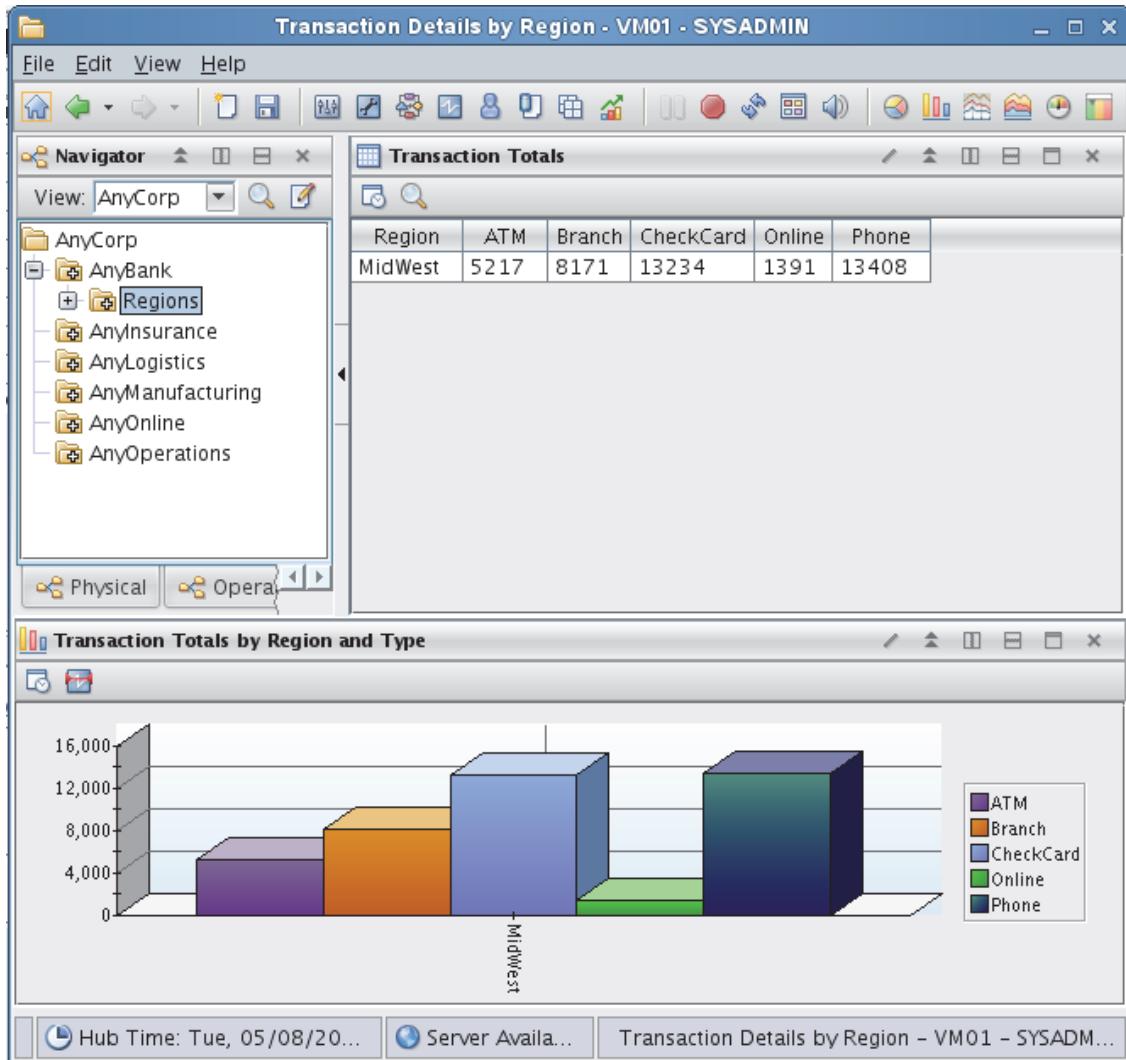
### Exercise 1. Creating advanced links to filter link target workspace content

30. Save your workspace.

31. Test the link by clicking the link next to any table row in the Transaction Totals table and selecting **ABK Transaction Details**.

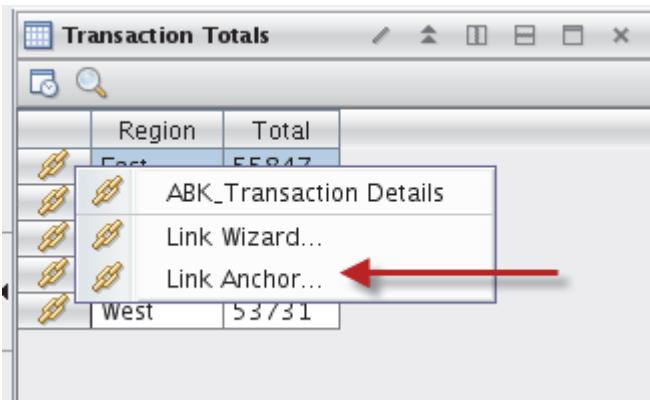


The linked workspace is opened and the expression value (Region) is substituted in the target workspace query. When you complete this step, you filter the query by the region list in the table row. The values displayed in the workspace views are only for the selected region.



32. Click **Back** in the portal toolbar.
33. Go back and forth between your source workspace and your target workspace. Click the link from different table rows, and use the **Back** button to navigate back from the target workspace.
34. Anchor the link to the **ABK Transaction Details** link so that you do not have to select it every time. Right-clicking from a cell in the table produces the options that are displayed in the

example. Right-clicking the link symbol displays fewer options, but also includes the **Link Anchor** choice.



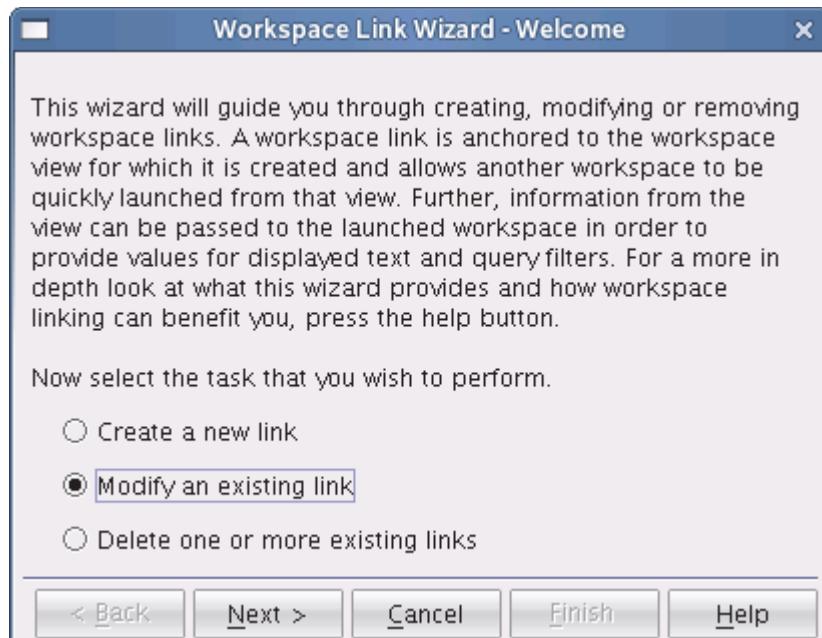
35. Save your workspace.

## Header and footer link symbols

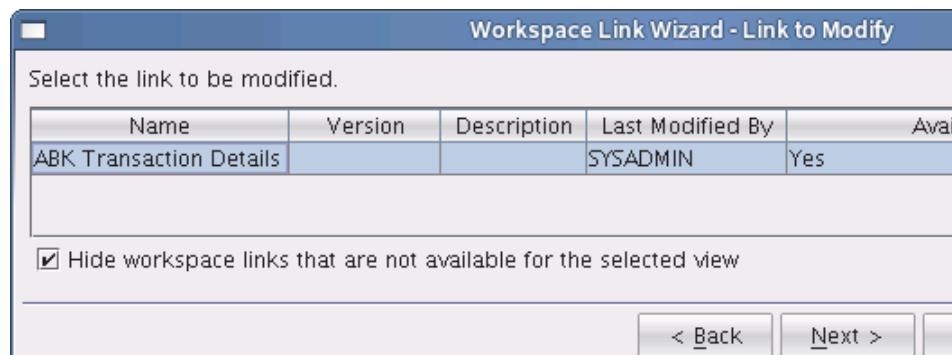
In the previous section, you created a link to dynamically modify the context of a target workspace based on data from the source workspace. The view content changes in your target workspace. The default text is displayed in the headers and footers of your target workspace views. This text can also be dynamically generated.

36. To define the substitutions for the header and footer in that view, open the source workspace **All Regions Transaction Overview**. Open the Link wizard by right-clicking a link icon in the table view and selecting **Link Wizard**.

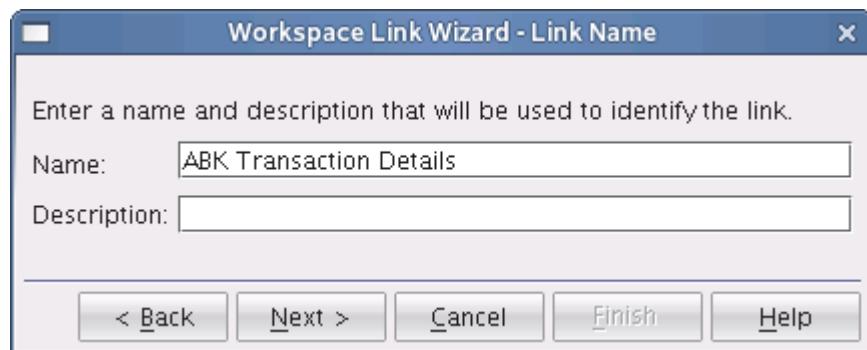
37. Select **Modify an existing link** and click **Next**.



38. Select the link that you created and click **Next**.



39. Do not change the name. Click **Next** to define the link parameters.

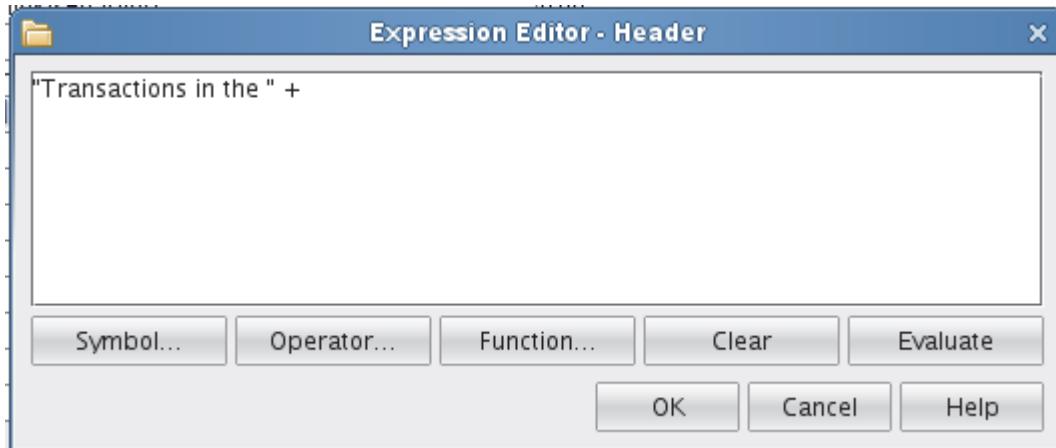


40. There is a header for each embedded view. Under **Table - Transaction Totals**, select the header for the Table view and click **Modify Expression** to open the Expression editor.

*Exercise 1. Creating advanced links to filter link target workspace content*

41. Type the following string concatenation expression. Any static text that you want to display in your headers must be entered in quotation marks.

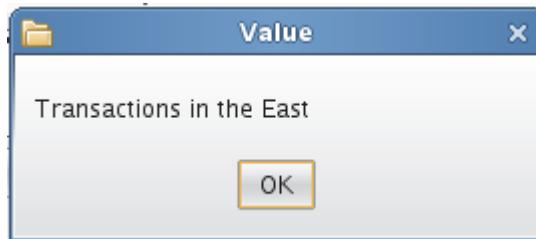
"Transactions in the " +



42. Click **Symbol** and select the **Region** attribute. This symbol substitutes the region name in your string concatenation expression.



43. Click **Evaluate** to see a preview of the header.



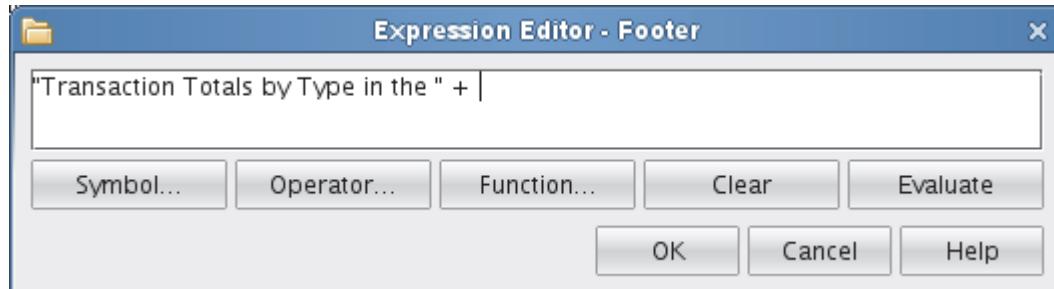
44. Click **OK** to exit the Value window. Click **OK** to exit the Expression Editor - Header window.

There is a footer for each embedded view.

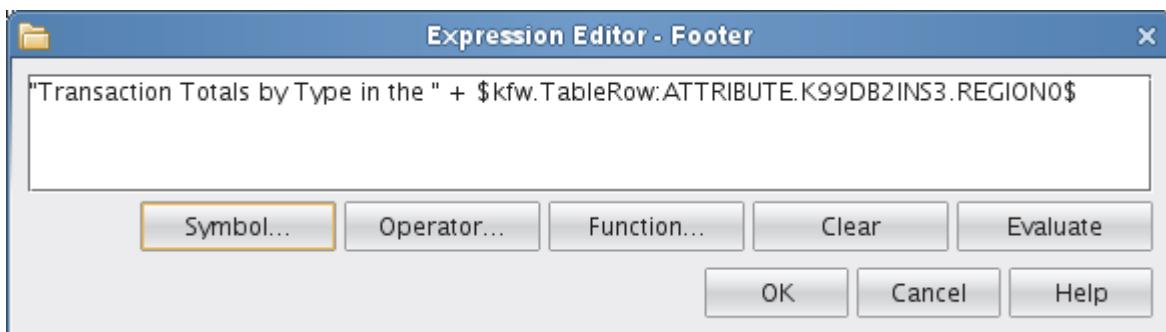
45. Under **Table - Transaction Totals**, select the footer for the Table view and click **Modify Expression** to open the Expression editor.

46. Type the following string concatenation expression. Any static text that you want to display in your footers must be entered in quotation marks.

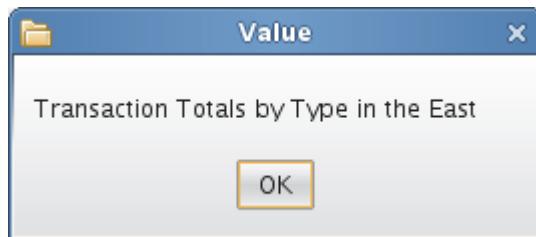
"Transaction Totals by Type in the " +



47. Click **Symbol** and select the **Region** attribute. This symbol substitutes the region name in your string concatenation expression.

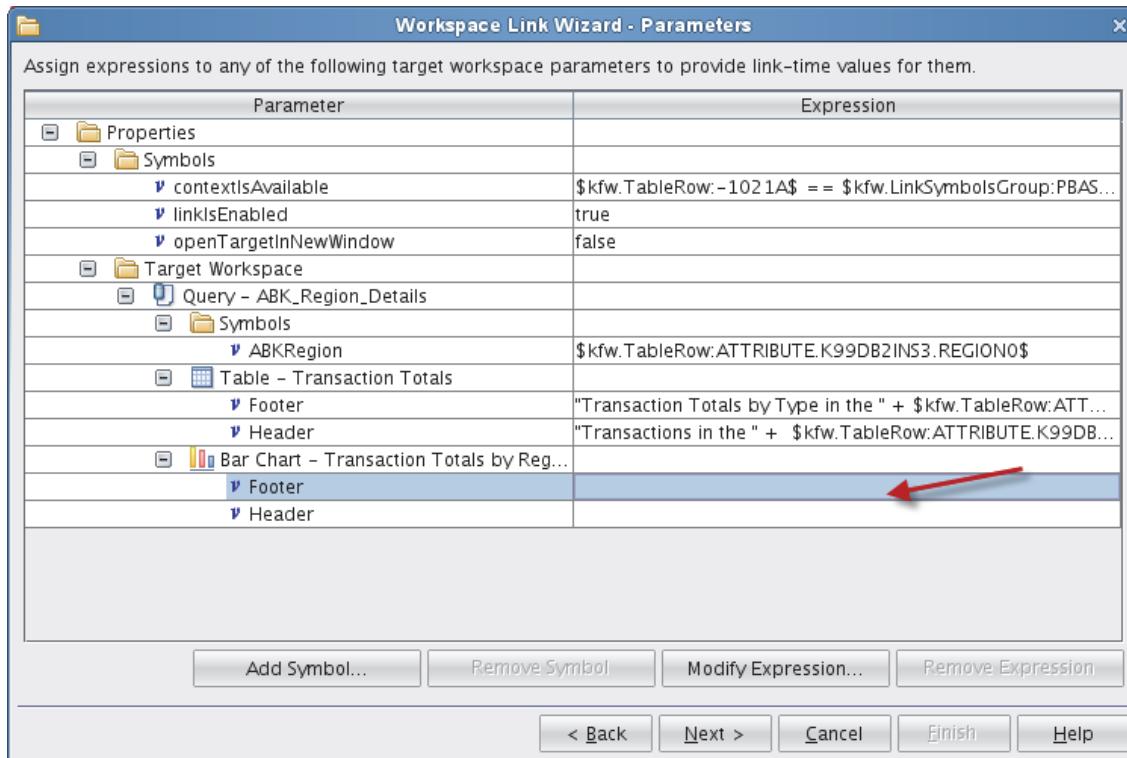


48. Test the result by clicking **Evaluate**. Click **OK** to exit the Value window.



49. Click **OK** to close the Expression editor.

50. Select the footer for the bar chart view and click **Modify Expression** to open the Expression editor again.

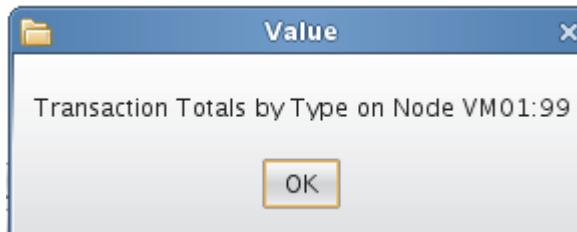


51. Modify the expression for the bar chart footer. Add the following text:

"Transaction Totals by Type on Node " +

52. Insert the symbol for the node after the static text.

53. Evaluate the expression, which looks like the following screen capture.



54. Click **OK**.

55. Modify the expression for the bar chart header. Add the following text:

"Transactions on Node " +

56. Insert the symbol for the node after the static text.

57. Click **Next** and click **Finish** to save your link and test it from your source workspace. Your dynamic headers and footers are substituted in your target workspace views.

The screenshot shows the IBM SPSS Modeler interface with two workspaces open:

- Top Workspace:** Title: Transaction Details by Region - VM01.tivoli.edu - SYSADMIN
  - Navigator View:** Shows a tree structure with nodes like AnyCorp, AnyBank, Regions, East, MidWest, North, South, West, AnyInsurance, AnyLogistics, and AnyManufacturing. The 'Regions' node is selected.
  - Table View:** Title: Transactions in the North
 

Region	ATM	Branch	CheckCard	Online	Phone
North	4983	7365	14112	8309	7430
  - Chart View:** Title: Transaction Totals by Type in the North
 

A 3D bar chart showing transaction totals for different categories. The Y-axis ranges from 0 to 16,000. The X-axis is labeled 'North'. The legend indicates five categories: ATM (purple), Branch (orange), CheckCard (blue), Online (green), and Phone (dark blue).
- Bottom Workspace:** Title: Transaction on Node VM 01:99
  - Table View:** Title: Transaction on Node VM 01:99
 

Region	ATM	Branch	CheckCard	Online	Phone
North	4983	7365	14112	8309	7430
  - Chart View:** Title: Transaction Totals by Type on Node VM01:99
 

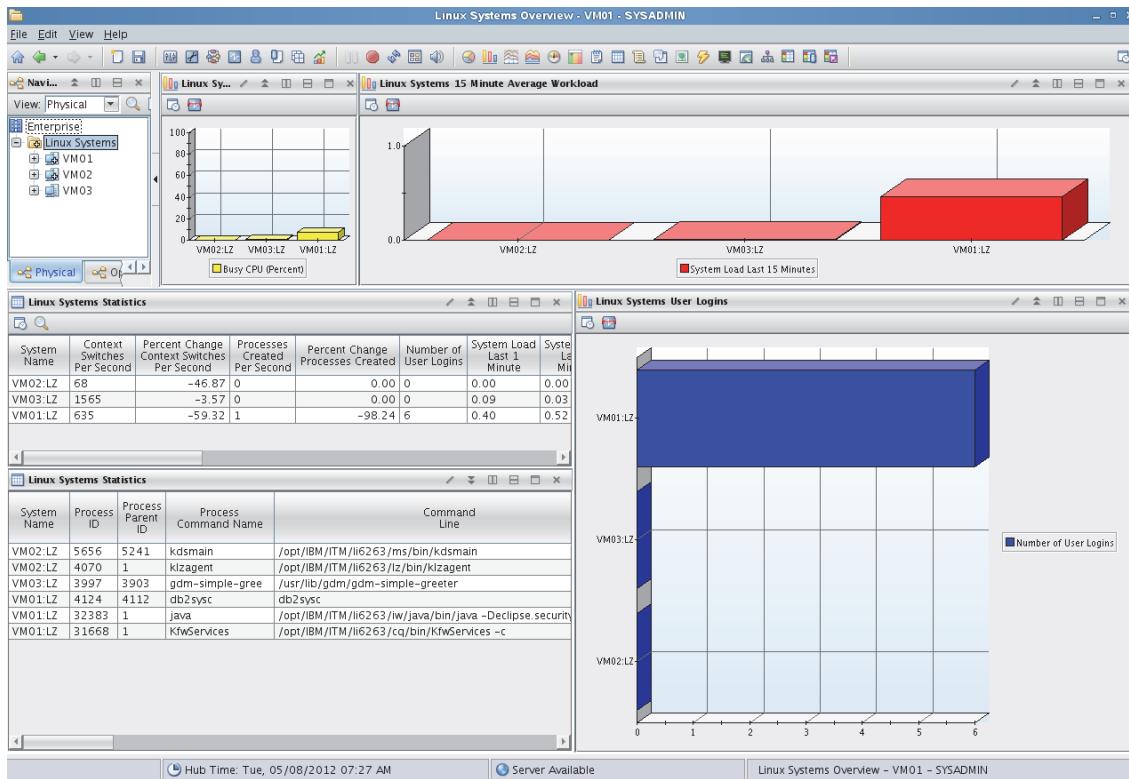
A 3D bar chart showing transaction totals for different categories. The Y-axis ranges from 0 to 16,000. The X-axis is labeled 'North'. The legend indicates five categories: ATM (purple), Branch (orange), CheckCard (blue), Online (green), and Phone (dark blue).

58. Try some of the links from different regions to see the region and node name substitutions in the headers and footers.

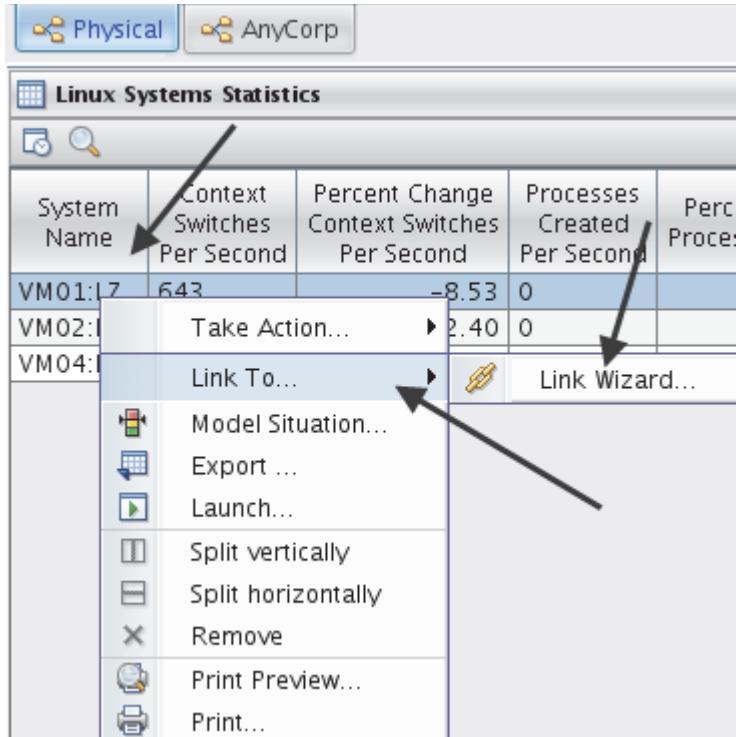
## Exercise 2. Creating dynamic links to dynamically select a link target

You created links that have a one-to-one relationship between the source workspace and the target workspace. When you open the linked workspace from one workspace, it always opens one specific target workspace. However, you might want to open different target workspaces depending on data from the source workspace. This exercise demonstrates how to create a dynamic link for this situation.

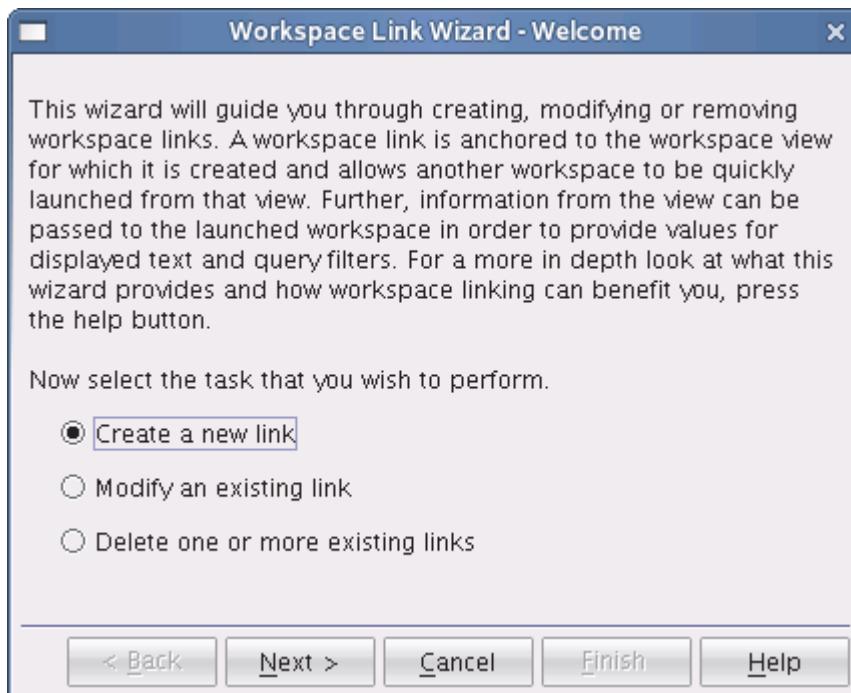
1. Switch to the Physical Navigator view.
2. Select **Linux Systems** to open the Linux Systems Overview workspace. Each view within the workspace contains information from all active Linux Operating System agents in your Tivoli Monitoring environment. These systems are typically **VM01**, **VM02**, and **VM04**.



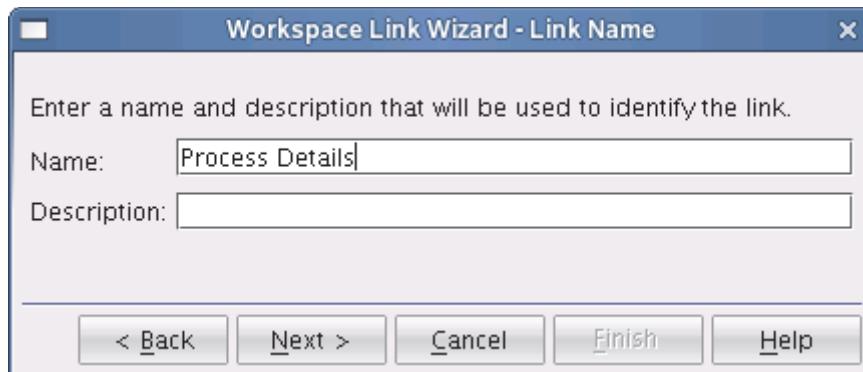
3. To create a link that opens a workspace with more detailed information about the overview data, the link target must be dynamic. Create a link by right-clicking a row in the **Linux Systems Statistics** view and selecting **Link to > Link Wizard**.



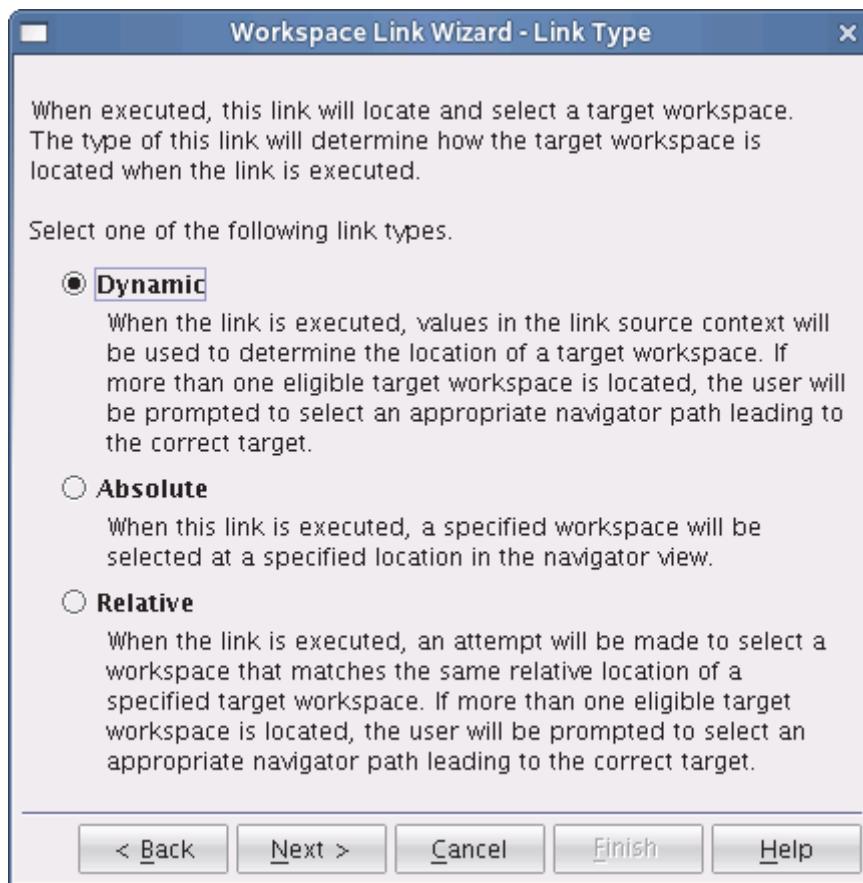
4. Select **Create a new link** and click **Next**.



5. Name the link **Process Details** and click **Next**.

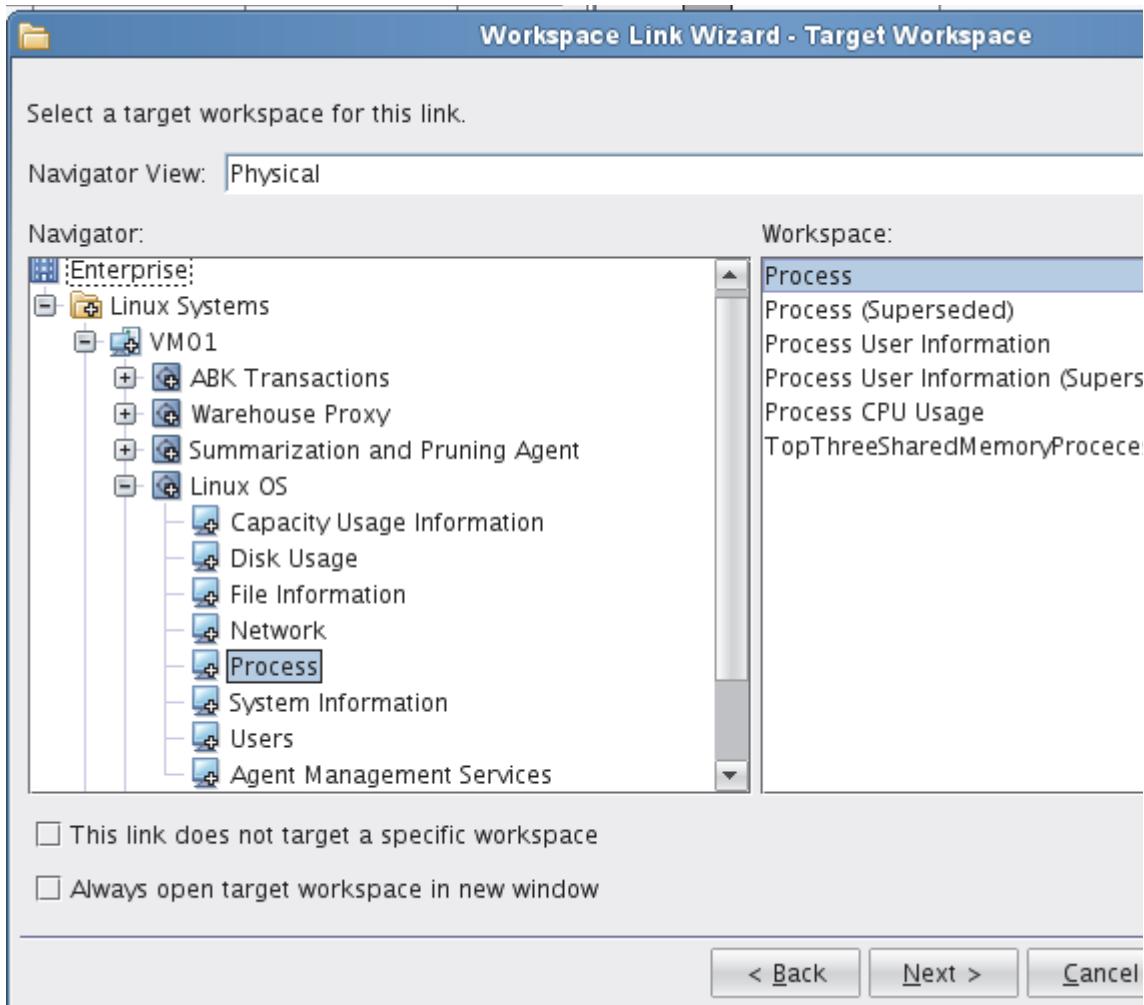


6. Select **Dynamic** for the link type and click **Next**. This type allows you to keep the link target workspace dynamic, based on certain criteria.



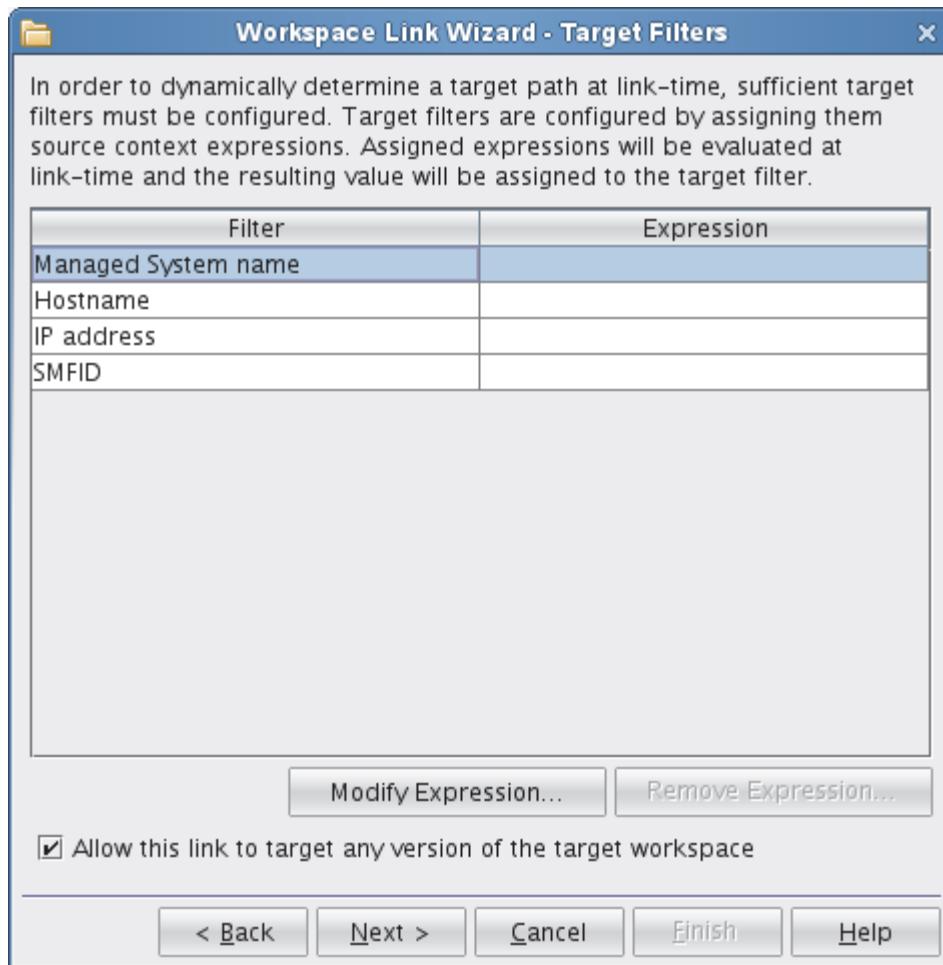
7. For the target workspace, select the **Process** workspace under the **Enterprise > Linux Systems > VM01 > Linux OS > Process** Navigator item. You can select the Navigator item from any of the Linux OS Managed Systems. Because this link is dynamic, you identify only the type of workspace to open as the link target. Depending on the data that is selected in the

source workspace, the correct node is opened by the link.

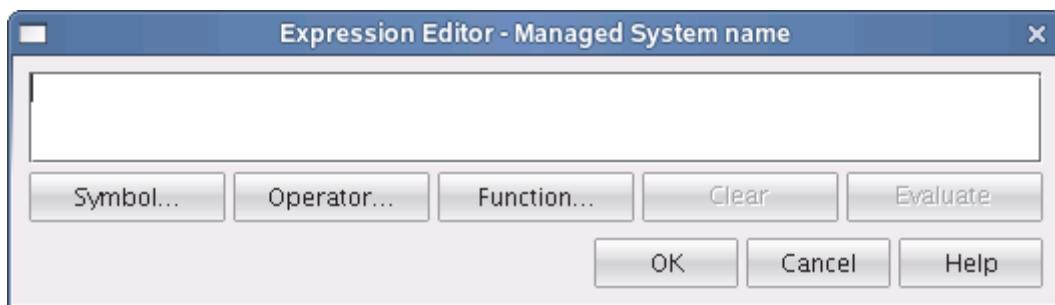


8. Click **Next** to define the Target Filters list.

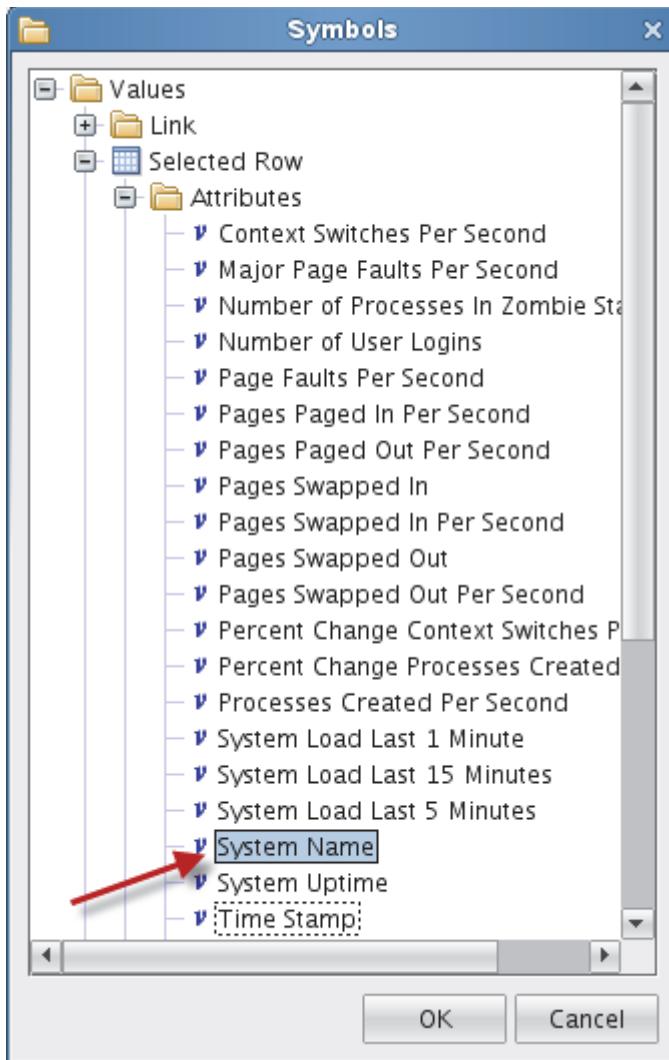
9. Select **Managed system name** and click **Modify Expression**.



10. Click **Symbol** to select a symbol from the source context.



11. Select the **System Name** attribute and click **OK**.



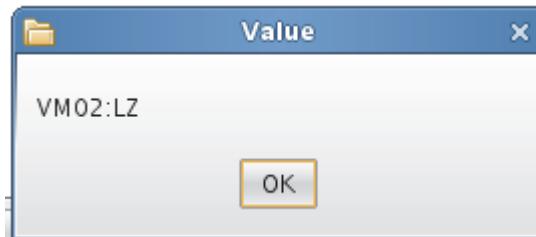
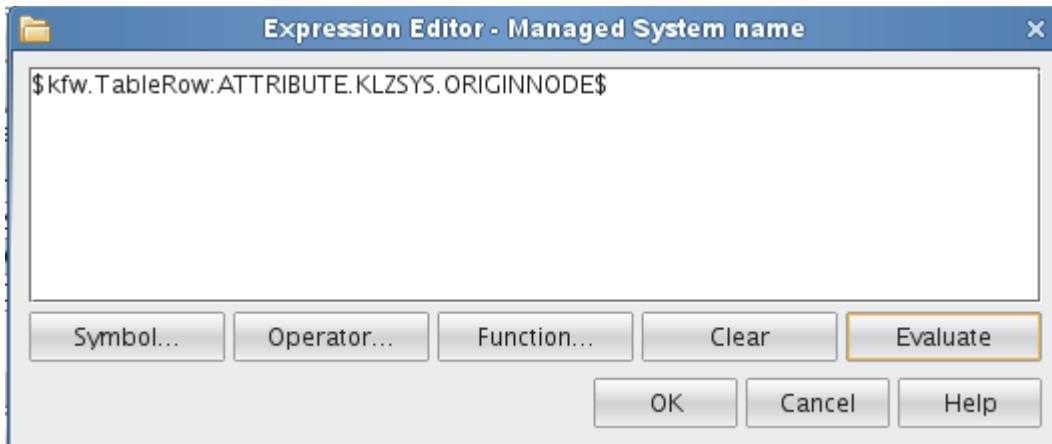
12. Test the expression by clicking **Evaluate**.

This test displays the name of the Linux system that is associated with the bar that you right-click when creating the link. The managed system name is used as the criteria to filter all possible target workspaces. In the Navigator Physical view, you access the Process workspace that corresponds

## 2 Advanced link topics exercises

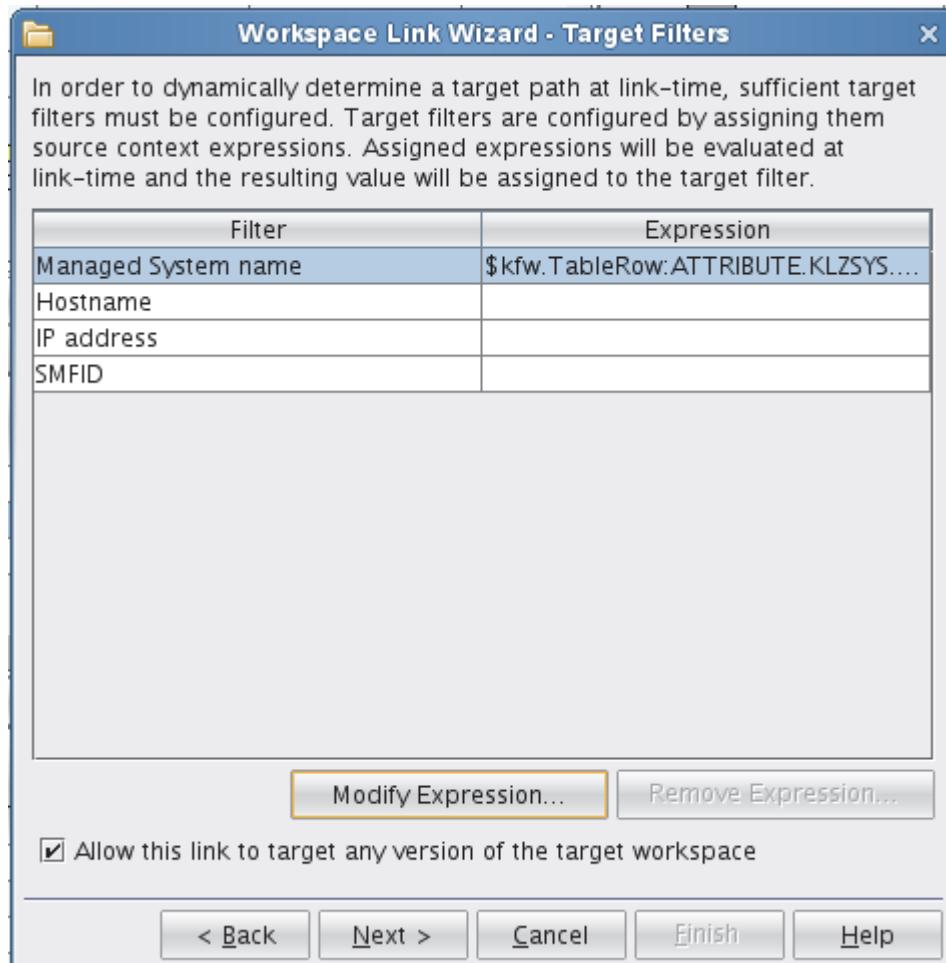
### Exercise 2. Creating dynamic links to dynamically select a link target

to the managed system for which the link was opened.

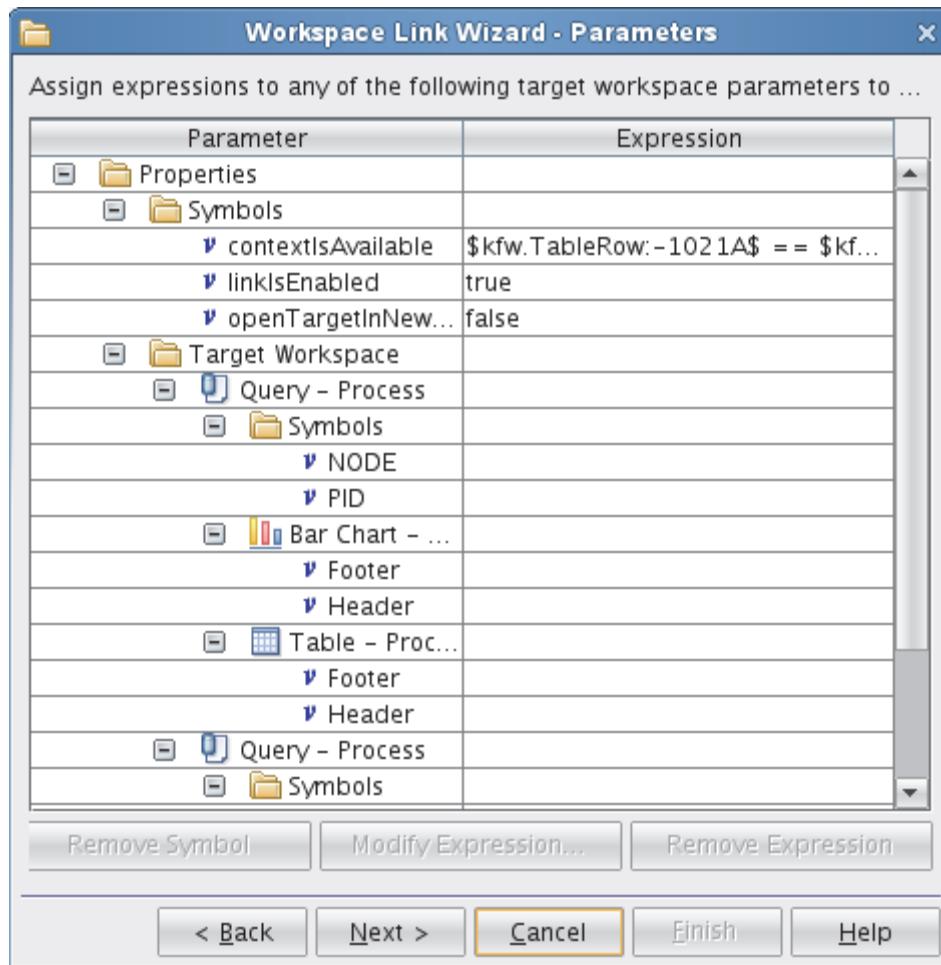


13. Click **OK** to close the Value window. Click **OK** to exit the Expression editor window.

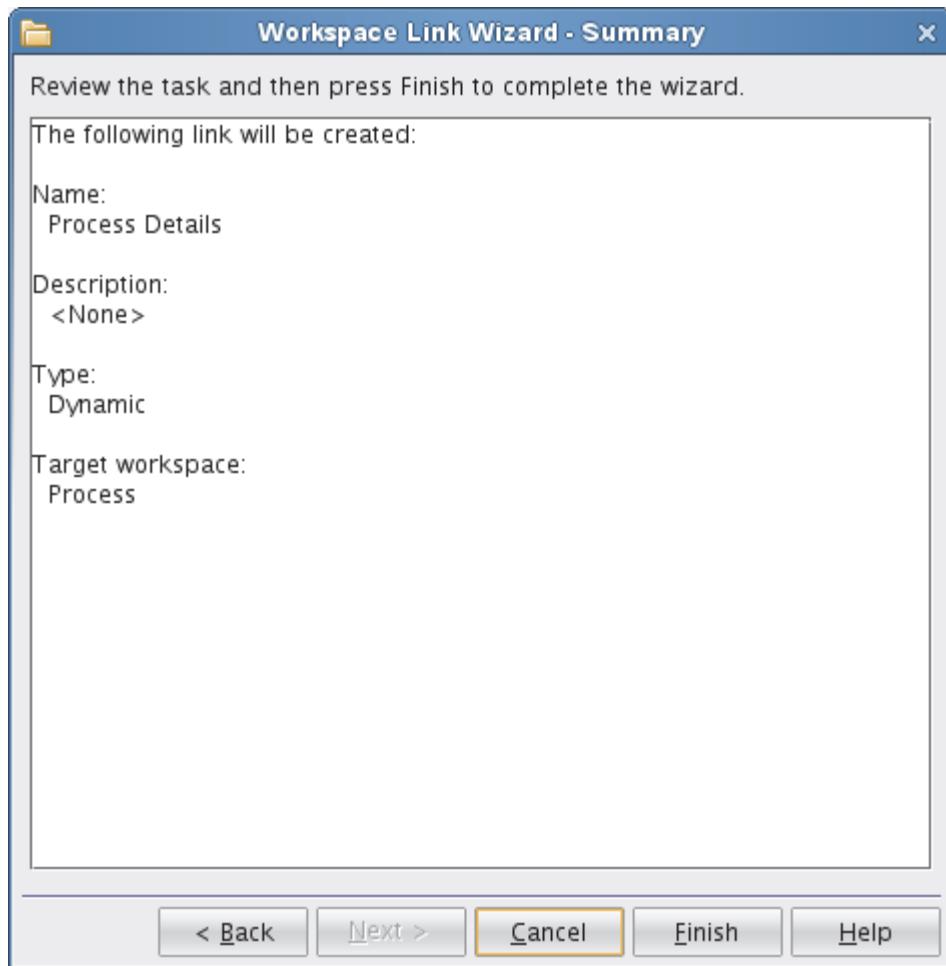
14. From the Workspace Link Wizard - Target Filters window, click **Next**.



15. From the Workspace Link Wizard - Parameters window, click **Next**.

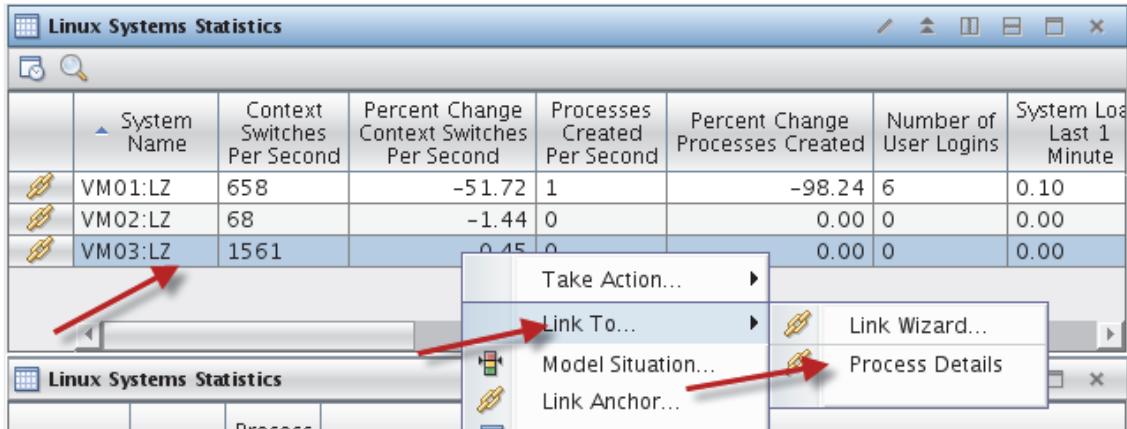


The Workspace Link Wizard - Summary is displayed.



16. Click **Finish**.

17. Test the link by right-clicking a row in the Linux System Statistics view and selecting **Link To > Process Details**.



You can verify from the expansion of the Navigator that the link goes to the selected managed system. The Process workspace for the managed system is highlighted. The heading at the top of

**Exercise 2. Creating dynamic links to dynamically select a link target**

the workspace always shows VM01. It is the name of the Tivoli Enterprise Portal Server that you are logged in to.

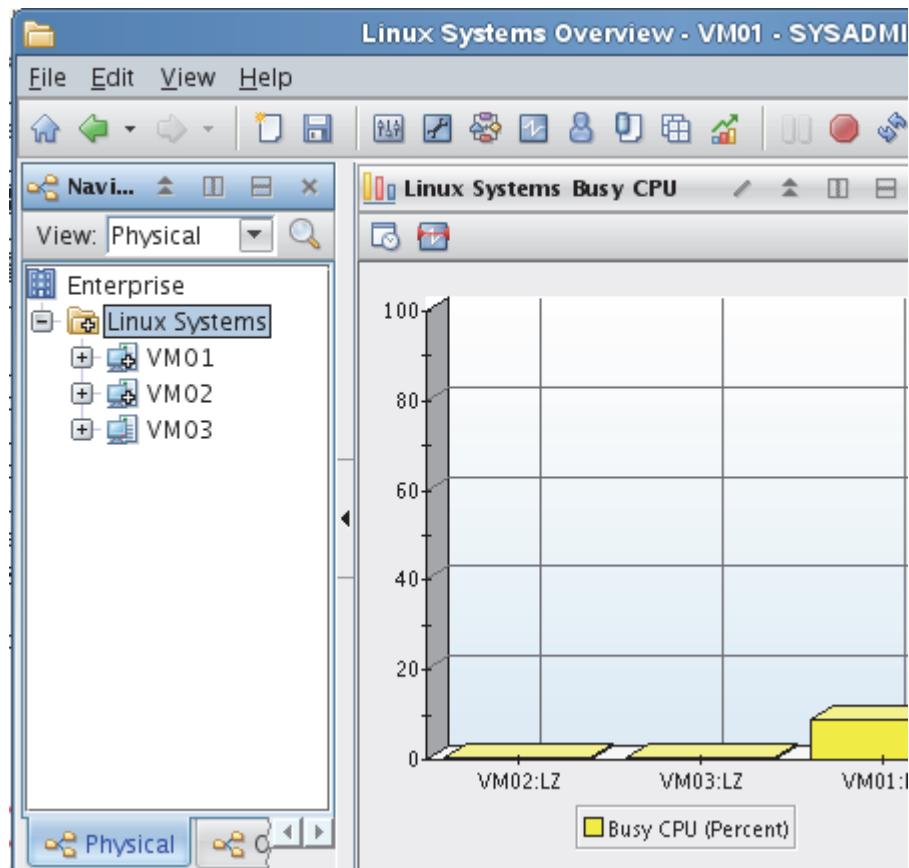
The screenshot shows the IBM Tivoli Monitoring 6.3 Advanced Administration interface. The Navigator pane on the left displays a tree view of system resources under the 'Enterprise' category. Under 'Linux Systems', nodes for 'VM01', 'VM02', and 'VM04' are visible, with 'VM04' expanded to show its 'Linux OS' and 'Process' sub-nodes. Two arrows point from the text 'the workspace always shows VM01' to the icons for 'VM01' and 'Process'. Below the Navigator is a toolbar with various icons. The bottom navigation bar has tabs for 'Physical' and 'AnyCorp', with 'Physical' currently selected. The main content area is titled 'Process Information Detail' and contains a table with the following data:

	Process Command Name	Process ID	Process Parent ID	Process State	Process System CPU (Percent)	Process User CPU (Percent)	Cu
	init	1	0	Sleeping	0.00	0.00	
	kthreadd	2	0	Sleeping	0.00	0.00	
	migration/0	3	2	Sleeping	0.00	0.00	
	ksoftirqd/0	4	2	Sleeping	0.00	0.00	
	events/0	5	2	Sleeping	0.00	0.00	

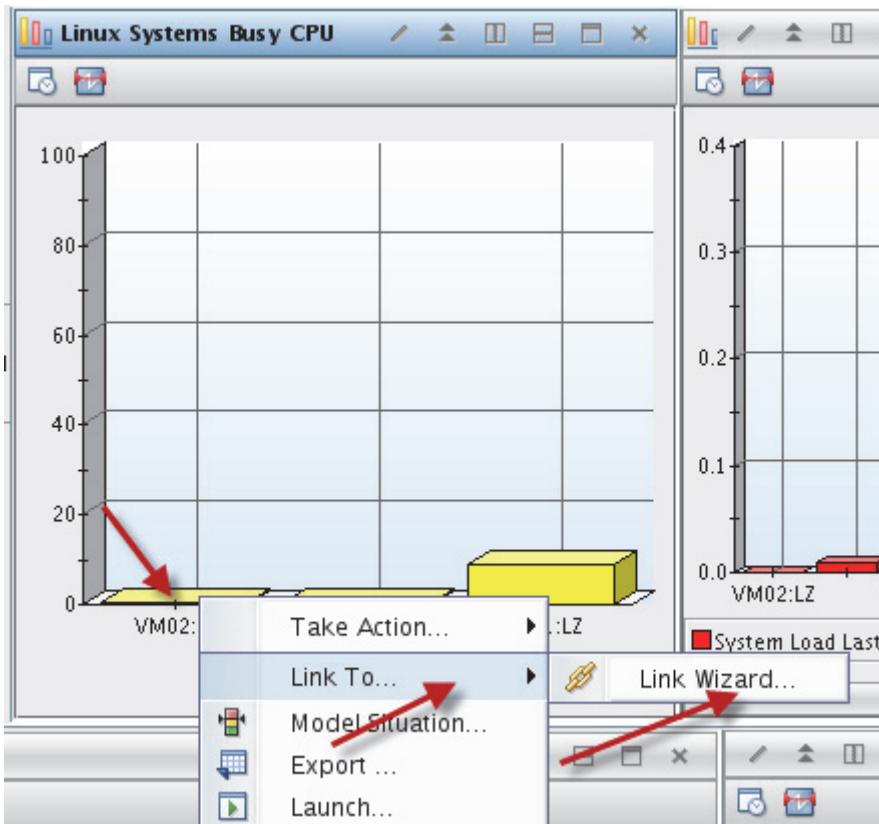
## Exercise 3. Creating a relative link

With relative links, you select a workspace target. The target can be relative to a specific managed system.

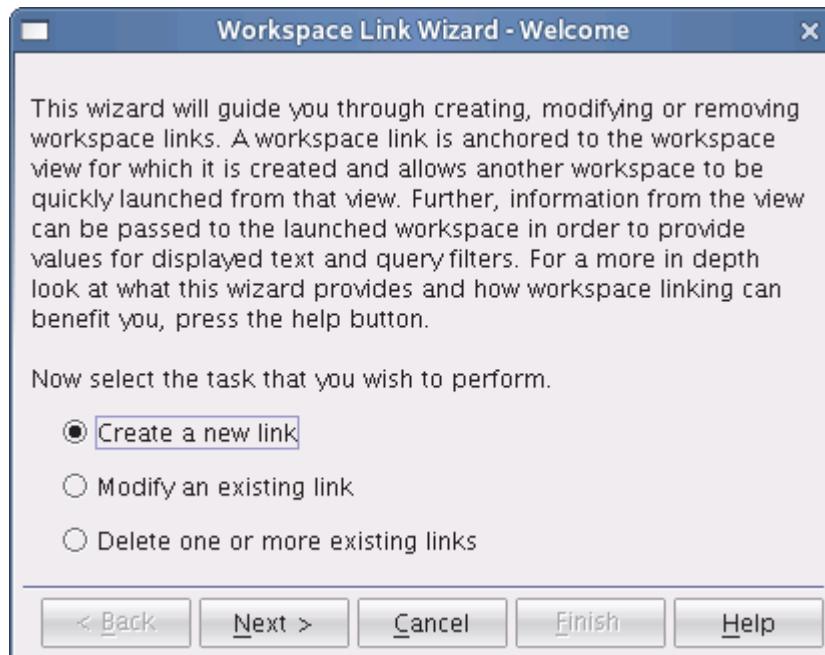
1. Select the **Enterprise > Linux Systems** workspace.



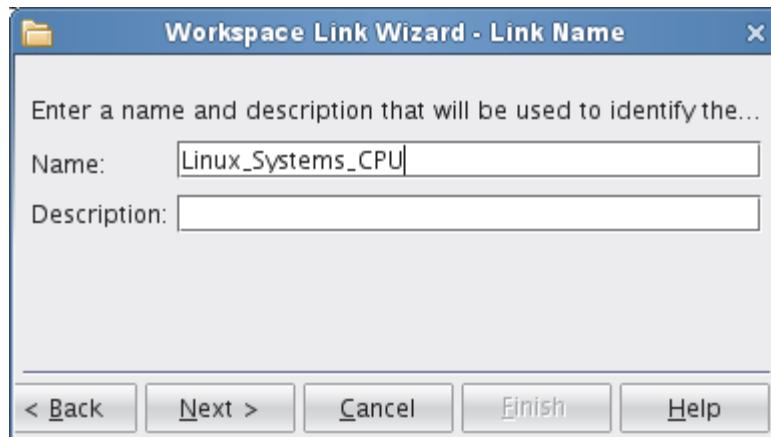
- From the Linux Systems Busy CPU view, right-click one of the bars and select Link To > Link Wizard.



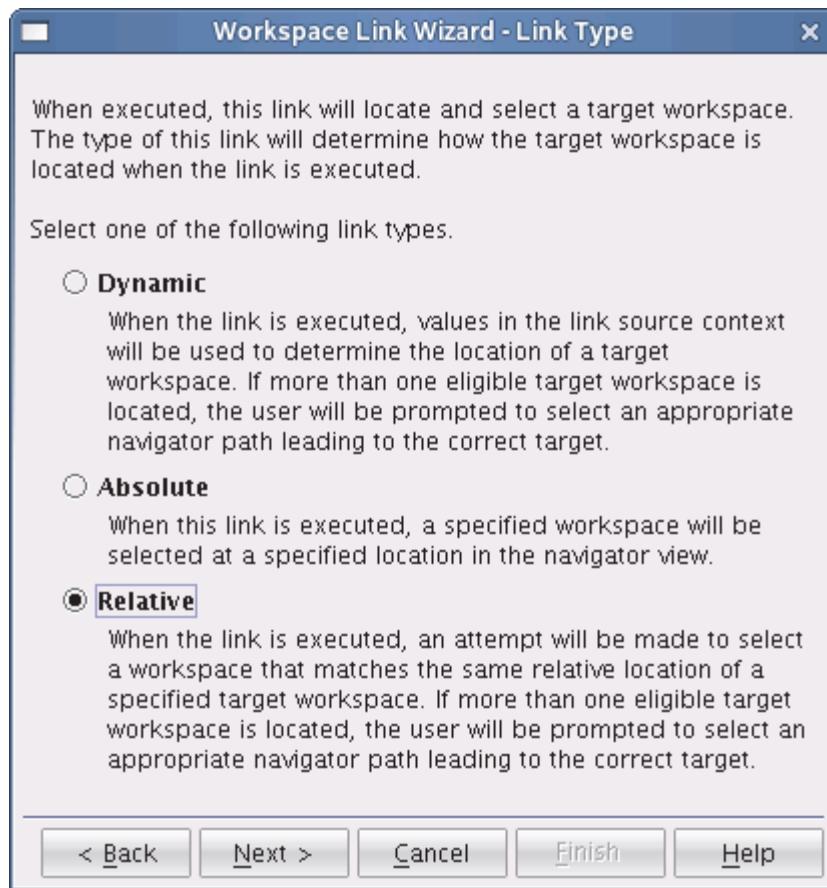
- Select Create a new link and click Next.



4. Enter the link name **Linux\_Systems\_CPU** and click **Next**.

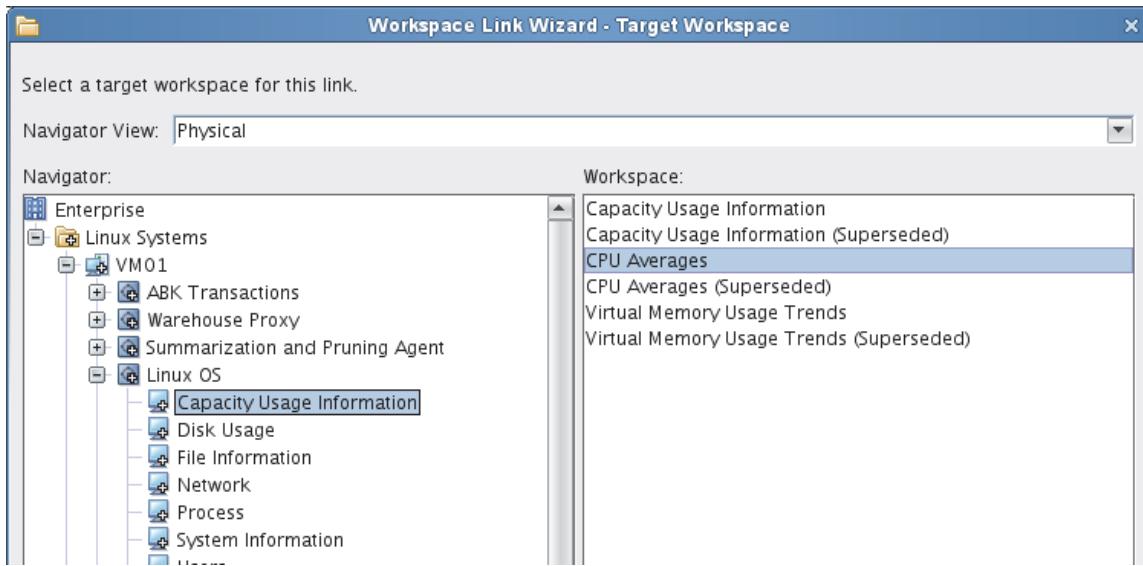


5. From the Link Type window, select **Relative** and click **Next**.

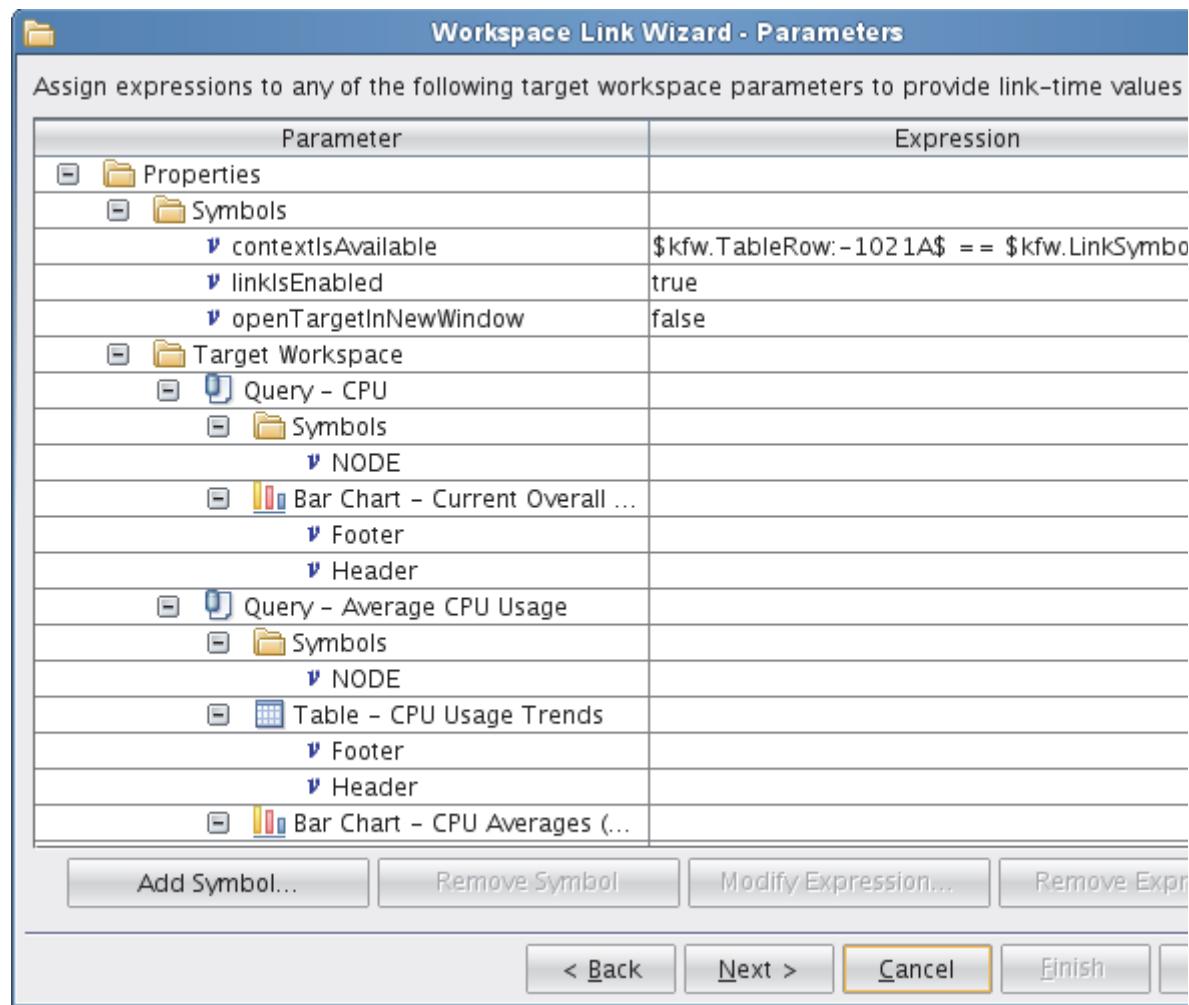


6. In the Workspace Link Wizard - Target Workspace window, expand the Navigator to select **Enterprise > Linux Systems > VM01 > Linux OS > Capacity Usage Information**.

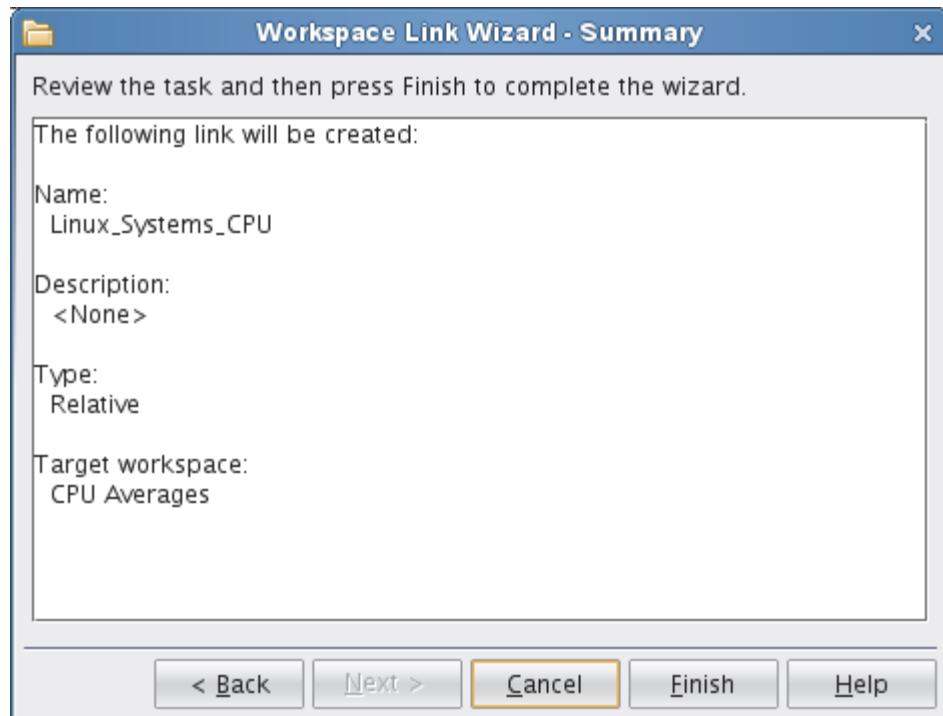
In the right pane, select **CPU Averages**. Click **Next**.



7. In the Workspace Link Wizard - Parameters window, click **Next**.



8. In the Workspace Link Wizard - Summary window, click **Finish**.

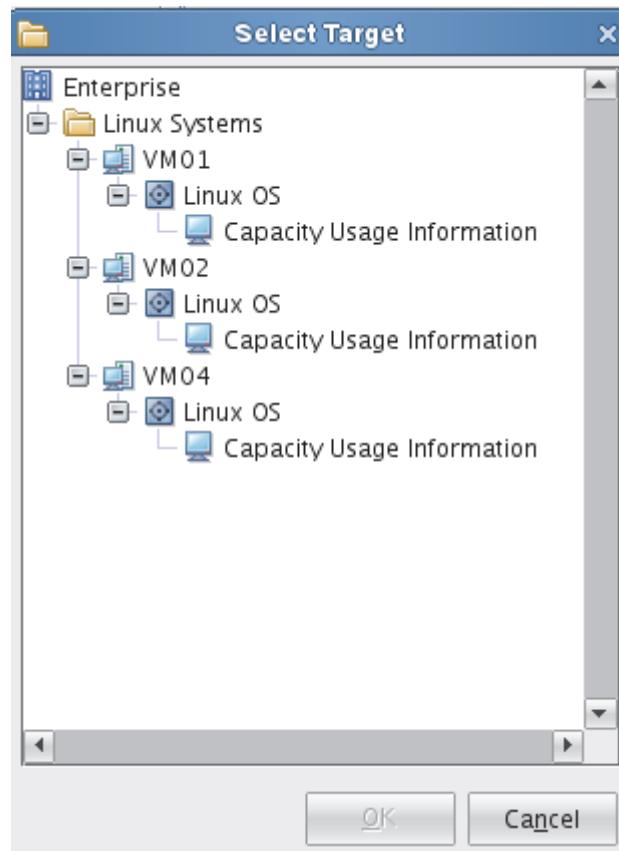


9. In the Linux Systems Busy CPU view, right-click a bar in the view. From the drop-down menu, select **Link To > Linux\_Systems\_CPU**.

A Select Target window opens.

You can see that the same target workspace is available from multiple managed systems. With relative links, you must select the workspace from a specific target managed system that you want to display.

10. Select **Capacity Usage Information** under **VM02**. Click **OK**.

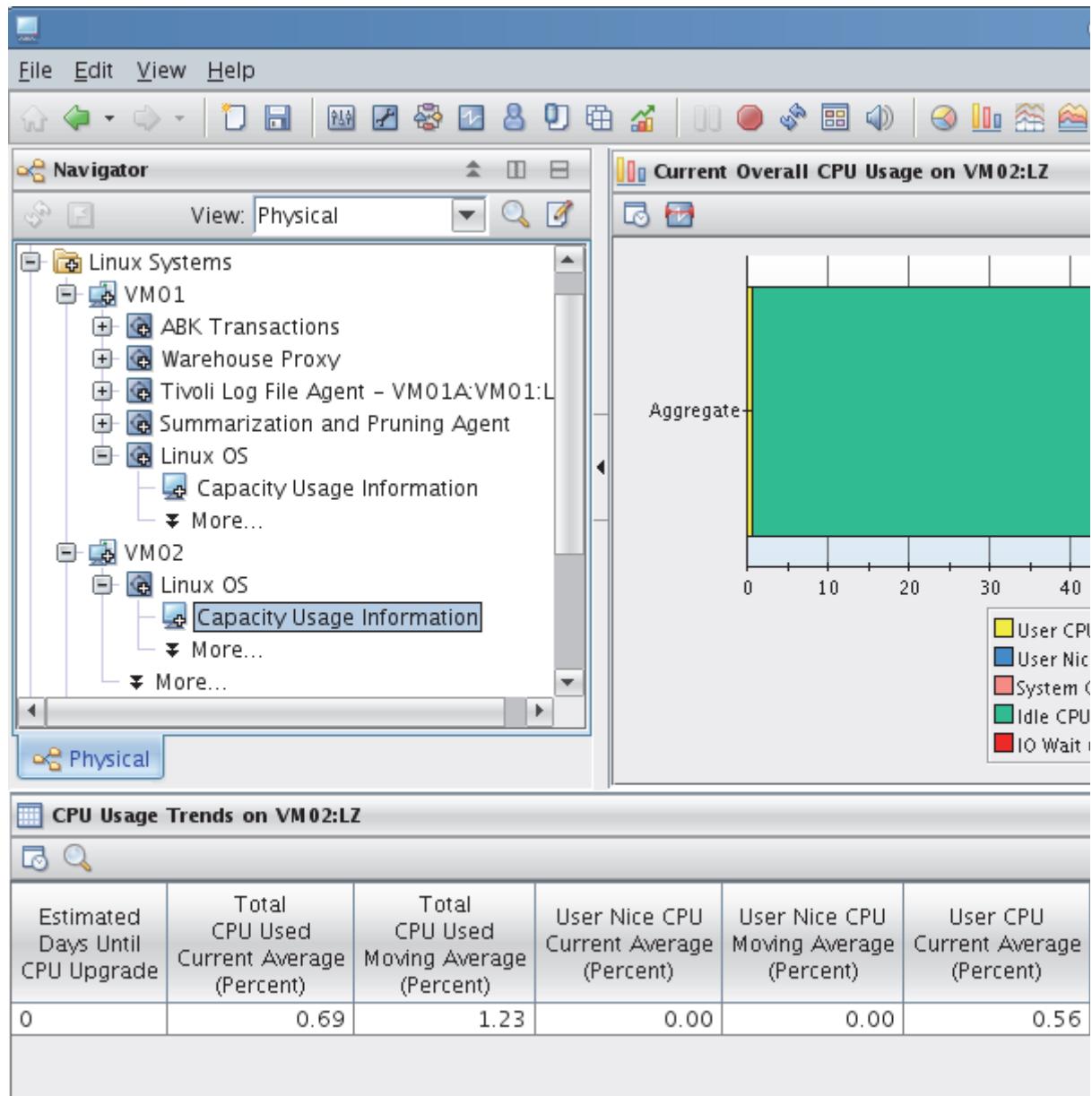


From the Navigator view, you can see that the Capacity Usage Information workspace from VM02 is now being displayed.

The screenshot shows the IBM Business Analytics interface. The top navigation bar includes File, Edit, View, Help, and various toolbar icons. The Navigator pane on the left displays a hierarchical tree structure under the 'Physical' view. It shows 'Enterprise', 'Linux Systems' (containing 'VM01', 'VM02' (with 'Linux OS' and 'Capacity Usage Information' selected), and 'VM04' (with 'Linux OS' and 'Process' listed). Below these are 'Windows Systems'. At the bottom of the Navigator pane are tabs for 'Physical' and 'AnyCorp'. The main content area features a chart titled 'Current Overall C' (partially visible) showing an aggregate value of approximately 100. Below the chart is a 'CPU Usage Trends' table with the following columns:

Estimated Days Until CPU Upgrade	Total CPU Used Current Average (Percent)	Total CPU Used Moving Average (Percent)	User Nice CPU Current Average (Percent)	User Nice CPU Moving Average (Percent)	User CPU Current Average (Percent)	

11. Optional: From what you learned about substitutions into headers and footers, modify the **Linux\_System\_CPU** relative link. Insert the node name into the header of the views as illustrated in the following example.





### 3 Monitoring your enterprise using advanced situation techniques exercises

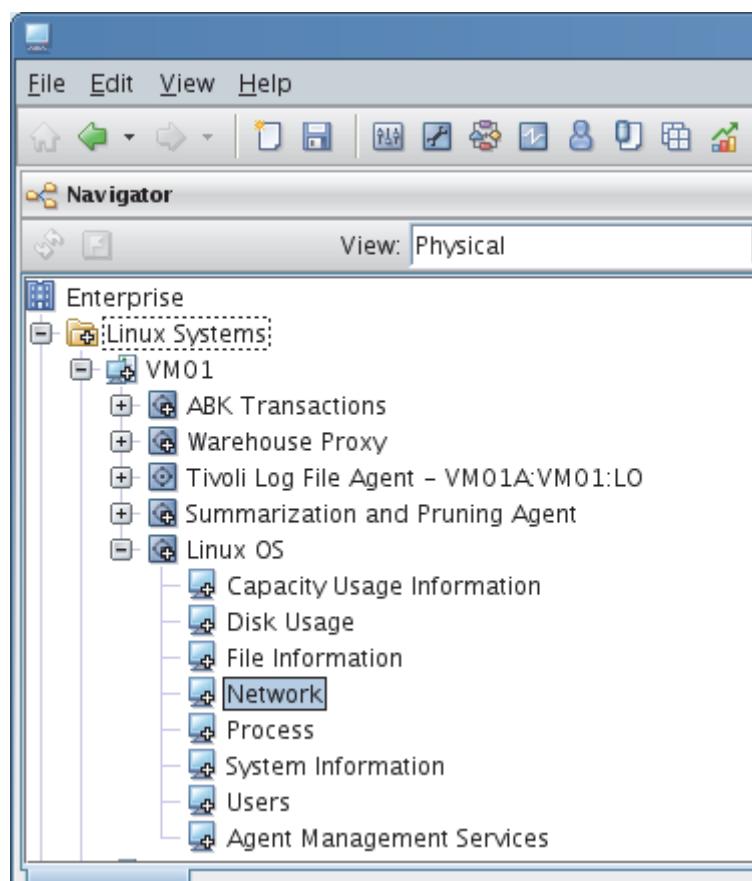
In the prerequisite class, you learned how to create simple situations. In this set of exercises, you learn how to use model situations, embedded situations, correlated situations, and dynamic thresholds.

# Exercise 1. Modeling situations

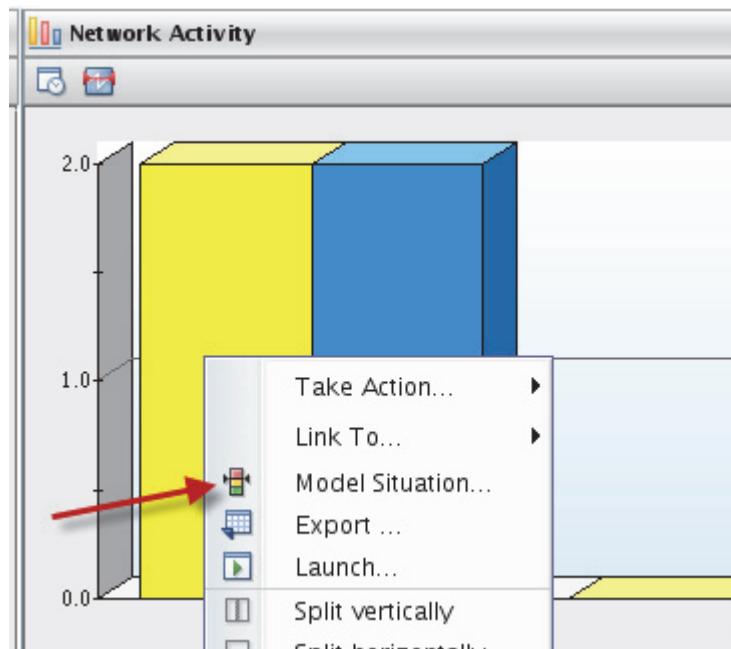
You can create situations that are based on actual values, rather than arbitrary conditions. You can create situations by using current values, historical values, or values that you modify by using statistical functions.

## Using historical monitored data

1. Open the VM01 Linux OS Network workspace.



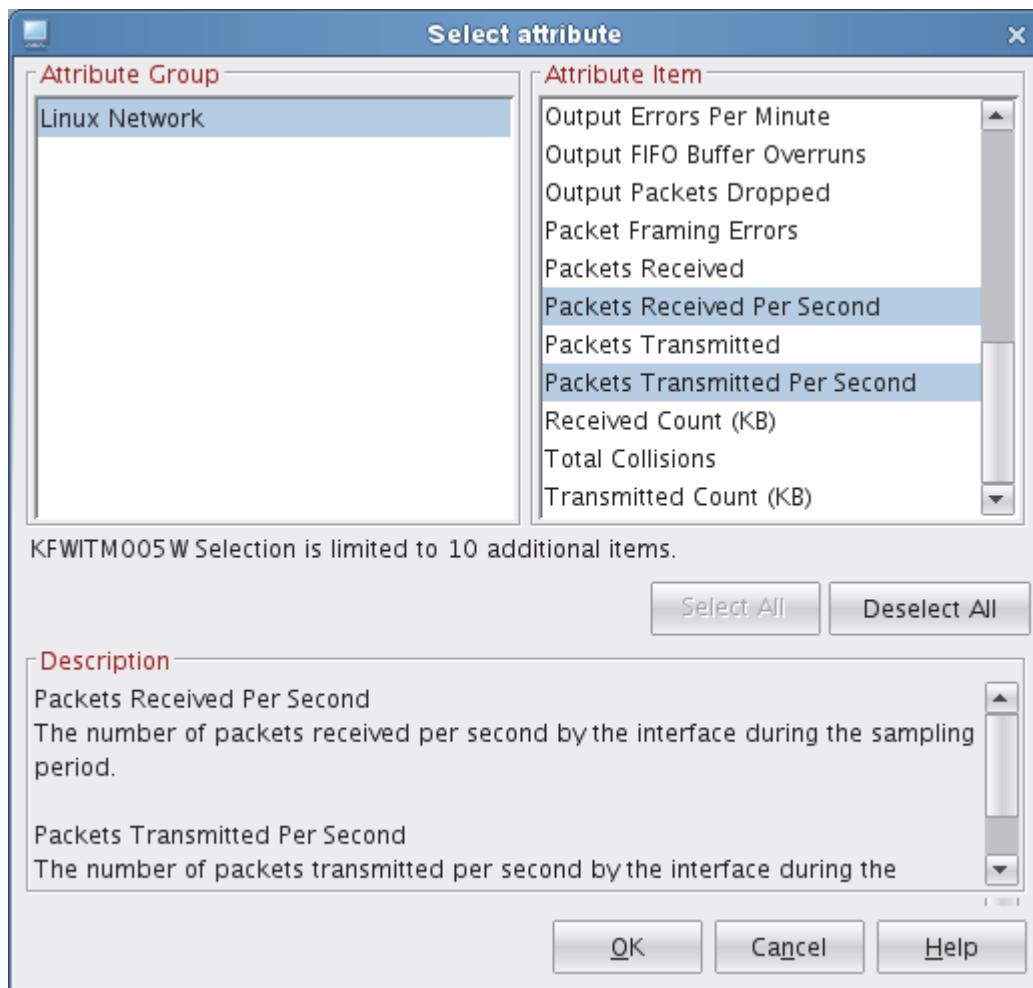
2. Locate the **Network Activity** bar chart view. Right-click one of the bars for the **Io** interface.  
Click **Model Situation**.



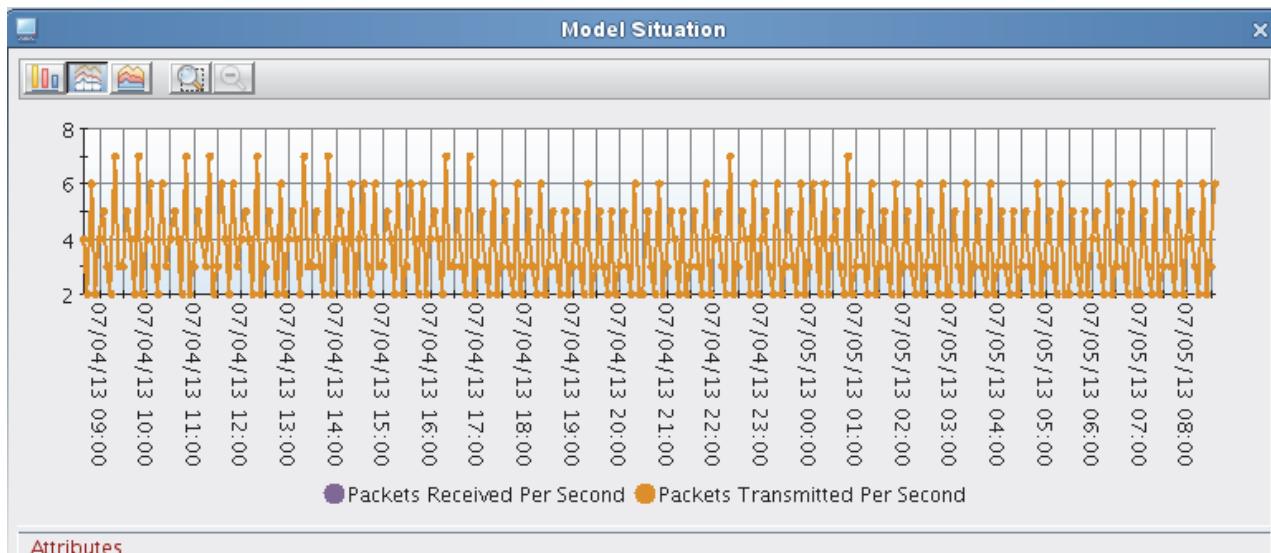
The Select attribute window opens.

## Exercise 1. Modeling situations

3. Scroll down to observe that the **Packets Received Per Second** and **Packets Transmitted Per Second** attributes are selected.

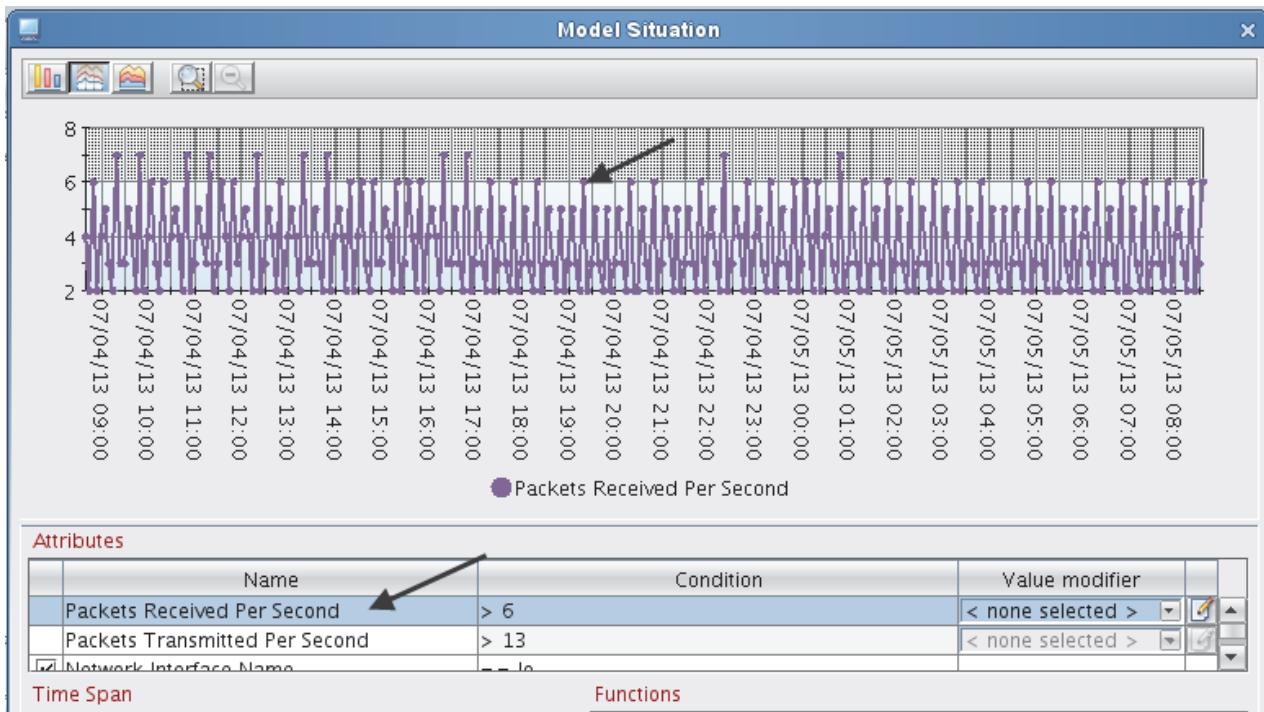


4. Click **OK** to open the Model Situation window.



Several tools are provided to help visualize the collected data.

- Click the **Packets Received Per Second** attribute. This action causes the area above the current value to be shaded. Click the edge of the shaded area and drag it up and down, observing that the condition value changes correspondingly.

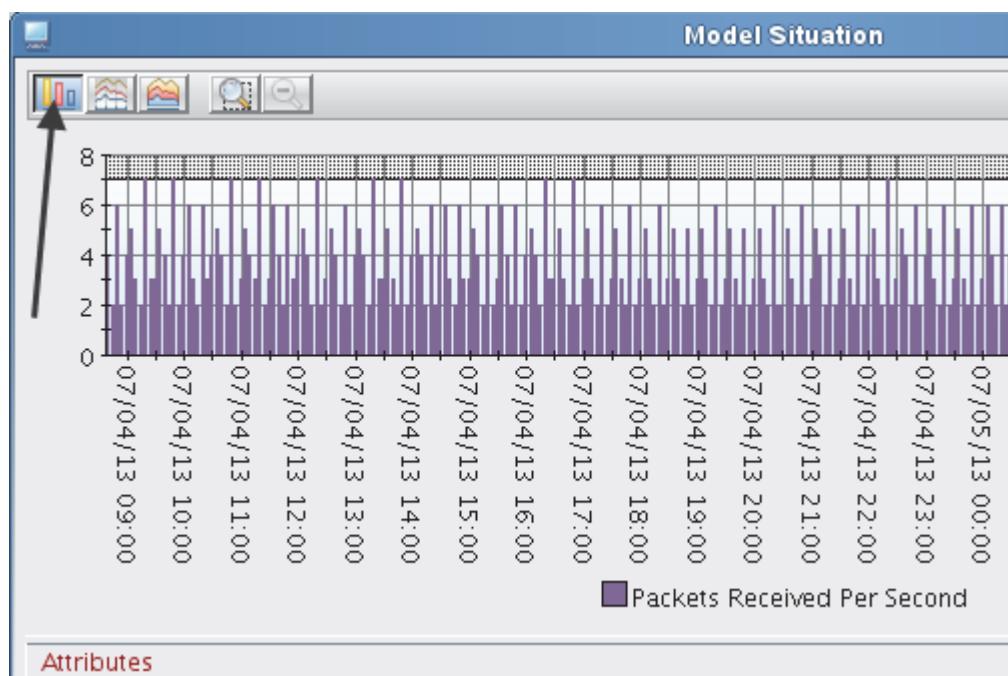


- Click one of the peaks in the plot line.

The shaded area moves, and the condition value is replaced by the value of the plot point.

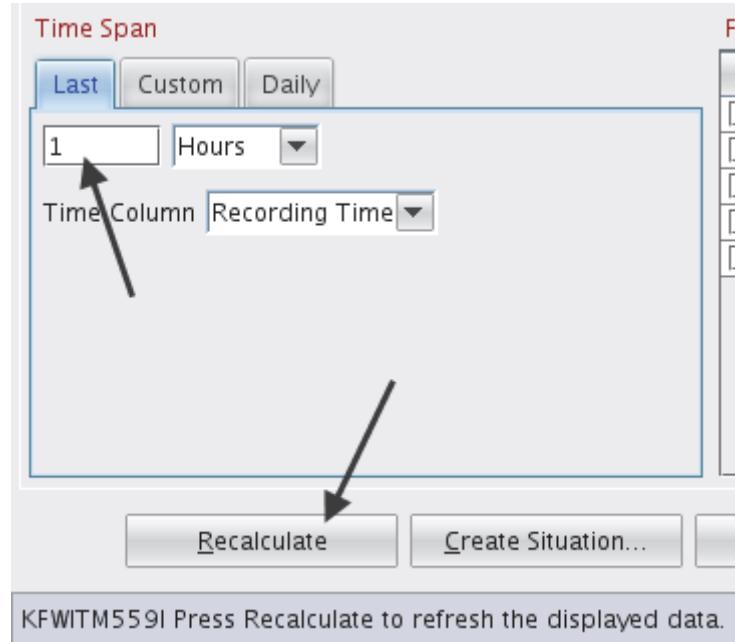
You can change the view type to help visualize the data.

7. Switch the view from a plot chart to a bar or area chart by clicking the icons in the upper left of the window.



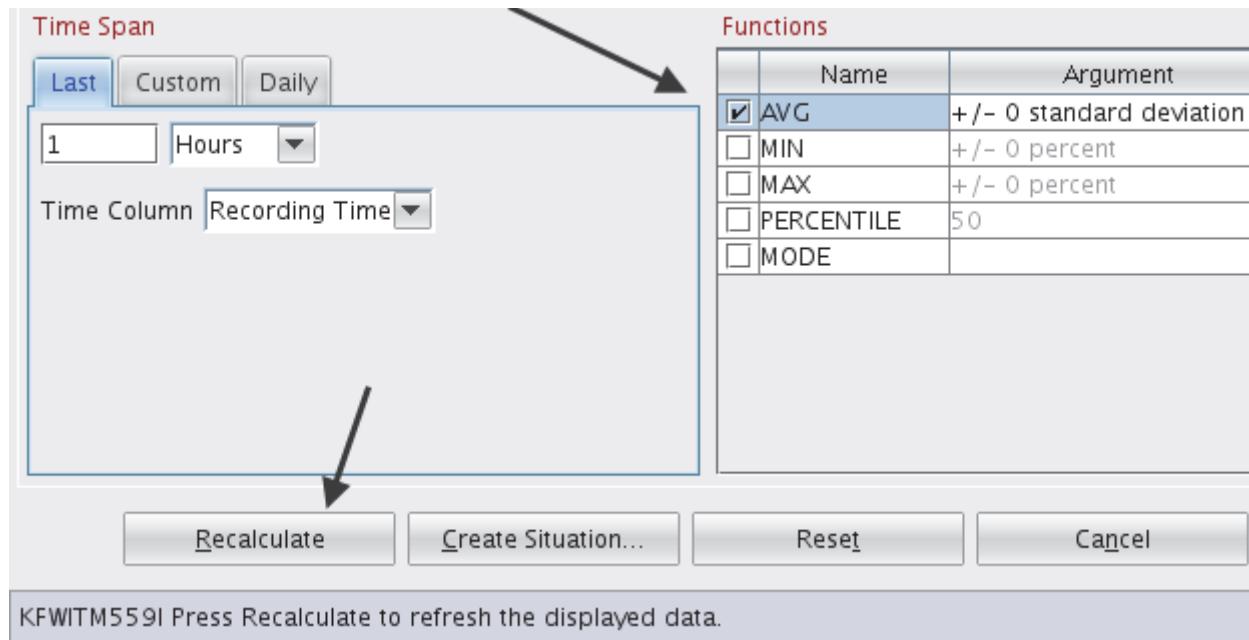
You can vary the amount of time that is displayed. If you are not collecting historical data for the attributes in the view, the **Time Span** area is not available.

8. Notice that the view shows the past 24 hours. Change the historical time range to the last **1 hour** and click **Recalculate**.



Use the **Functions** area to apply statistical functions to the condition and draw lines on the chart. These lines show you what the attribute value is in relation to ranges derived from statistical functions.

9. Plot the value average. Select the check box beside **Avg** and click **Recalculate**.



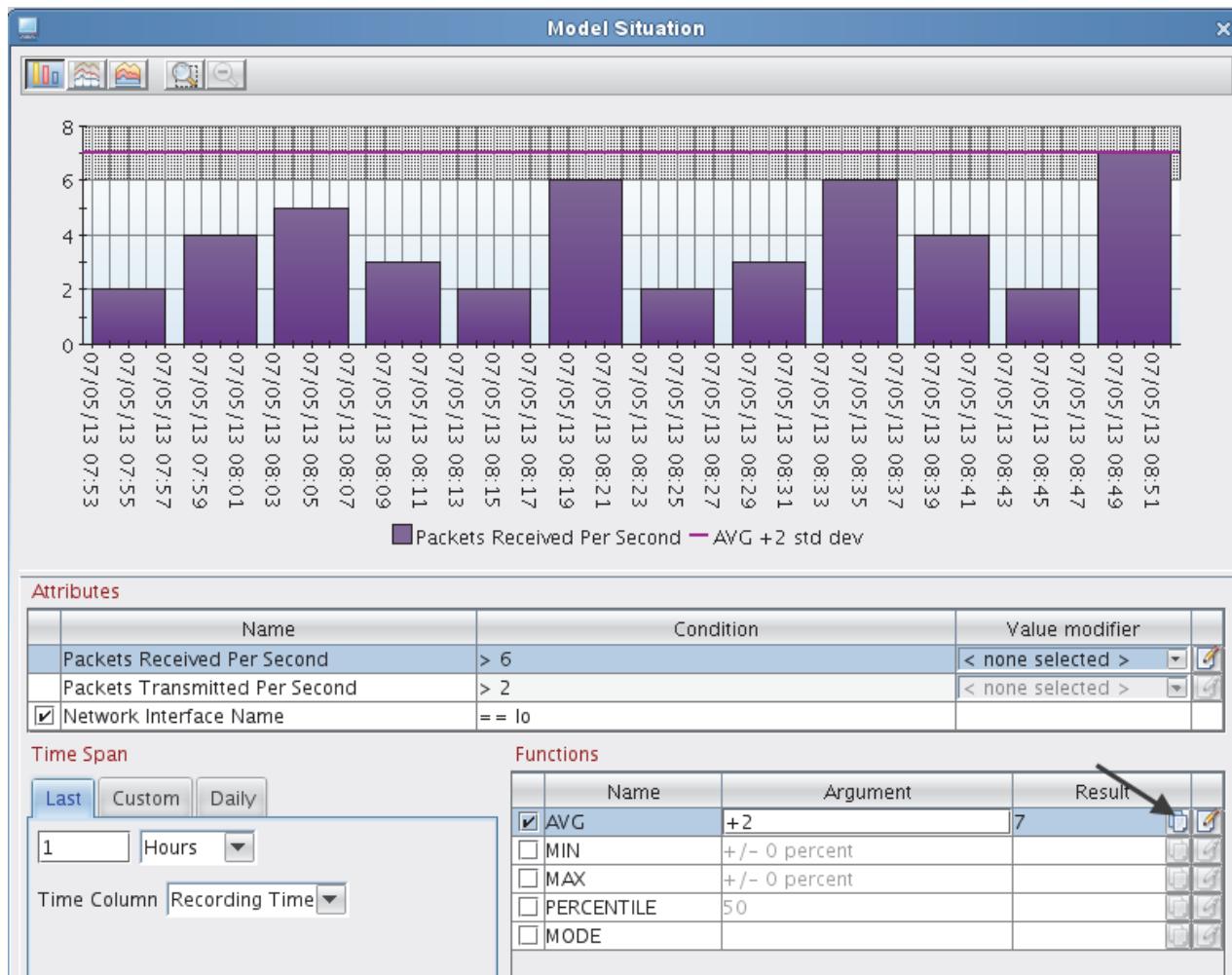
A line is drawn on the chart showing the average of the collected values. The function is added to the chart legend.

10. Click the **Argument** field for **Avg**, type **+2**, and press Enter. Click **Recalculate**.

The plot line is updated for two standard deviations above the average.

## Exercise 1. Modeling situations

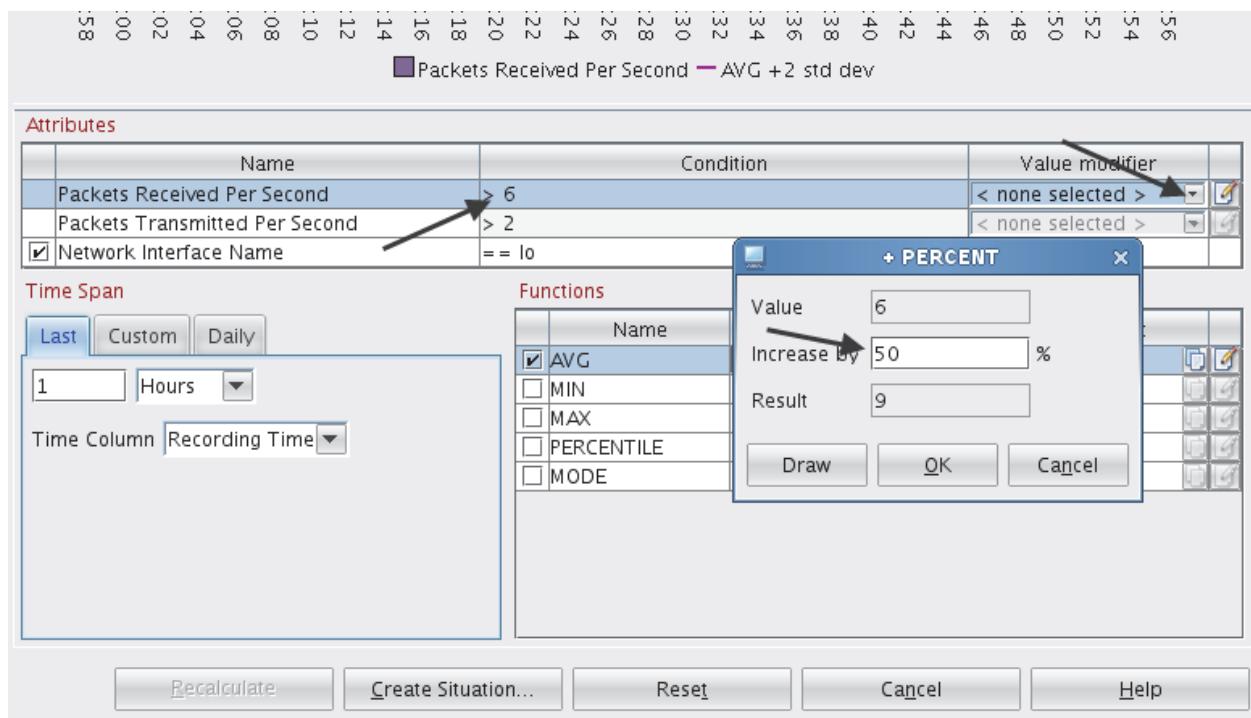
11. Assume that you are satisfied with the calculated value of the condition. Update the **Condition** field by clicking the **Copy Results** tool, which is to the right of the **Result** field.



This action copies the calculated value to the **Condition** field in the **Attributes** frame.

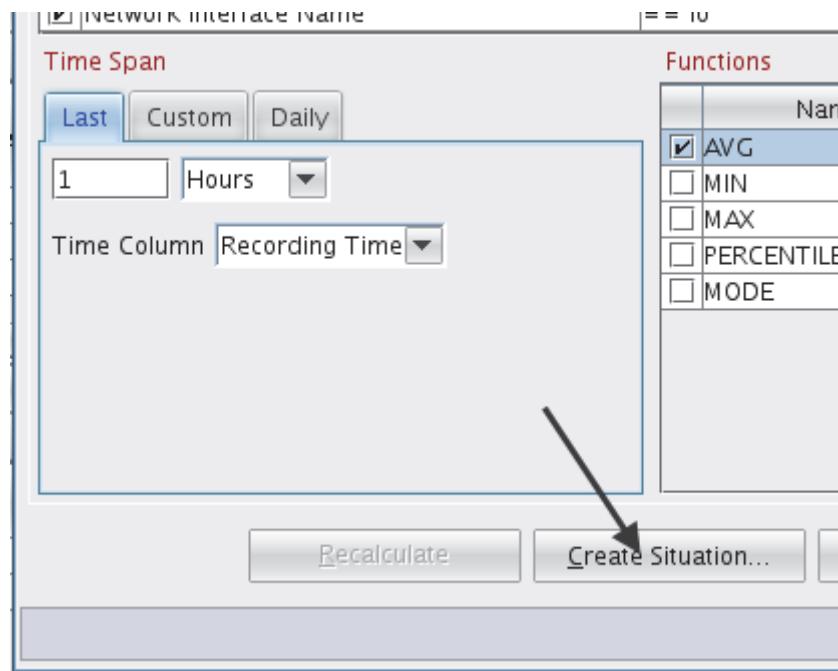
12. Experiment with the other functions to observe their effect on the condition. You can use all the functions to modify the condition, but you can apply only one at a time. You can further modify

the value of the condition by clicking the down arrow in the **Value modifier** field and entering a plus or minus percentage.



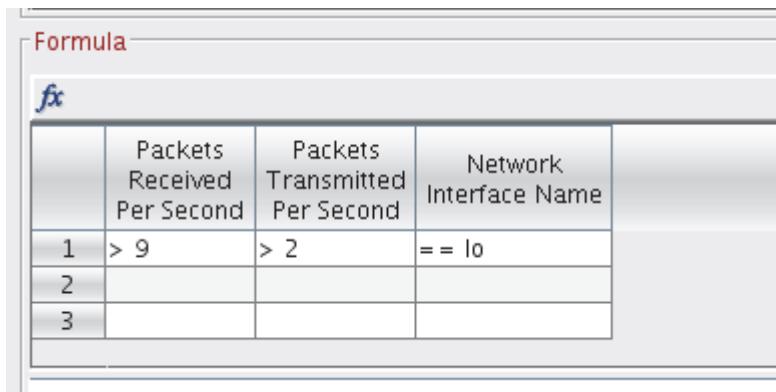
The **Result** field is updated interactively when you enter values. Clicking **OK** copies the **Result** field to the **Condition** field.

- When you are satisfied with the value of the conditions, click **Create Situation**.



- Name the situation **Linux Network by Model** and click **OK**.

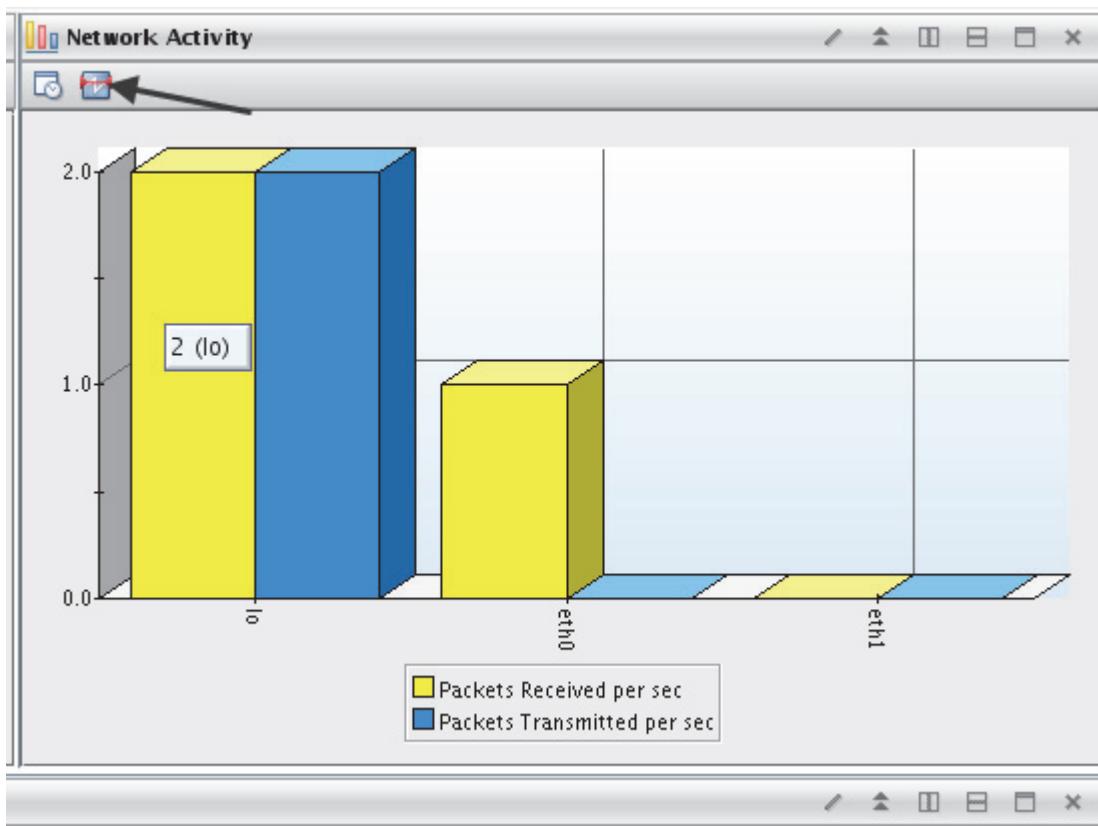
The Situation editor opens with the attributes and condition values that you modified.



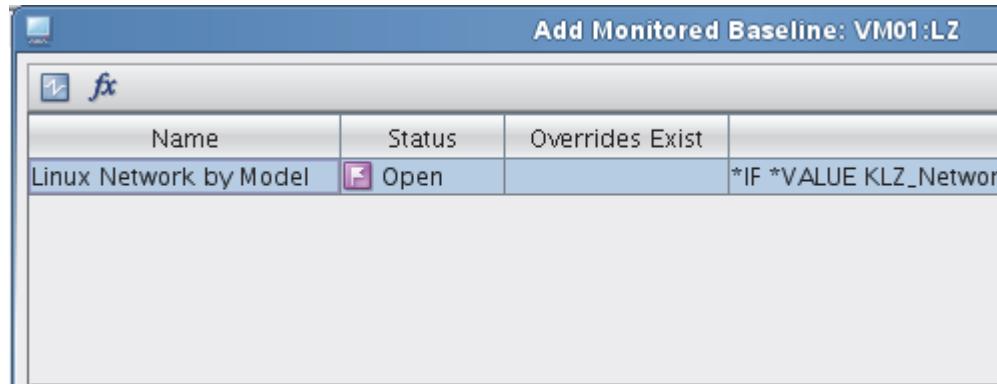
15. Leave the Network Interface Name equal to **lo**. The Situation modeler passed those values from the view. Set the sampling interval to **30 seconds** and change the State to **Minor**. Click **OK** to start the situation and close the Situation editor.

You can use the Baseline tool to draw the situation conditions value on the view.

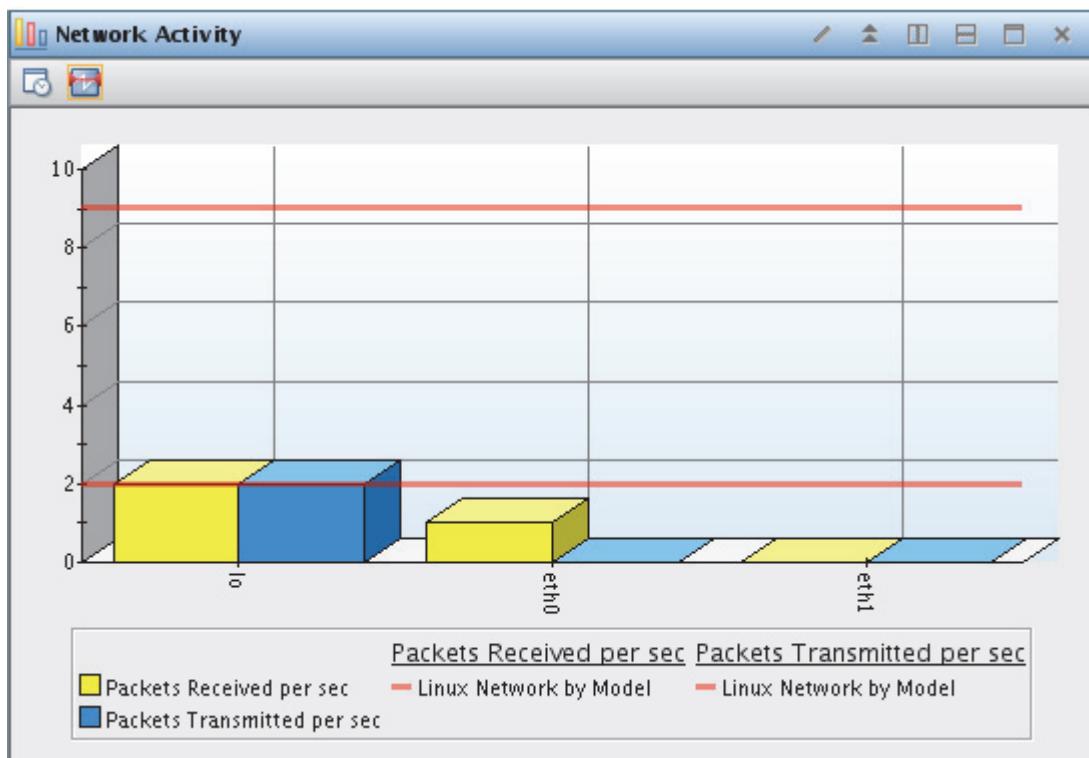
16. Click the **Add Monitored Baseline** icon.



17. Click the modeled situation name to select it and click **OK**.



The condition values are shown in the view as solid lines.



18. Save the updated workspace with a new name, **My\_Linux\_Network**.

## Exercise 2. Using embedded situations

To monitor and manage your enterprise, you can create situations to detect problems with your enterprise resources. You create most situations to test whether a single attribute setting matches a value that causes the situation to become true. If the situation is true, it raises a situation event. Sometimes, situations are more complex. This exercise covers the creation of embedded and correlated situations.

During this exercise, you generate a situation that triggers a situation event when a logon fails during nonworking hours. You modify an existing situation called **Invalid\_ssh\_attempt**. In that situation, you embed a situation that provides a condition filter for all hours outside the time frame between 8 a.m. and 5 p.m.

1. Switch to the VM02 system and open a terminal session. Issue the system command and then press Enter several times. This action provides a bad password.

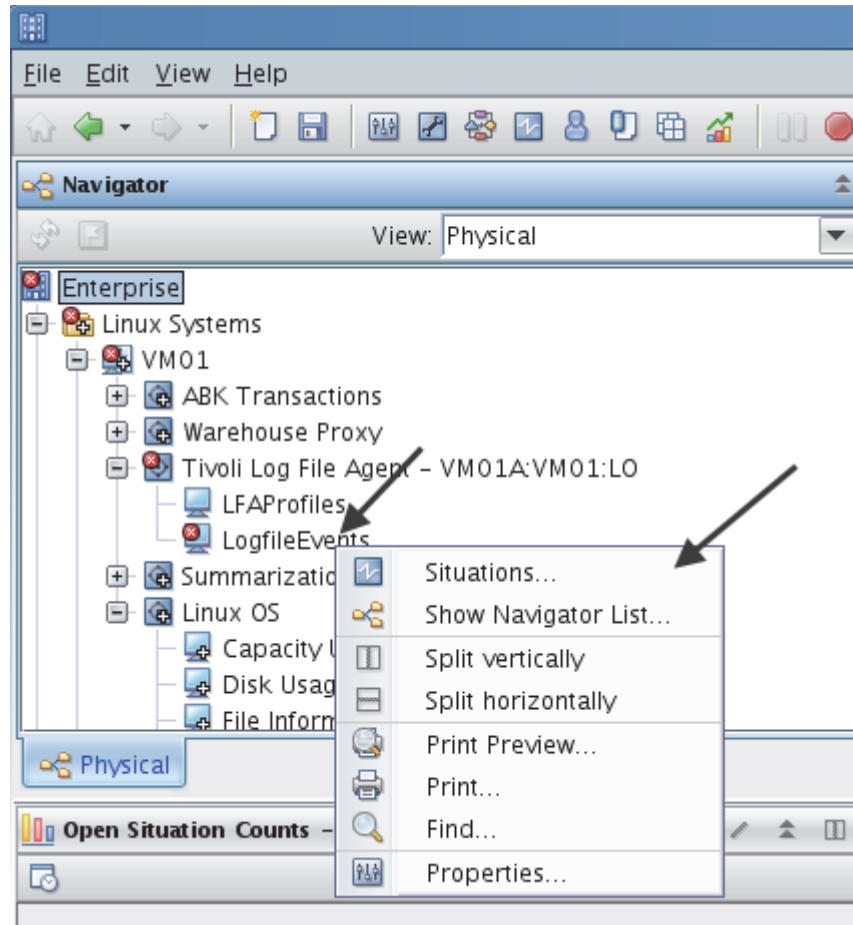
```
ssh vm01
```

The screenshot shows a terminal window with the title 'Terminal'. The menu bar includes 'File', 'Edit', 'View', 'Terminal', and 'Help'. The terminal prompt is 'VM02:~ #'. The user enters 'ssh vm01' and is prompted for a password three times. Each password attempt fails with the message 'Permission denied, please try again.'. After three failed attempts, the terminal displays 'Received disconnect from 192.168.100.101: 2: Too many authentication failures'. The user then types 'r root' and logs in successfully, indicated by the prompt 'VM02:~ #'. The background of the terminal window is light blue, and the text is black.

```
VM02:~ # ssh vm01
Password:
Password:
Password:
root@vm01's password:
Permission denied, please try again.
root@vm01's password:
Permission denied, please try again.
root@vm01's password:
Received disconnect from 192.168.100.101: 2: Too many authentication failures
r root
VM02:~ #
```

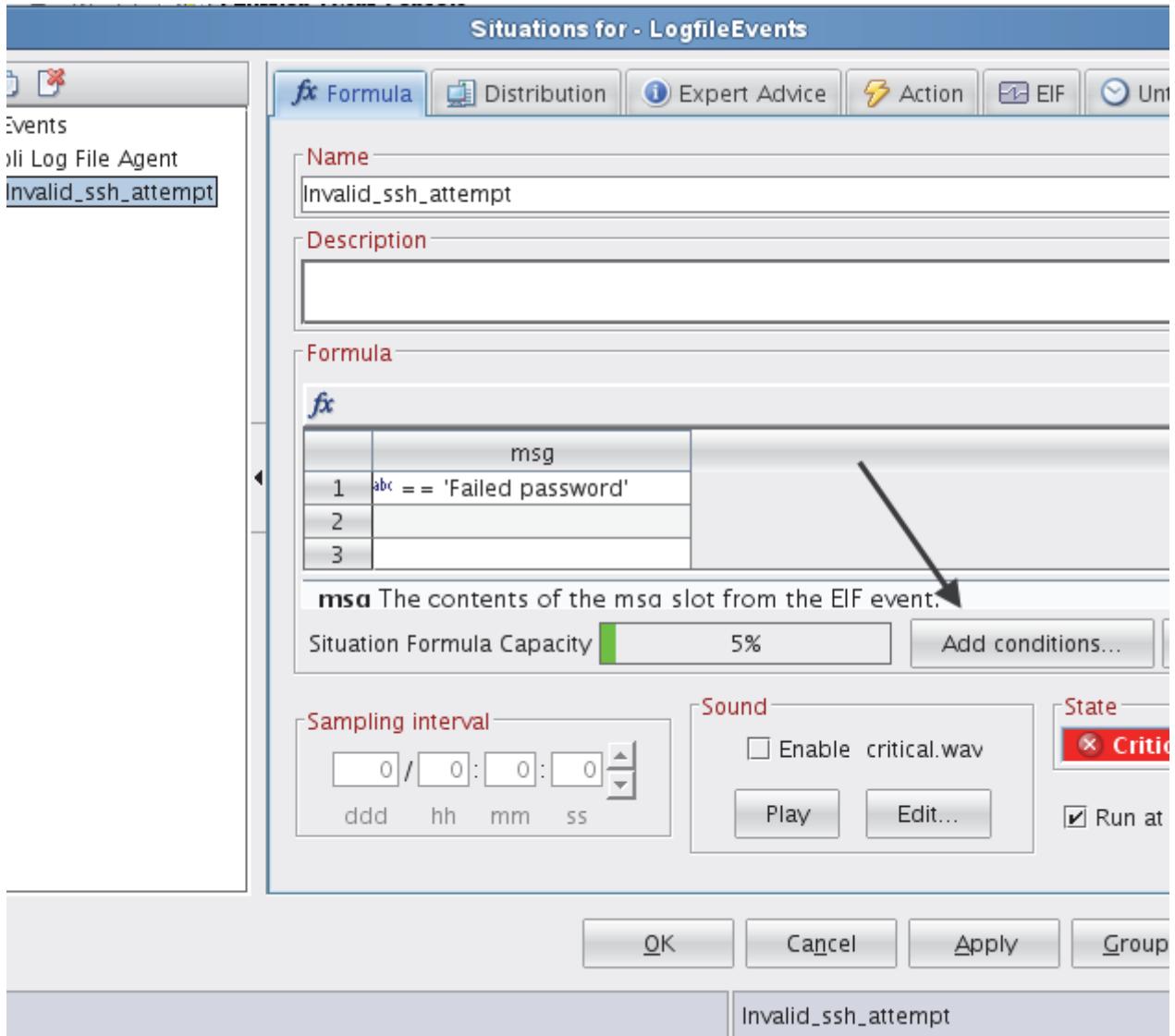
2. On VM01, locate the situation in the Situation Event Console view by selecting the Enterprise Navigator item in the Tivoli Enterprise Portal.

3. On VM01, open the Situation editor from the **LogfileEvents** item for **VM01**.

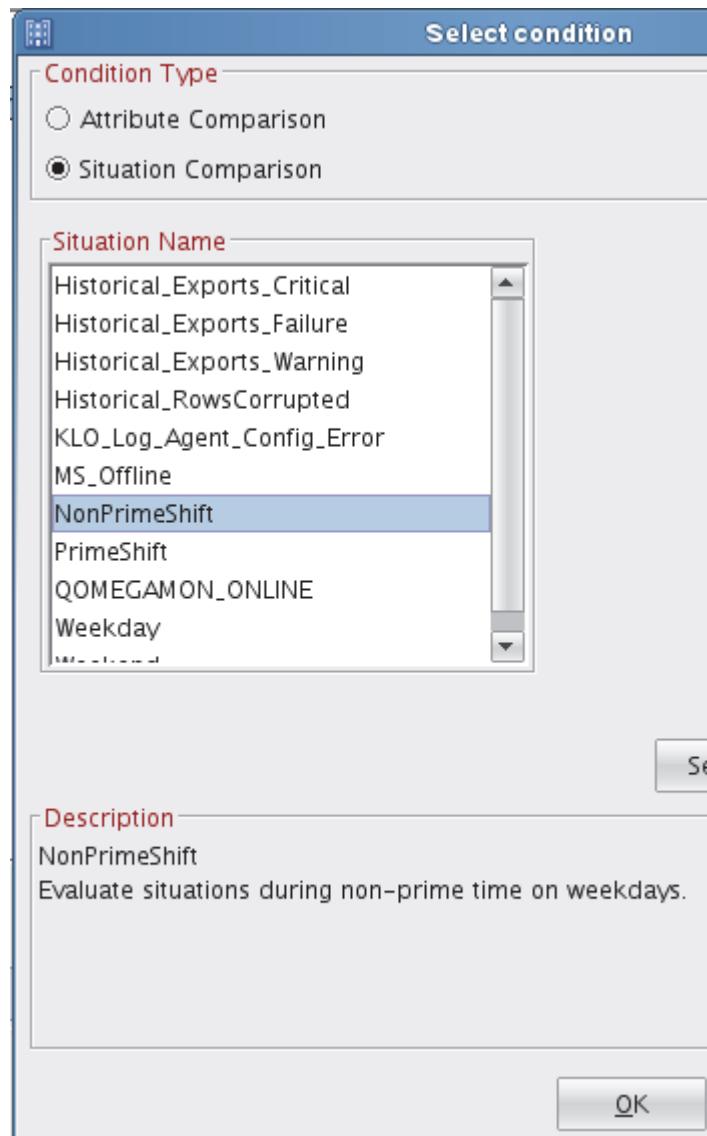


4. Access the **Invalid\_ssh\_attempt** situation.

5. Click **Add conditions**.



6. Set the Condition Type to Situation Comparison and select the NonPrimeShift situation.  
Click OK.



7. In the formula, click the empty field below the **NonPrimeShift** column header.

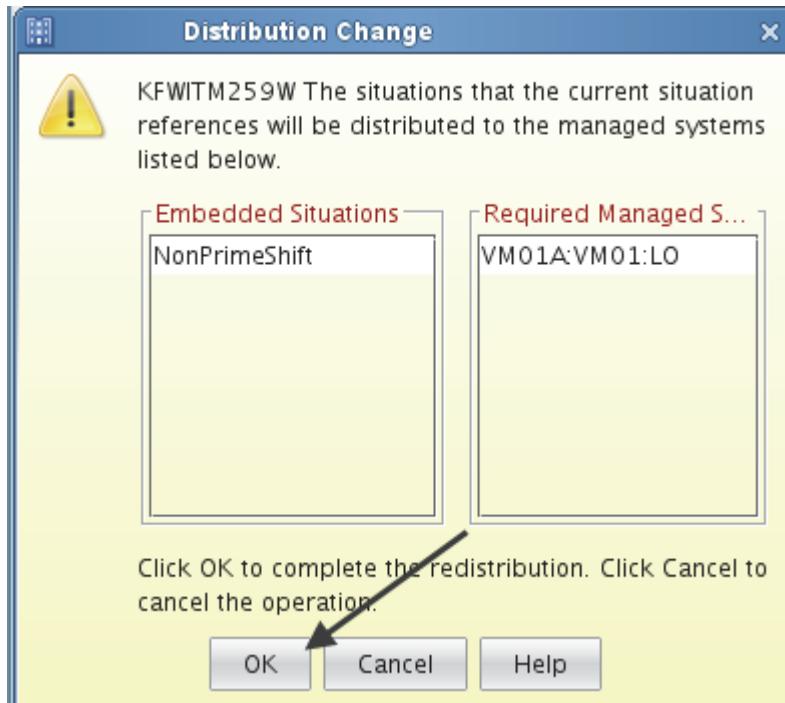
The screenshot shows a 'Formula' editor. It displays a table with two columns: 'msg' and 'NonPrimeShift'. The 'msg' column contains the formula 'abc == 'Failed password''. The 'NonPrimeShift' column has an empty cell for row 1, which is indicated by an arrow. Below the table, there is a description of the 'msa' slot and a progress bar for 'Situation Formula Capacity' at 5%.

This action enters a True statement.

Formula		
	msg	NonPrimeShift
1	abc == 'Failed password'	True
2		
3		

msa The contents of the msa slot from the FIF event

8. Click **OK** to save your situation.
9. Click **OK** to approve the distribution of the embedded situation.



10. Switch to the VM02 system and open a terminal session. Issue the system command and then press Enter several times. This action provides a bad password.

```
ssh vm01
```

This time, you do not see a situation event for your invalid logon attempts if you are working during prime shift hours (based on the local time of VM01).

11. Verify that the situation works correctly by modifying your situation formula to embed the **PrimeShift** situation instead of the **NonPrimeShift** situation.
  - a. Access the **Invalid\_ssh\_attempt** situation.
  - b. Right-click the **NonPrimeShift** column header and select **Delete**.

- c. Click **Add conditions**.
- d. Select **Situation Comparison** and the **PrimeShift** situation name. Click the empty field below the column header to set the condition to **True**.

Formula		
	msg	PrimeShift
1	abc ... 'Failed password'	= = True
2		
3		

12. Activate your situation and attempt to **ssh** from VM02 to VM01 with an incorrect password several times.

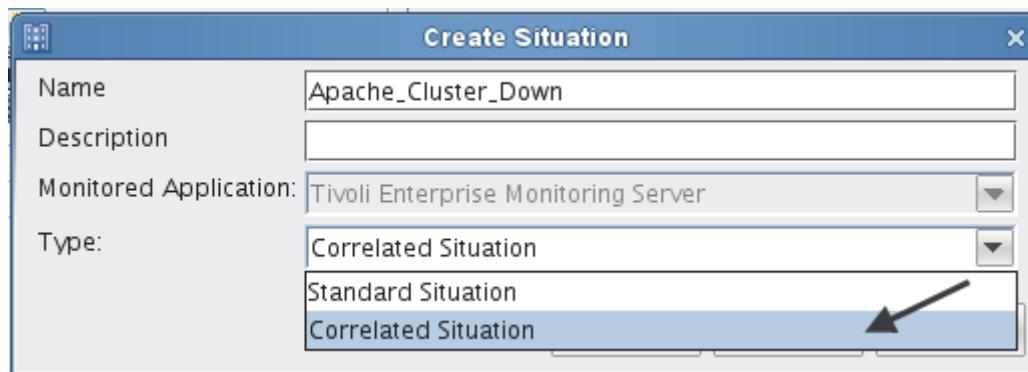
If the current time on your system is during working hours, you see a situation event.

Situation Event Console						
	Severity	Status	Owner	Name	Display Item	Source
	Critical	Open		Invalid_ssh_attempt		VM01A:VM01:LO

## Exercise 3. Creating correlated situations across multiple managed systems

Consider the case where having two situations on different Tivoli Enterprise Monitoring agents can be used to create an additional situation. The web server monitoring situation **Apache\_Down\_Linux** triggers a Warning situation event. If the website is not available on any server, you see a Critical situation event. The most critical scenario is when both web servers are down and none of the websites are accessible.

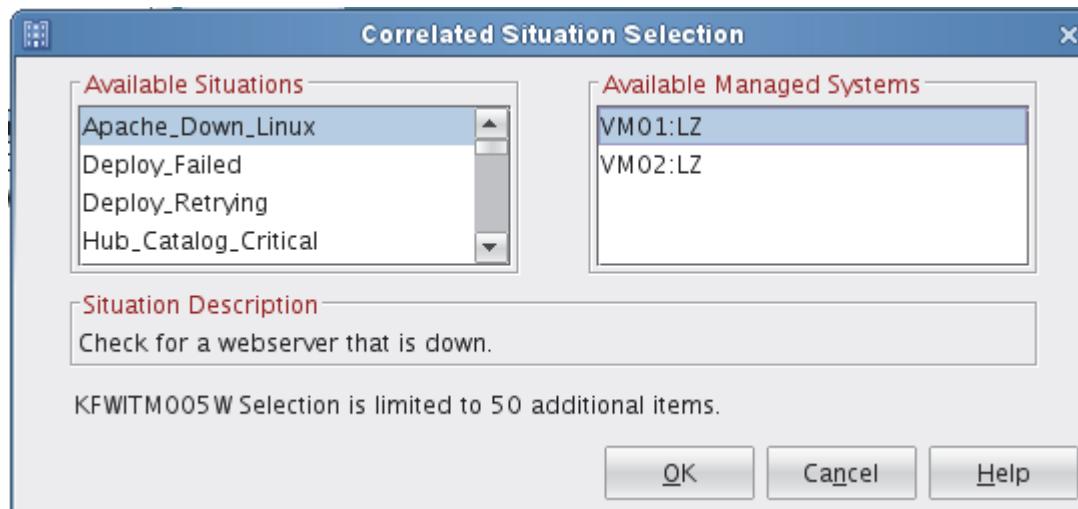
1. Access the Situation editor from the Enterprise Navigator item.
2. From the toolbar, click **Create new Situation**.
3. Enter the situation name **Apache\_Cluster\_Down** and select **Correlated Situation**.



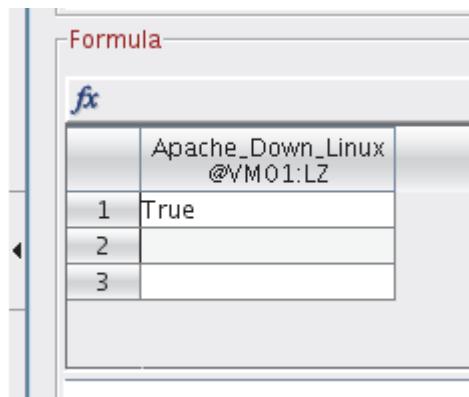
The monitored application becomes All Managed Systems and cannot be changed.

4. Confirm the selection.
5. Include situations from different managed systems in your formula. Select the **Apache\_Down\_Linux** situation as your first situation.

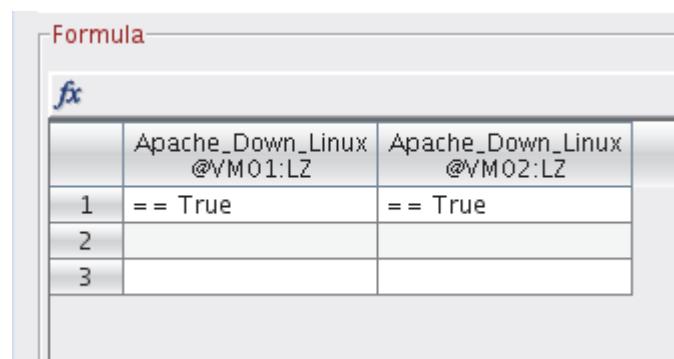
6. Select the Linux system **VM01:LZ** as the managed system, where the situation must be true to be considered for this correlated situation. Click **OK**.



7. In the Formula pane, click the empty field under the situation to enable it as part of the formula. The field displays **True**. Next, the situation symbol changes to **==True**.

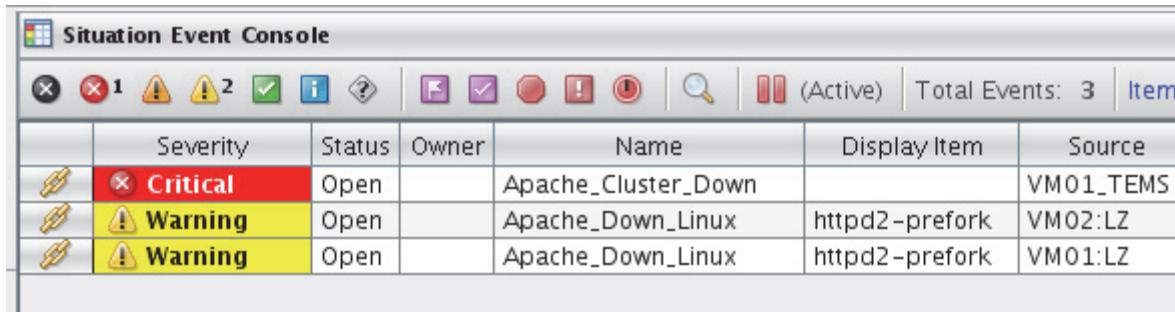


8. Click **Add conditions** to add the second situation.
9. Select the **Apache\_Down\_Linux** situation and select **VM02:LZ** as the managed system. Set the formula field to **True**.



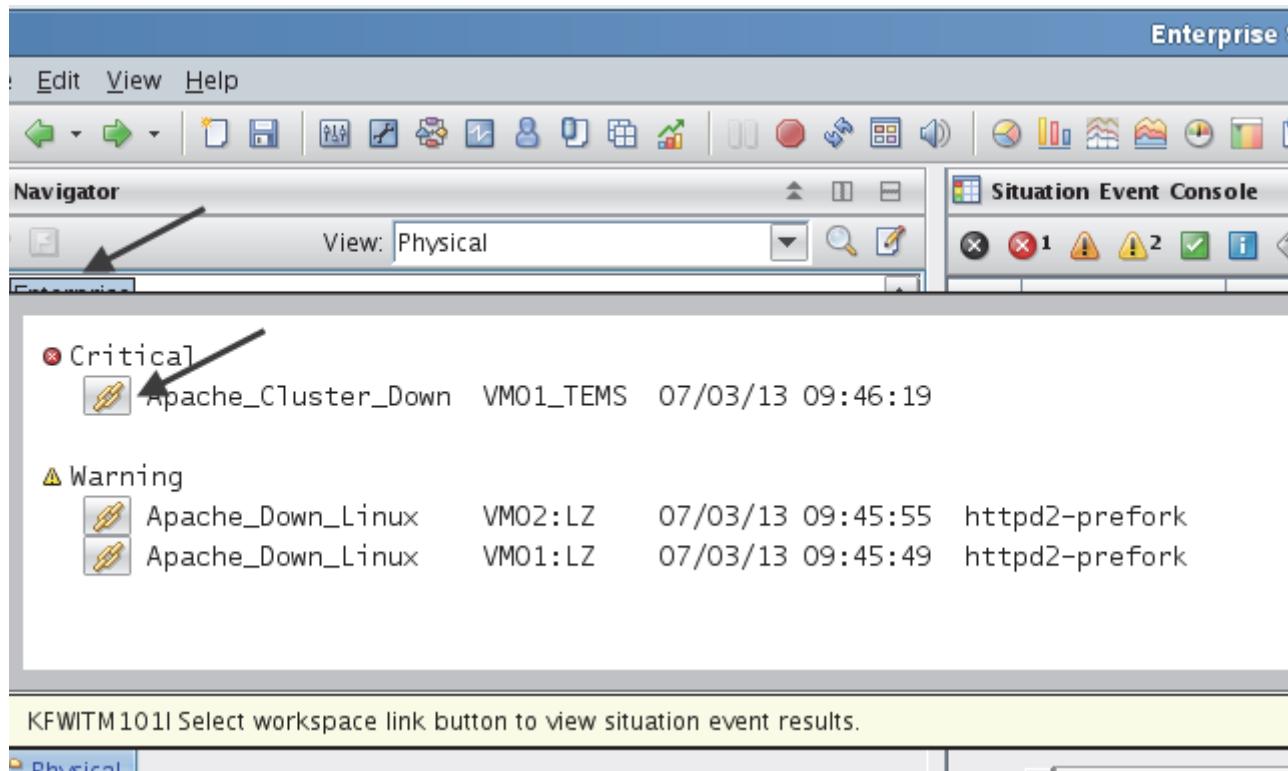
10. Set your sampling interval to **30** seconds and leave the state **Critical**.
11. Stop the Apache web server on both VM01 and VM02 by using the following commands from a terminal window:

```
cd /etc/init.d
./apache2 stop
```
12. Click **OK** to close the Situation editor and watch for the situation event to be displayed in the Situation Event Console view.



	Severity	Status	Owner	Name	Display Item	Source
	Critical	Open		Apache_Cluster_Down		VM01 TEMS
	Warning	Open		Apache_Down_Linux	httpd2-prefork	VM02 LZ
	Warning	Open		Apache_Down_Linux	httpd2-prefork	VM01 LZ

13. Position your cursor over the Enterprise Navigator item to determine the basis for the situation event. Access the Situation Event Results workspace by selecting the link icon for **Apache\_Cluster\_Down**.



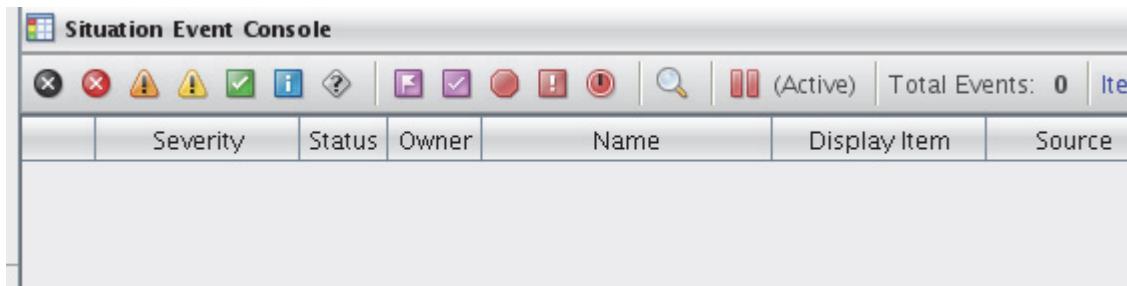
The screenshot shows the IBM Tivoli Monitoring 6.3 Advanced Administration interface. The top navigation bar has tabs for Edit, View, and Help. Below the navigation bar is a toolbar with various icons. On the left, there is a Navigator pane with a tree view and a "View: Physical" dropdown. A red arrow points to the "Physical" view button. In the center, there is a large workspace area. At the top of this workspace, there is a header for "Situation Event Console" with a list of events. Below this, the workspace displays two sections: "Critical" and "Warning". The "Critical" section contains one event: "Apache\_Cluster\_Down" from "VM01 TEMS" at "07/03/13 09:46:19". The "Warning" section contains two events: "Apache\_Down\_Linux" from "VM02 LZ" at "07/03/13 09:45:55" and "Apache\_Down\_Linux" from "VM01 LZ" at "07/03/13 09:45:49", both associated with "httpd2-prefork". At the bottom of the workspace, a message says "KFWITM101I Select workspace link button to view situation event results." A green arrow points to the "Physical" view button in the Navigator.

14. Click the **Back** button to close the Situation Event Results workspace.

15. Use the following commands to restart the Apache web server on VM01 and VM02. Observe the results in the Situation Event Console view.

```
VM01:~ # cd /etc/init.d  
VM01:/etc/init.d # ./apache2 start
```

```
[Unit: httpd.service           main process (pid: 1111)  
VM01:/etc/init.d # ./apache2 start  
Starting httpd2 (prefork)  
VM01:/etc/init.d # ]
```



## Exercise 4. Using dynamic thresholds

You can distribute a situation to multiple managed systems, but you might want different monitoring settings for one or more targets. You can apply dynamic thresholds to selected managed systems and override the situation settings.

In this exercise, you learn how to create situation overrides. You apply situation overrides to situations when they are distributed. You set the overrides individually for each managed system or managed system group.

**Exercise 4. Using dynamic thresholds**

This exercise works with the Monitoring Agent for Linux OS running on VM01 and VM02. The situation **Linux\_Low\_Space\_Available** is distributed to the default **LINUX\_SYSTEM** group.

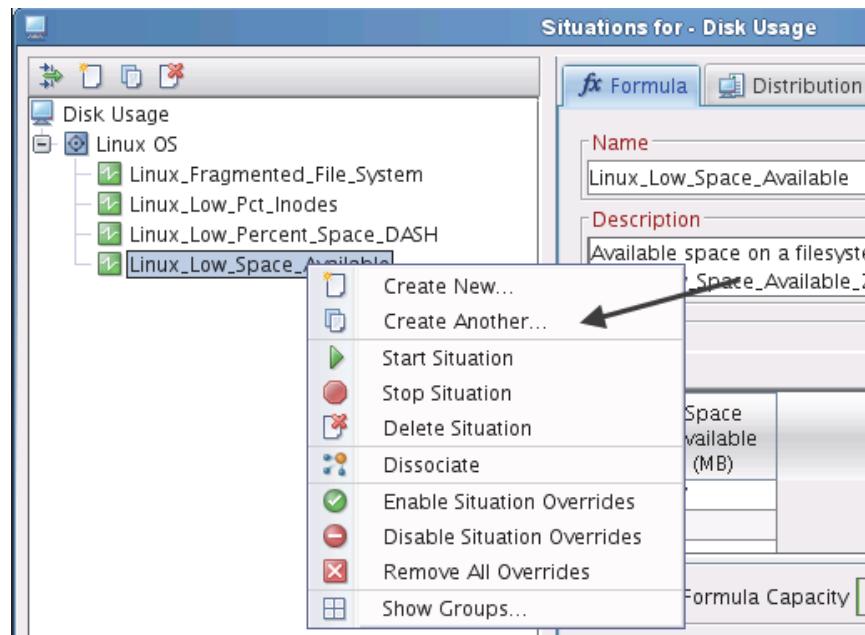
1. Open the Disk Usage workspace on VM01. Notice the values for the Disk Name and Disk Free (MB) in the Disk Usage view.

The screenshot shows the IBM Tivoli Monitoring 6.3 Advanced Administration interface. The Navigator pane on the left displays a tree structure of monitored systems: Enterprise, Linux Systems, VM01, and various agent components like ABK Transactions, Warehouse Proxy, and Tivoli Log File Agent. Under VM01, the Linux OS node is expanded, showing Capacity Usage Information, Disk Usage, File Information, and Network. The Disk Usage item is selected and highlighted with a red box. The Disk Usage view on the right lists disk usage details for the root mount point (/). The table has columns: Mount Point, Disk Name, Size (MB), Disk Used (MB), Disk Free (MB), and Total Inodes. The 'Disk Free (MB)' column for the root entry is also highlighted with a red box. The data in the table is as follows:

Mount Point	Disk Name	Size (MB)	Disk Used (MB)	Disk Free (MB)	Total Inodes
/	/dev/sda2	38673	29592	7510	249529

2. Start the Situation editor from the Linux OS Agent **Disk Usage** Navigator item on VM01.

- Right-click the situation **Linux\_Low\_Space\_Available** and select **Create Another**. Append your initials to the situation name to give it a unique name.



- Set the sampling interval to **30 seconds**. Click **Add conditions**. Select **Disk Name**. Click **OK**.
- Under the Disk Name attribute, enter **/dev/sda2**.
- Under the Space available attribute, enter the value **25000**.
- Start the situation by selecting **Run at startup**.
- Select **Critical** for the state.
- Ensure that the situation is distributed to VM01 and VM02, not the LINUX\_SYSTEM group.
- Click **No** if prompted to update EIF settings.

This situation checks for the disk named **/dev/sda2** only if it has less than 25 GB of disk space free.

- Click **OK** and observe that the situation is **True** on both VM01 and VM02.

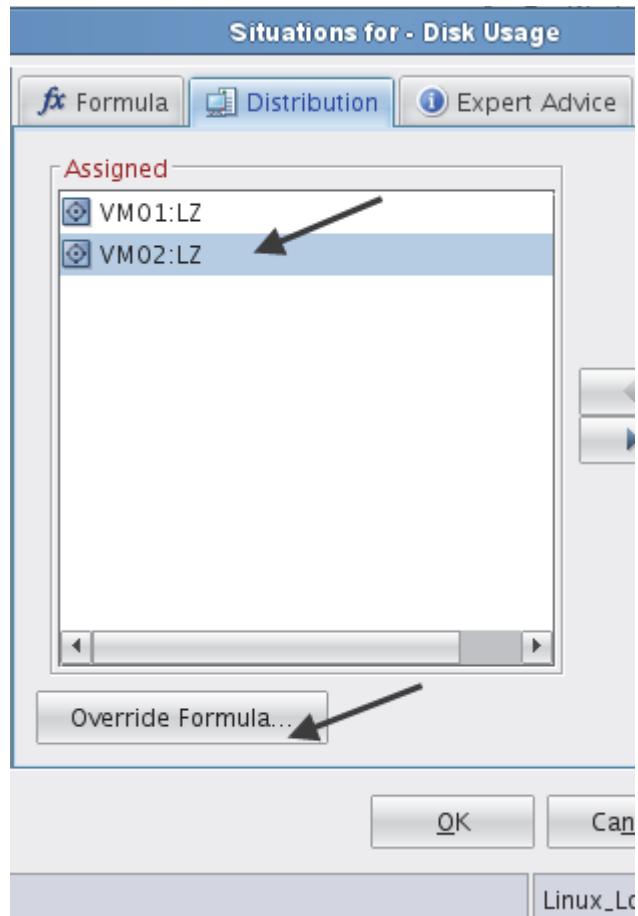
The screenshot shows the 'Situation Event Console' window. At the top, there are various icons and a toolbar. Below that is a header row with columns for Severity, Status, Owner, Name, Display Item, and Source. There are two rows of data in the table:

	Severity	Status	Owner	Name	Display Item	Source
	<b>Critical</b>	Open		Linux_Low_Space_Available_AAA	/dev/sda2	VM02:LZ
	<b>Critical</b>	Open		Linux_Low_Space_Available_AAA	/dev/sda2	VM01:LZ

*Exercise 4. Using dynamic thresholds*

You can apply Situation overrides on the **Distribution** tab. You can set different formula overrides for situations that are distributed to multiple systems. For this exercise, override the situation expressions for VM02 only.

12. Reaccess the situation. On the **Distribution** tab, click **VM02:LZ**. Click **Override Formula**.



13. Locate the **Reference formula** area.

The screenshot shows the 'Add Expression Overrides' dialog. In the 'Reference formula' section, there is a table with two columns: 'Space Available (MB)' and 'Disk Name'. The first row contains the expression ' $< 25000$ ' and the value ' $== '/dev/sda2'$ '. An arrow points to the 'Disk Name' column. Below this is the 'Formula overrides' section, which has tabs for 'Expressions' and 'Details'. Under 'Expressions', there is another table with columns for 'Disk Mount Point', 'Mount Point (Unicode)', 'Space Available (MB)', and 'Disk Name'. The 'Space Available (MB)' column contains the expression ' $< 25000$ ' and the value ' $== '/dev/sda2'$ .

These values represent the original situation settings and cannot be changed.

14. Examine the **Formula overrides** area. You can modify situation expressions for a specific managed system or managed system group. Only situation attributes that are eligible for overriding are in this area.
15. Assume that you want less restrictive settings on VM02 for this situation. Click the **Space Available** attribute and provide a value of 6000 instead of 25000.

The screenshot shows the 'Add Expression Overrides' dialog with the 'Formula overrides' section selected. The 'Expressions' tab is active. In the table under 'Expressions', the 'Space Available (MB)' column now contains the value '6000' instead of ' $< 25000$ '. An arrow points to this new value. The rest of the dialog remains the same as the previous screenshot.

16. Click **Apply**.

Space Available (MB)	Disk Name
< 6000	== '/dev/sda2'

17. Notice that the expression contains a check box in the first column. Select the check box. The **Remove Expression** button becomes active. You can remove an override expression that you no longer want.

Formula overrides

Expressions Details

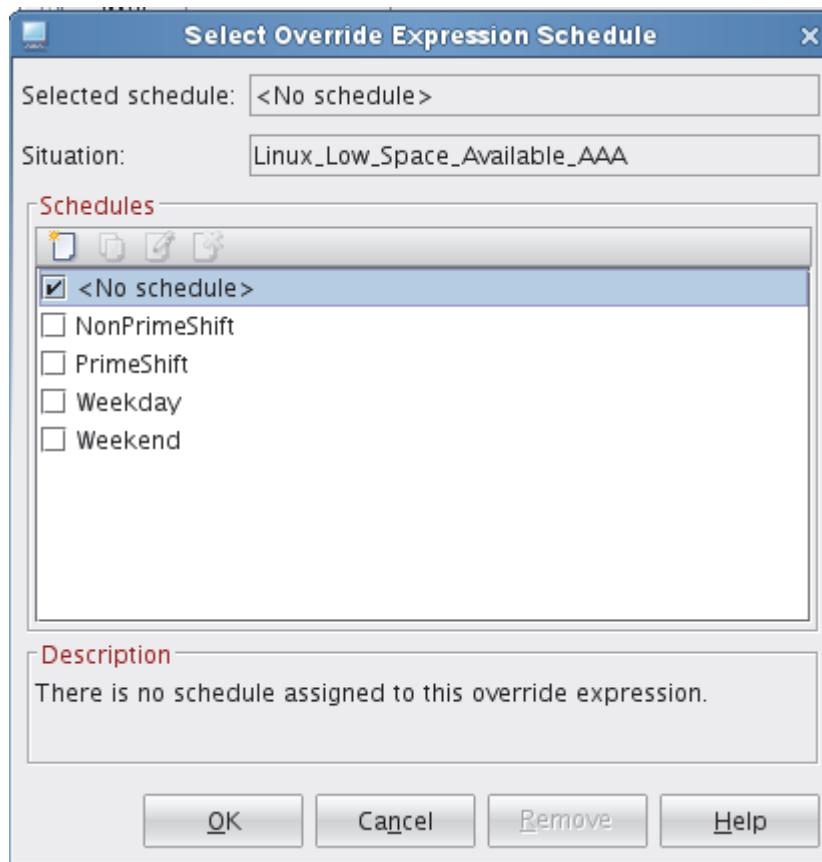
fx

	Disk Mount Point	Mount Point (Unicode)
<input checked="" type="checkbox"/>	< 6000	== '/dev/sda2'

Add Expression Remove Expression Model Expression...

18. Hold the mouse pointer over the calendar icon in the left pane. This icon is the **Override Schedule** button.

19. Click **Override Schedule** and examine the available choices.



20. Click each schedule to see a description of its values. These schedules are product-provided.  
You can add more calendar entries by using line commands. Do not select an override  
schedule now. Click **Cancel** and return to the Add Expression Overrides window.



**Note:** The product-provided schedules match the product-provided situations of the same names. You can examine those situations by accessing the **All Managed Systems** category in the Situation editor.

21. Click **OK** to save your expression overrides. Save the situation.
22. Validate your solution. Watch the Situation Event console and see which situations open. (It opens only on VM01 because you changed the threshold for VM02.)
23. Stop the situation that you created when you are satisfied that it is working correctly.



## 4 Agent autonomy exercises

In these exercises, you learn how to send SNMP traps to event receivers without a connection to a monitoring server. This feature has implications in the architecture of your solution, the scalability of IBM Tivoli Monitoring, and the way that the monitoring agents are configured. These exercises show you the basics of this function.

### Exercise 1. Monitoring without a Tivoli Enterprise Monitoring Server

You can create private situations in the following two ways:

- You can export an existing enterprise situation to XML and use that definition.
- You can create the XML by hand.

In this exercise, you learn how to use both methods.

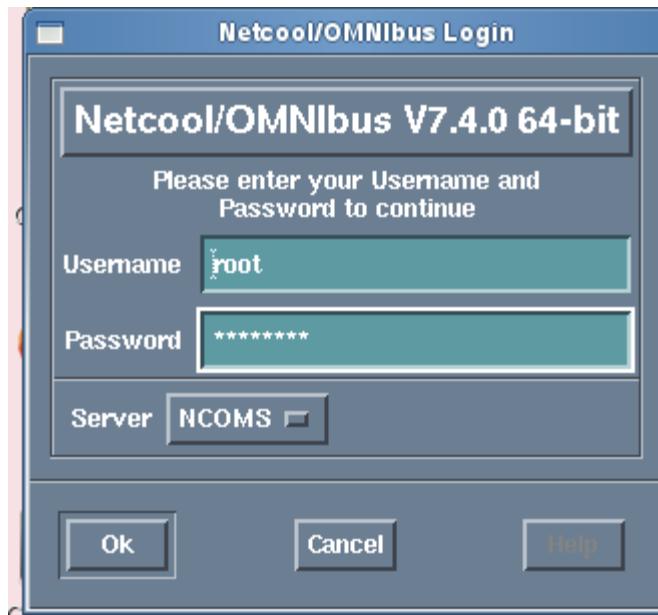
#### Creating a private situation by exporting an existing situation to XML

To create a private situation by exporting an existing enterprise situation to XML, perform the following steps:

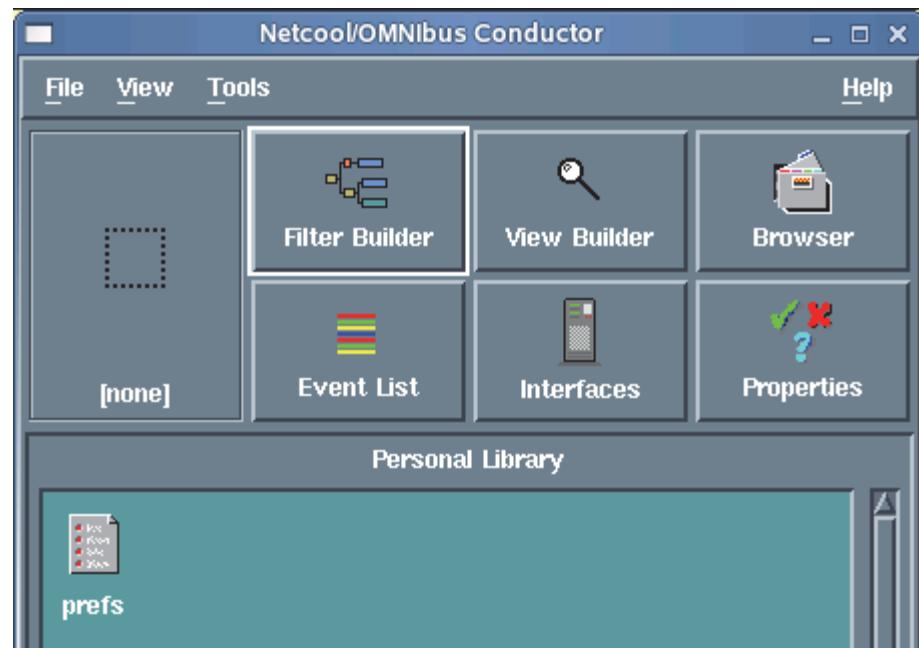
1. Log in to VM04 and open a terminal window.
2. Open the Netcool®/OMNIbus® console by entering the following commands:

```
cd $OMNIHOME  
cd bin  
. /nco &
```
3. Locate the Netcool/OMNIbus login window. It might be behind the splash screen.

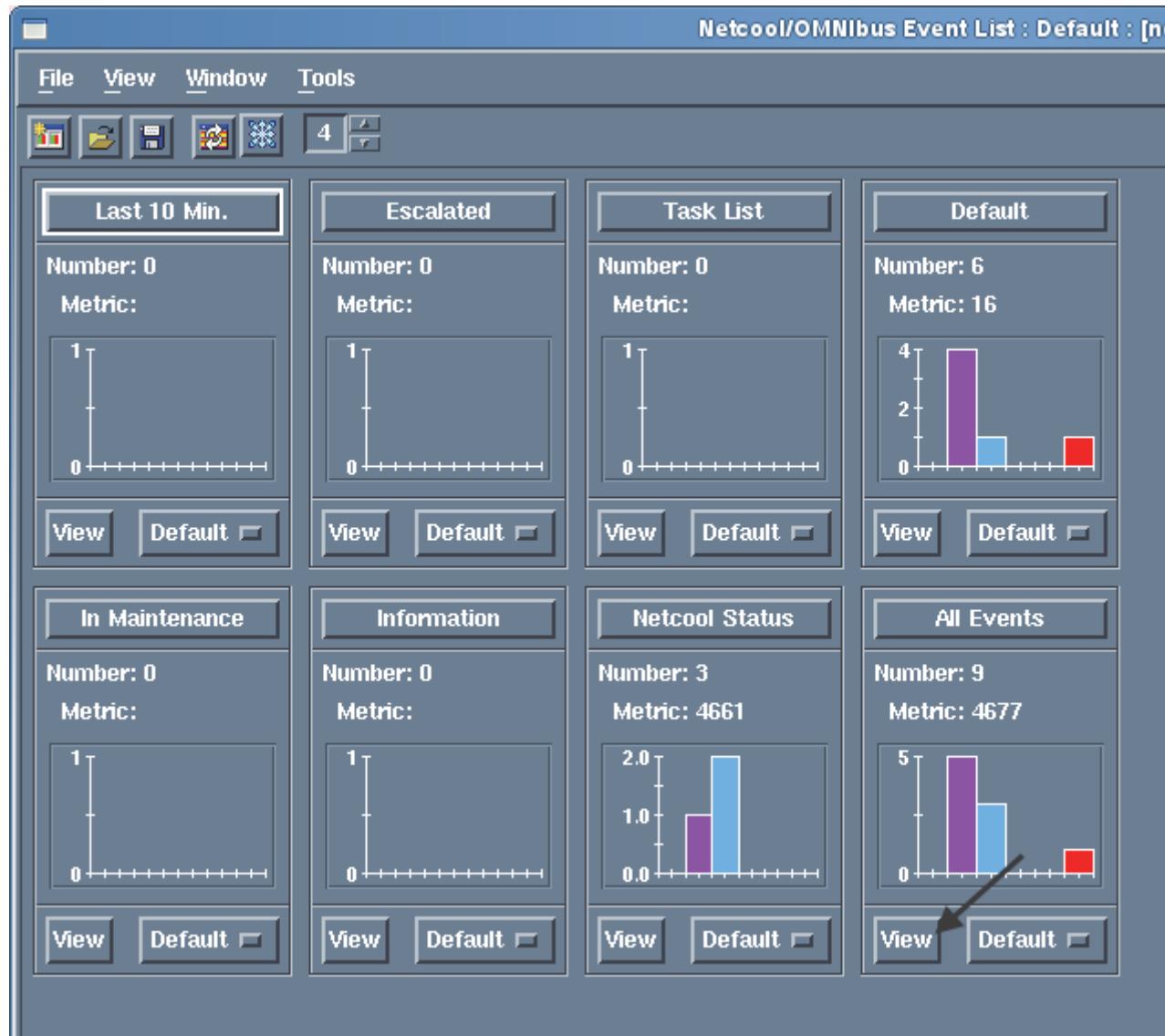
4. Enter a password of **object00** and click **OK**.



5. Click **Event List**.

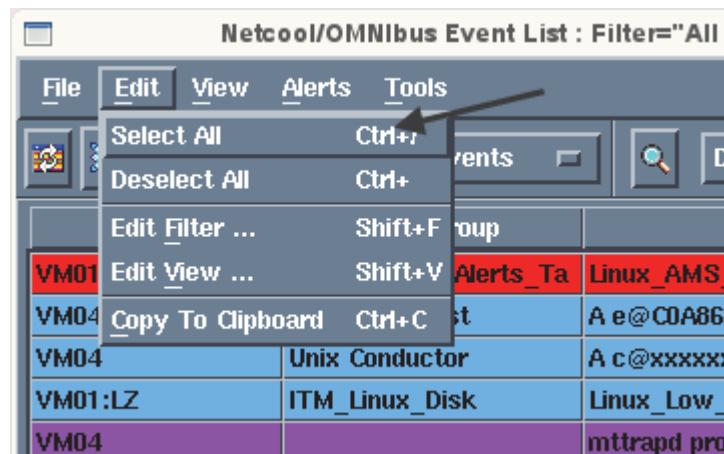


6. Expand the Event List window and click **View** in the All Events pane in the lower right.

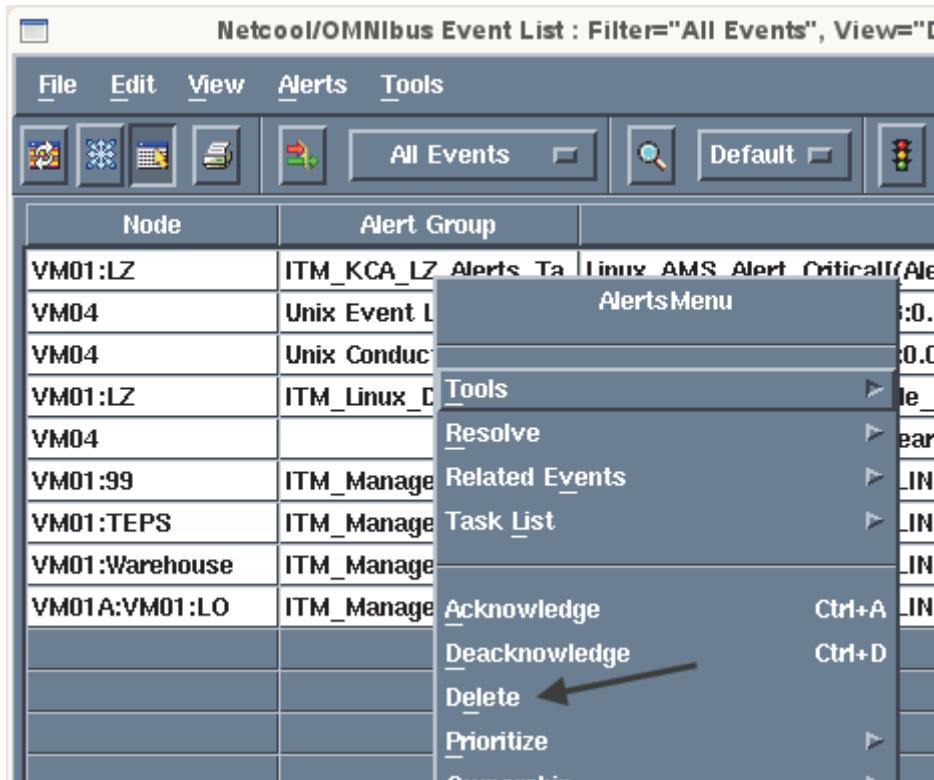


Your Event Viewer opens and looks like the following screen capture.

7. Select all the OMNIbus events by selecting **Edit > Select All**.



8. Clear the current OMNIbus events so that your exercises match the screen captures in this exercise. Right-click one of the events and select **Delete**.



9. Open a terminal session on **VM02** and change to the following directory:

```
/opt/IBM/ITM/bin
```

10. Enter the following command to stop the remote monitoring server:

```
./itmcmd server stop VM02_RTEMS
```

A key point to understand about agent autonomy is that the monitoring agent does not need to be connected to a monitoring server to function. The monitoring agent that is running on VM02 is connected to the monitoring server named RTEMS, which is running on VM02.

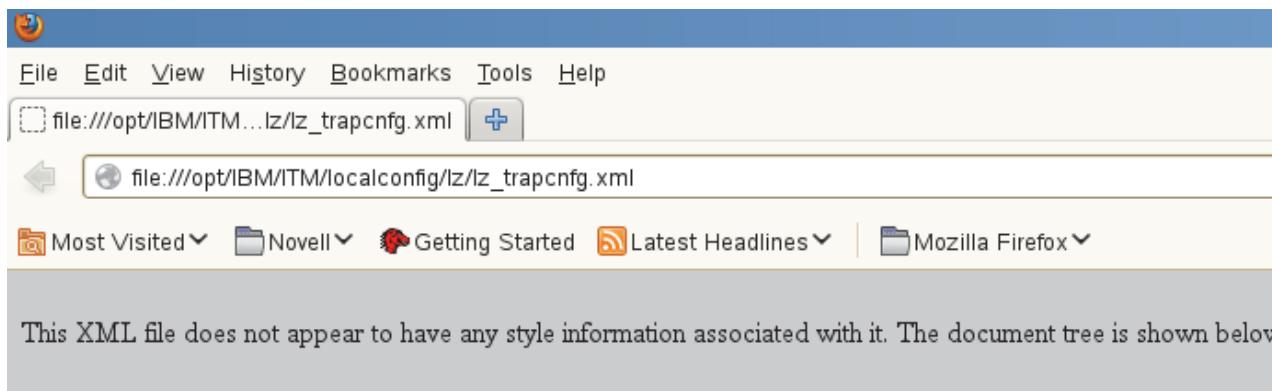
11. On VM02, open a new Gnome Terminal session and change the directory to this location:

```
/opt/IBM/ITM/localconfig/lz
```

12. Create the trap configuration file **lz\_trapcfg.xml** by using gedit. This file sends all situations to a Netcool/OMNIbus found on VM04.

```
<SNMP>
<TrapDest name="VM04" Address="192.168.100.104" BindAddress="192.168.100.102"
/>
<Situation name="*" target="VM04" />
<StatTrap name="EE_HEARTBEAT" sev="1" interval="1" cat="3" />
</SNMP>
```

13. Open the newly created **lz\_trapcfg.xml** file by using Firefox to validate that the XML is created correctly.



This XML file does not appear to have any style information associated with it. The document tree is shown below

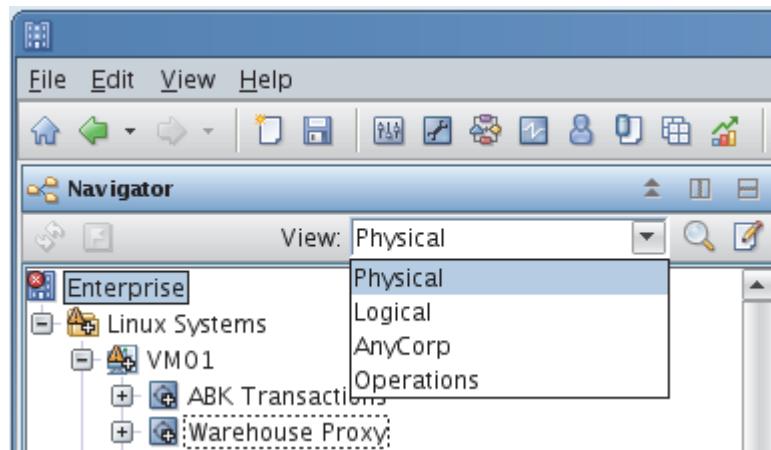
```
- <SNMP>
  <TrapDest name="VM04" Address="192.168.100.104" BindAddress="192.168.100.102"/>
  <Situation name="*" target="VM04"/>
  <StatTrap name="EE_HEARTBEAT" sev="1" interval="1" cat="3"/>
</SNMP>
```



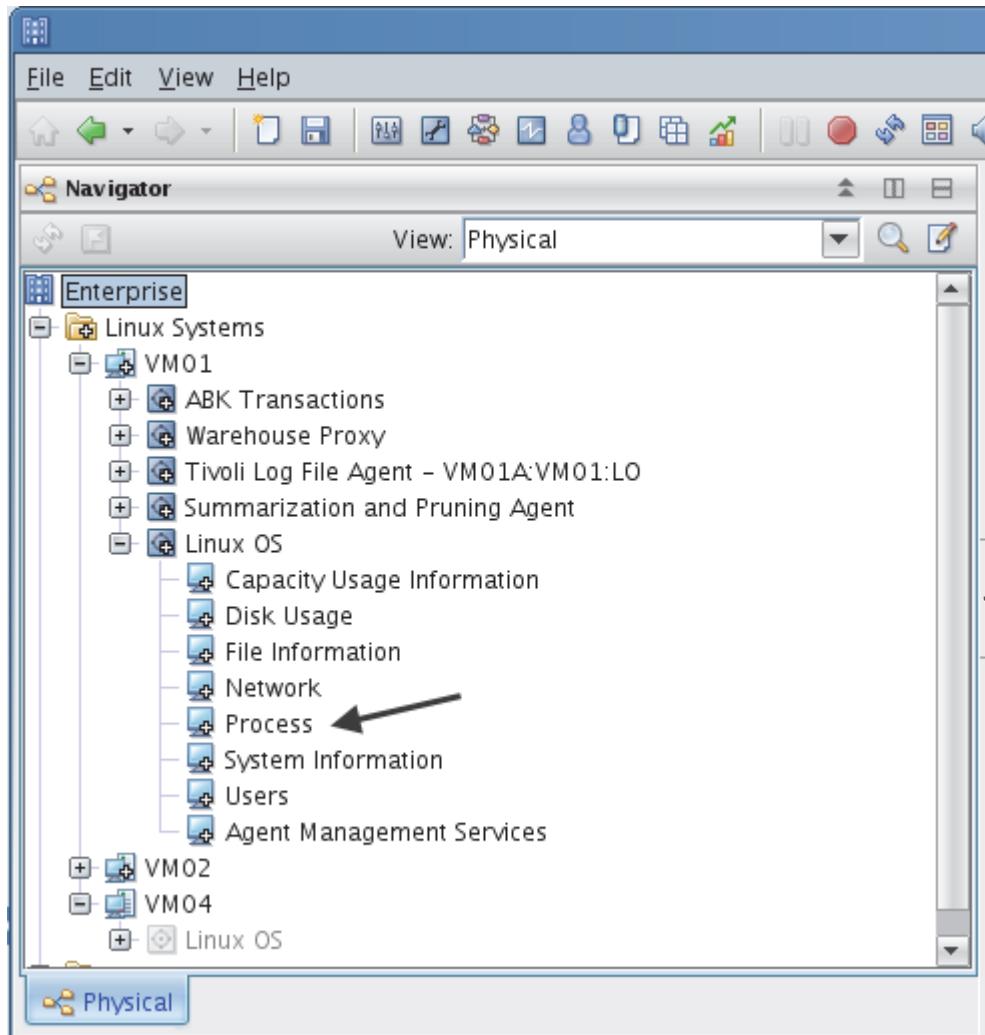
**Note:** You might get a syntax error or discover that later traps do not show up on VM04. If you have trouble getting the XML to parse correctly, you can copy a solution file from the following location on VM02.

`/labfiles/adv/autonomous`

14. On VM01, open the Navigator Physical view.



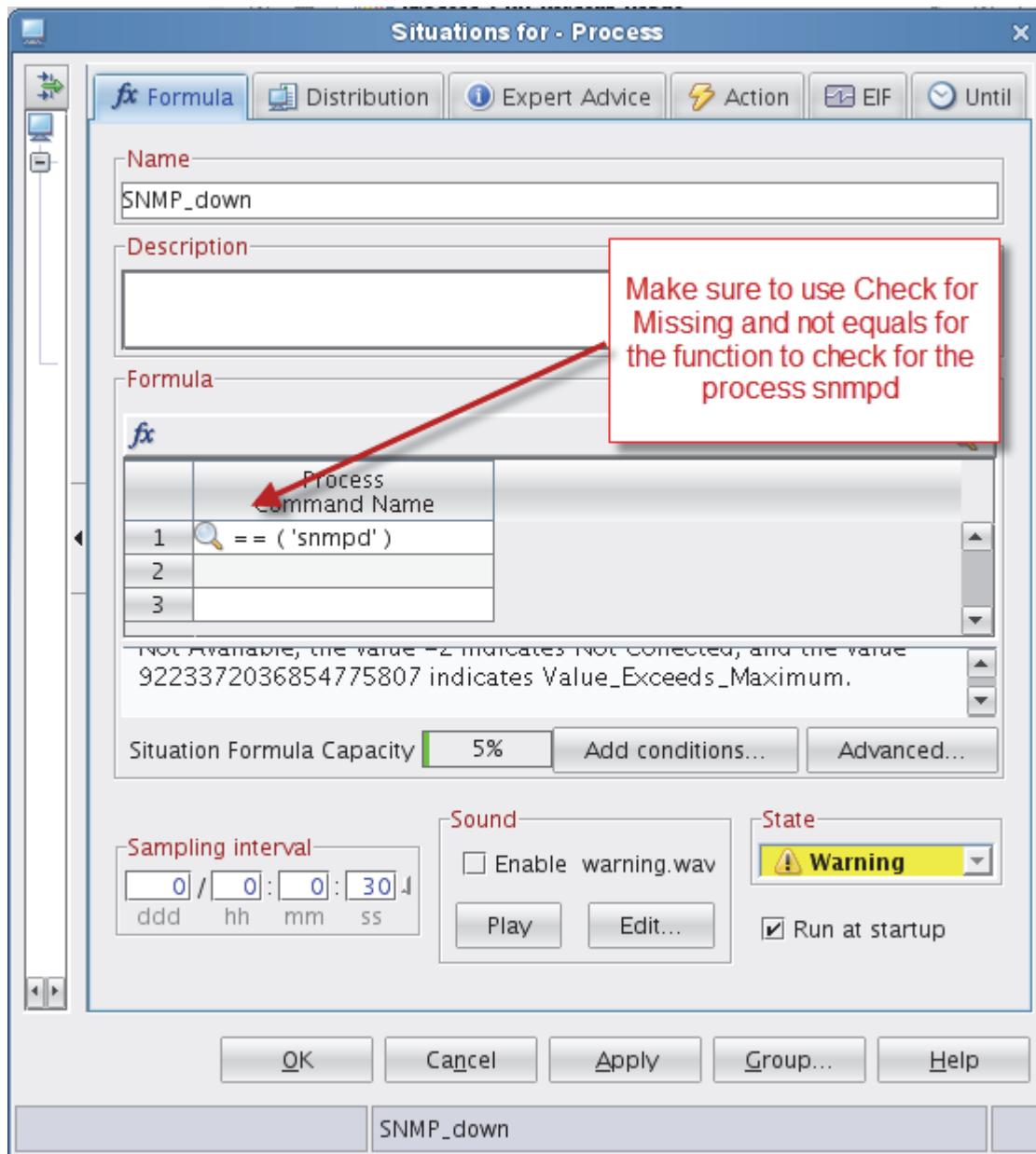
15. Go to **Linux Systems > VM01 > Linux OS > Process**.



16. Right-click **Process** and select **Situations**.

17. Using the Situation editor on VM01, create a situation named **SNMP\_down** on the Linux OS agent that monitors for the Linux process **snmpd**. This agent monitors every 30 seconds and sends a warning situation event if this process is down. The situation that you create is like the following screen capture. Use the Process Command Name attribute. Distribute this situation to

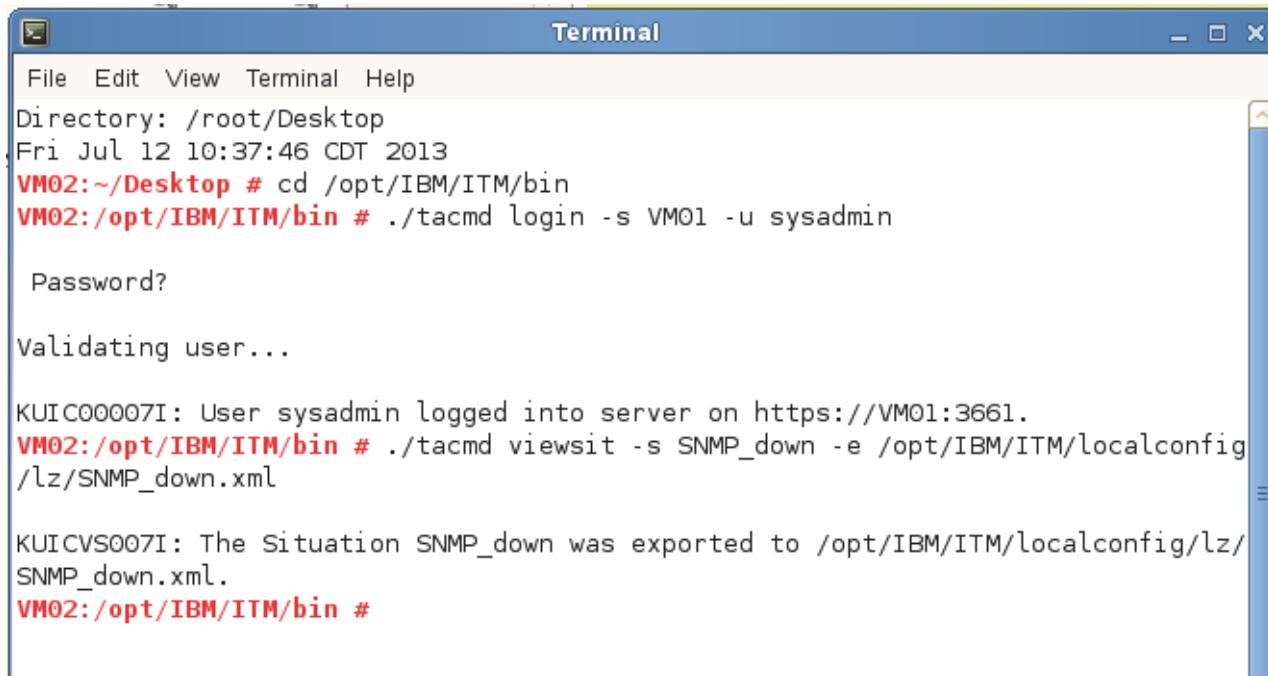
\***LINUX\_SYSTEM**, not VM01. If you distribute only to VM01, then the situation will not run on VM02, which is where you plan to run it.



**Note:** The situation name has uppercase and lowercase letters.

18. On VM02, use the following commands to log in to the hub monitoring server with a user ID of **sysadmin**. These commands also export the enterprise situation to an **.xml** file called **SNMP\_down** in the **/opt/IBM/ITM/localconfig/lz** directory.

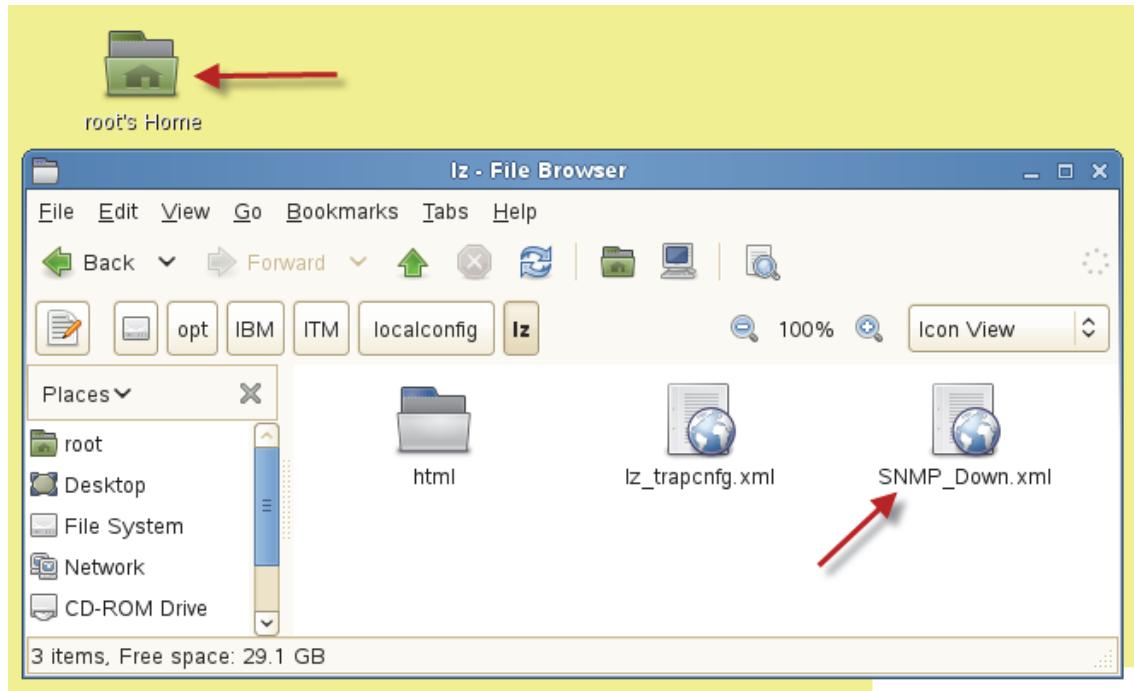
```
cd /opt/IBM/ITM/bin  
.tacmd login -s VM01 -u sysadmin  
.tacmd viewsit -s SNMP_down -e /opt/IBM/ITM/localconfig/lz/SNMP_down.xml
```



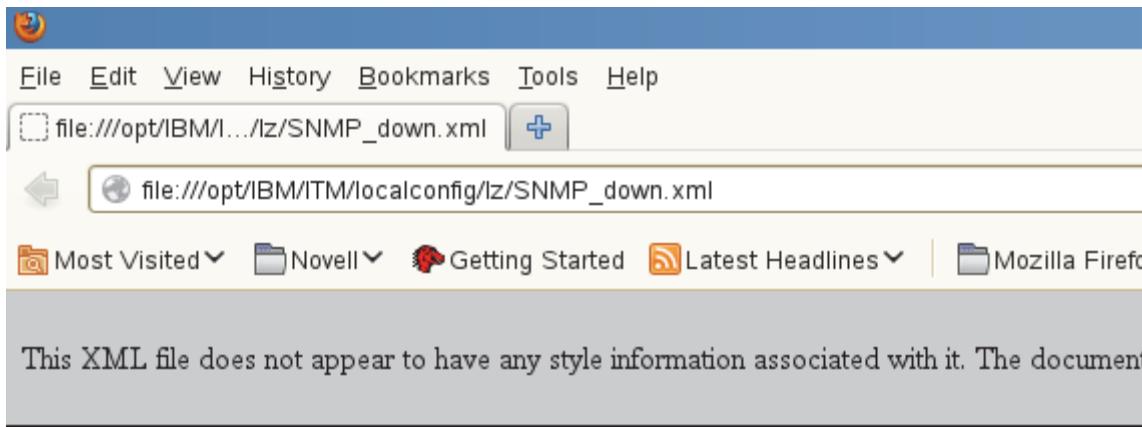
The screenshot shows a terminal window with the title "Terminal". The window contains the following text:

```
File Edit View Terminal Help  
Directory: /root/Desktop  
Fri Jul 12 10:37:46 CDT 2013  
VM02:~/Desktop # cd /opt/IBM/ITM/bin  
VM02:/opt/IBM/ITM/bin # ./tacmd login -s VM01 -u sysadmin  
  
Password?  
  
Validating user...  
  
KUIC00007I: User sysadmin logged into server on https://VM01:3661.  
VM02:/opt/IBM/ITM/bin # ./tacmd viewsit -s SNMP_down -e /opt/IBM/ITM/localconfig/lz/SNMP_down.xml  
  
KUICVS007I: The Situation SNMP_down was exported to /opt/IBM/ITM/localconfig/lz/SNMP_down.xml.  
VM02:/opt/IBM/ITM/bin #
```

19. Using the File Browser application, navigate to the **SNMP\_down.xml** file and inspect it by using gedit.



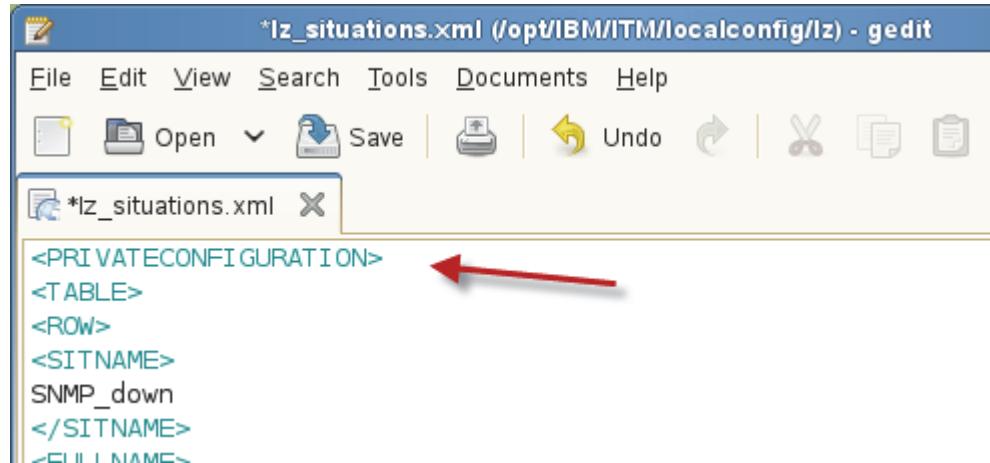
The file looks like the following screen capture.



```
- <TABLE>
  - <ROW>
    <SITNAME> SNMP_down </SITNAME>
    <FULLNAME> </FULLNAME>
    <TEXT> </TEXT>
    <AFFINITIES> %IBM STATIC134 0000000008##### </AFFINITIES>
  - <PDT>
    *IF *MISSING KLZ_Process.Process_Command_Name *EQ ('snmpd')
  </PDT>
  <REEV_DAYS> 0 </REEV_DAYS>
  <REEV_TIME> 000030 </REEV_TIME>
  <AUTOSTART> *YES </AUTOSTART>
  <ADVISE> *NONE </ADVISE>
  <CMD> *NONE </CMD>
  <AUTOSOPT> NNN </AUTOSOPT>
  <DISTRIBUTION> *LINUX_SYSTEM </DISTRIBUTION>
```

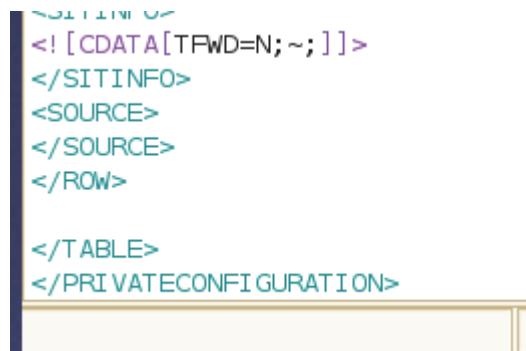
20. On VM02, copy the file **SNMP\_down.xml** to a file called **lz\_situations.xml** in the same **opt/IBM/ITM/localconfig/lz** directory. Add the following statement to the top of the **lz\_situations.xml** file:

```
<PRIVATECONFIGURATION>
```



21. Add the following line to the end of the **lz\_situations.xml** file and save the file:

```
</PRIVATECONFIGURATION>
```



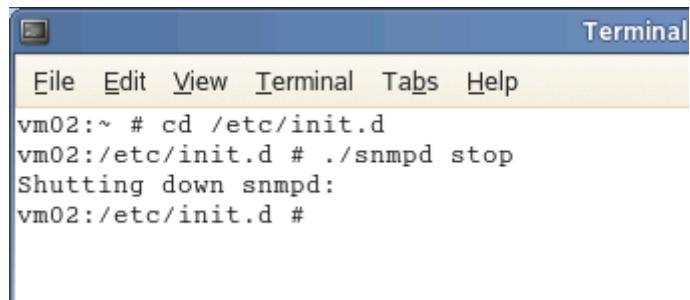
22. Recycle the Linux OS agent on VM02 by using the following commands:

```

cd /opt/IBM/ITM/bin
./itmcmd agent stop lz
./itmcmd agent start lz

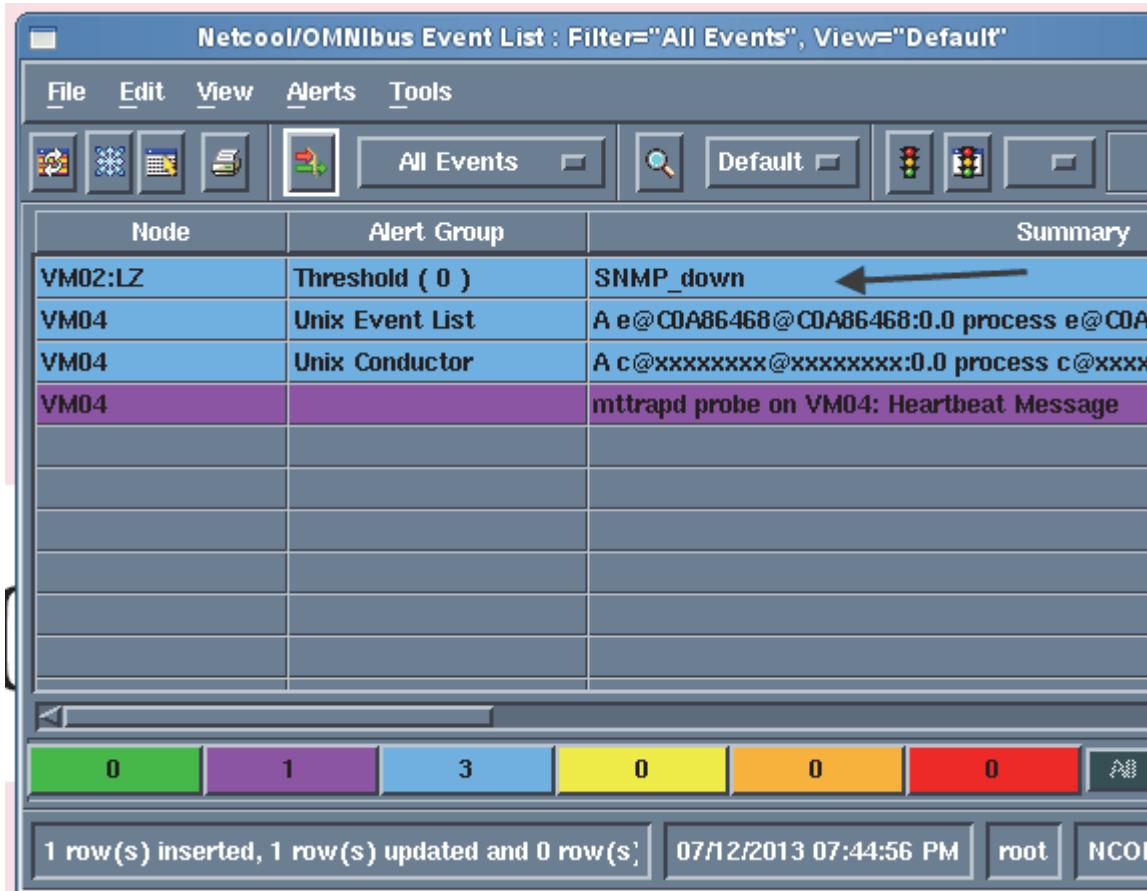
```

23. Shut down the SNMP daemon on VM02.



```
Terminal
File Edit View Terminal Tabs Help
vm02:~ # cd /etc/init.d
vm02:/etc/init.d # ./snmpd stop
Shutting down snmpd:
vm02:/etc/init.d #
```

24. On VM04, see if the SNMP\_down trap was sent to Netcool/OMNibus within the next minute.



The screenshot shows the Netcool/OMNibus Event List interface. The title bar reads "Netcool/OMNibus Event List : Filter="All Events", View="Default". The main area is a table with columns: Node, Alert Group, and Summary. There are four rows of data:

Node	Alert Group	Summary
VM02:LZ	Threshold ( 0 )	SNMP_down ←
VM04	Unix Event List	A e@C0A86468@C0A86468:0.0 process e@C0A
VM04	Unix Conductor	A c@xxxxxxxxxxxxxx@xxxxxxxx:0.0 process c@xxxx
VM04		mttrapd probe on VM04: Heartbeat Message

At the bottom, there are status indicators for various metrics (0, 1, 3, 0, 0, 0) and a message: "1 row(s) inserted, 1 row(s) updated and 0 row(s)". The timestamp is 07/12/2013 07:44:56 PM, and the user is root.

**Note:** If you have trouble getting the XML to parse correctly, you can copy a solution file from the following location:

/labfiles/adv/autonomous/fromEnterpriseSituation

# Creating a private situation manually

In this part of the exercise, you manually create a private situation to monitor the sshd daemon.

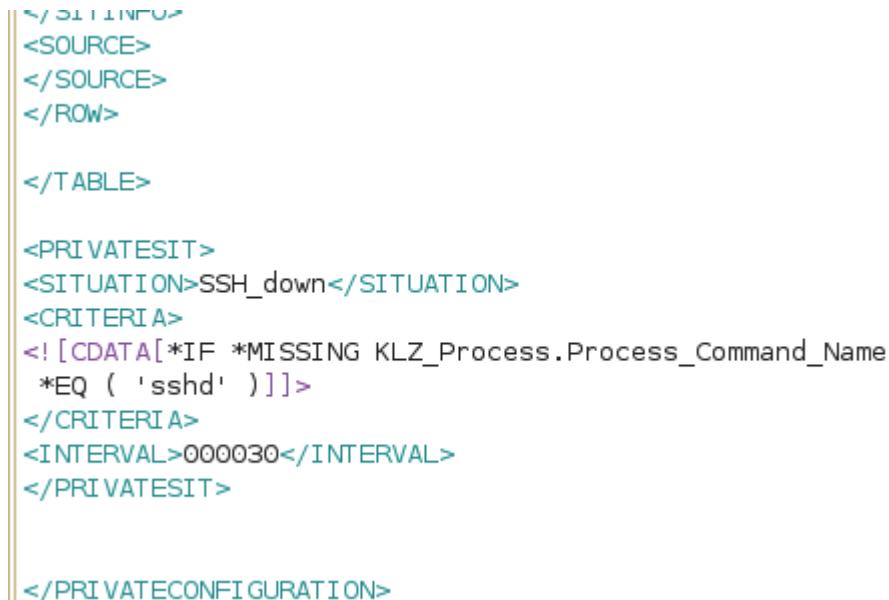
1. On VM02, use gedit to edit the file.

```
/opt/IBM/ITM/localconfig/lz/lz_situations.xml
```

2. Insert the following XML statements into the file after the </TABLE> statement, which is near the bottom of the file.

```
<PRIVATESIT>
<SITUATION>SSH_down</SITUATION>
<CRITERIA>
<! [CDATA[*IF *MISSING KLZ_Process.Process_Command_Name
*EQ ( 'sshd' )]]>
</CRITERIA>
<INTERVAL>000030</INTERVAL>
</PRIVATESIT>
```

The end of the **lz\_situations.xml** file now looks like the following screen capture.



```
</PRIVATECONFIGURATION>
```

3. Save the file.



**Note:** If you have trouble getting the XML to parse correctly, you can copy a solution file from the following location:

```
/labfiles/adv/autonomous/fromManuallyCreatedSituation
```

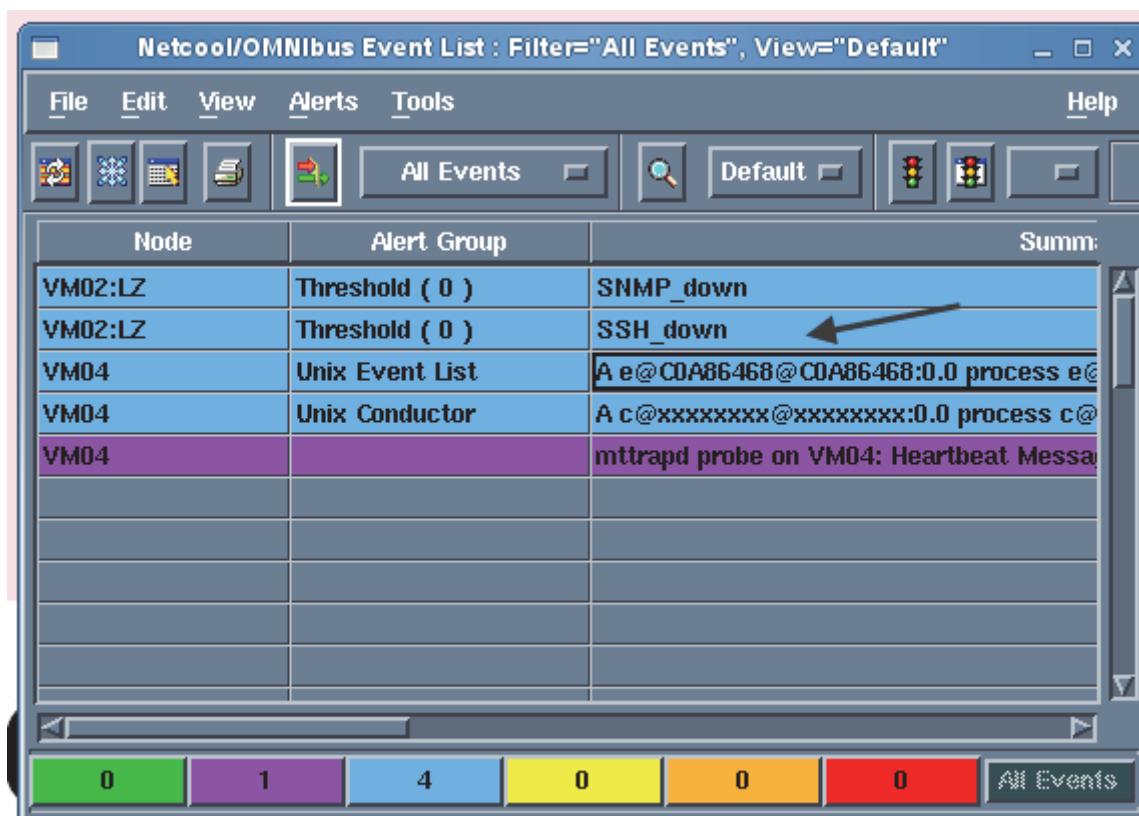
4. Start the ssh daemon on VM02 by using the following commands:

```
cd /etc/init.d  
./sshd start
```

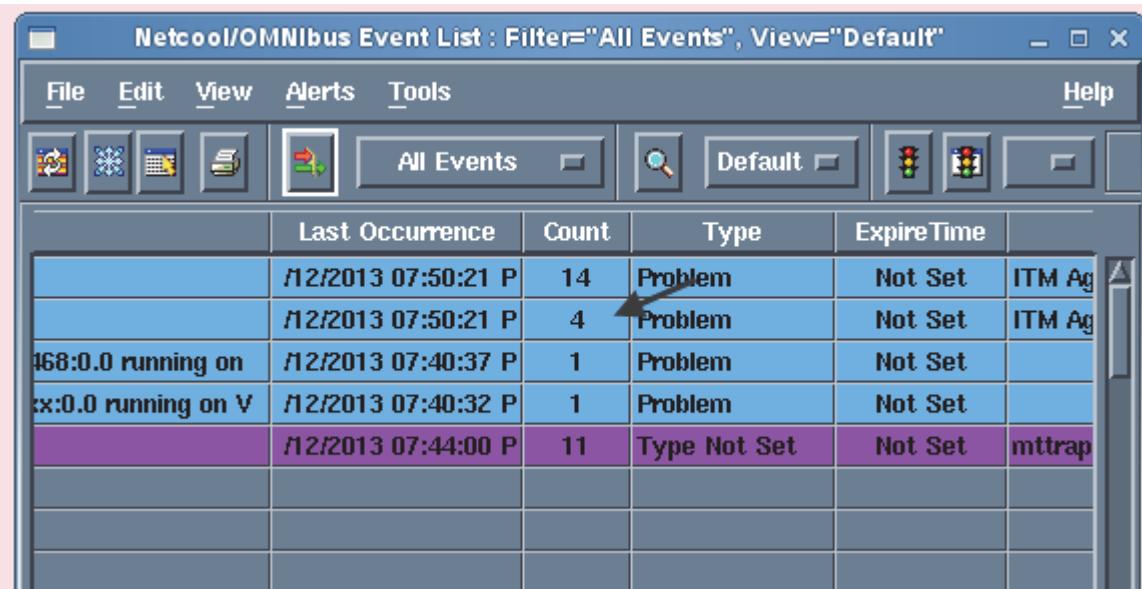
5. On VM02, open another terminal session. Stop and restart the Linux OS agent again to pick up the new definitions.
  6. Switch to VM04. Refresh the Netcool/OMNIbus console to verify that the private situation SSH\_Down event was not generated.
  7. Return to VM02 and close the sshd application.

```
VM02:/etc/init.d # ./sshd stop
Shutting down SSH daemon
VM02:/etc/init.d #
```

8. Return to VM04 and click **Refresh Events**. Wait until you see the event. If you do not see it, switch to VM02 and review the XML files that you created.



9. Wait a few minutes and notice that the count value increases.



10. Return to VM02 and restart the sshd application.

```
shutting down sshd daemon  
VM02:/etc/init.d # ./sshd start  
Starting SSH daemon  
VM02:/etc/init.d #
```

The count does not increase. However, by default, the trap is not cleared. For the trap to clear with a resolution trap, you update the **Iz\_trapconfig.xml** file.

11. On VM02, gedit the `lz_trapcfg.xml` file in `/opt/IBM/ITM/localconfig/lz`.

```
/opt/IBM/ITM/localconfig/lz # qedit lz trapcnfg.xml
```

12. Insert the following line into the file after the <TrapDest> statement, and save the file:

```
<situation name="SSH down" Target="VM04" Mode="HY" Sev="2" />
```





**Note:** The statement that you added sets the SSH\_down private situation to a mode called *Hysteresis mode*. This mode causes a resolution trap to be sent to Netcool/OMNibus when the situation is no longer true. The event closes in Netcool/OMNibus, instead of staying open and having the count no longer increase.

13. On VM02, stop and restart the Linux OS again to pick up the new definitions.
14. Ensure that the sshd daemon is down and locate the trap in Netcool/OMNibus. After the initial trap, the count should no longer rise.
15. Restart the sshd daemon and locate the resolution trap.

Summary	Last Occurrence	Count	Type
SNMP_Down	7/25/2013 09:56:15 P	30	Problem
Linux_Process_High_Cpu	7/25/2013 09:51:15 P	3	Problem
Linux_AMS_Alert_Critical ()	7/25/2013 09:51:04 P	2	Problem
A e@C0A86468@C0A86468:0.0 process e@C0A86468@C0A86468:0.0 running on	7/25/2013 09:02:51 P	1	Problem
A c@xxxxxxxx@xxxxxxxx:0.0 process c@xxxxxxxx@xxxxxxxx:0.0 running on V	7/25/2013 09:02:48 P	1	Problem
mttrapd probe on VM04: Heartbeat Message	7/25/2013 09:40:46 P	475	Type Not Set
SSH_down	7/25/2013 09:56:15 P	→	Resolution
SSH_down	7/25/2013 09:55:45 P	6	Problem



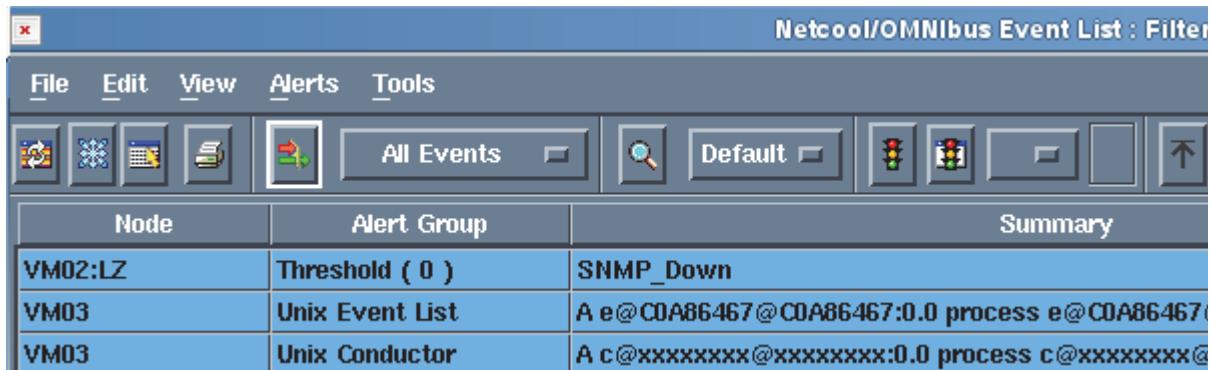
**Note:** If you have trouble getting the XML to parse correctly, you can copy a solution file from the following location:

/labfiles/adv/autonomous/withHysteresisEnabled

## Exercise 2. Using the Event Integration Facility

Agent autonomy now supports the use of the Event Integration Facility (EIF) and the standard trap interface. Typically, you use one or the other, but not both. However, in this exercise, you learn both interfaces. VM04 has both the Event Integration Facility and the trap interface enabled.

1. Open the event viewer on VM04 and locate the SNMP\_Down event.

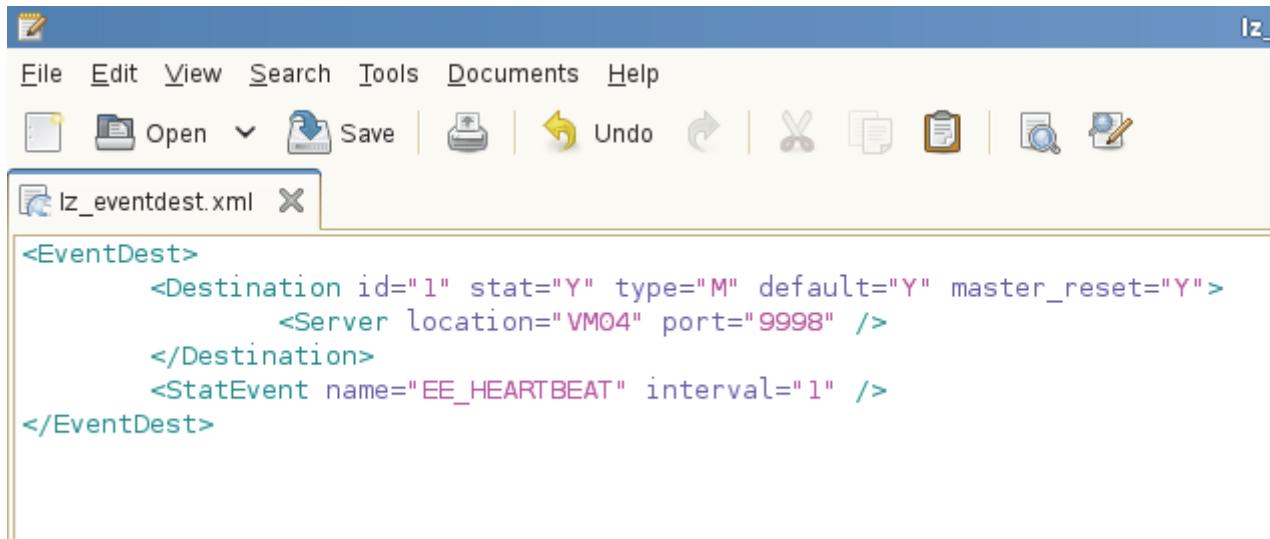


2. Scroll to the right and locate the Manager column. The Manager for this private situation is **mttrapd**. You created this situation earlier and sent it over as a trap. The name of the probe that received the trap is mttrapd.

3. On VM02, use gedit to create the file **lz\_eventdest.xml** in the **/opt/IBM/ITM/localconfig/lz** directory.

```
<EventDest>
  <Destination id="1" stat="Y" type="M" default="Y" master_reset="Y">
    <Server location="VM04" port="9998" />
  </Destination>
  <StatEvent name="EE_HEARTBEAT" interval="1" />
</EventDest>
```

This file forwards all private situations to VM04 on port 9998, which is the port that the Event Integration Facility probe is listening to.



4. Stop and restart the Linux OS agent on VM02.

5. Stop the SSH daemon, if it is still active, and locate the **SSH\_down** trap. This trap is in the list twice; once for each of the probes.

Node	Alert Group	Summary
VM02:LZ	Threshold ( 0 )	SNMP_down
VM02:LZ	Threshold ( 0 )	SSH_down
VM02:LZ	Threshold ( 0 )	Linux_AMS_Alert_Critical ()
VM04	Unix Event List	A e@C0A86468@C0A86468:0.0 process e@C0A8646
VM04	Unix Conductor	A c@xxxxxxxxx@xxxxxxxx:0.0 process c@xxxxxxxx
VM02:LZ	ITM_KLZ_Process	SSH_down[Process_Command_Name = sshd ]
VM04		mttrapd probe on VM04: Heartbeat Message

 **Note:** If you have trouble getting the XML to parse correctly, you can copy a solution file from the following location:

`/labfiles/adv/autonomous/eif/lz_eventdest.xml`

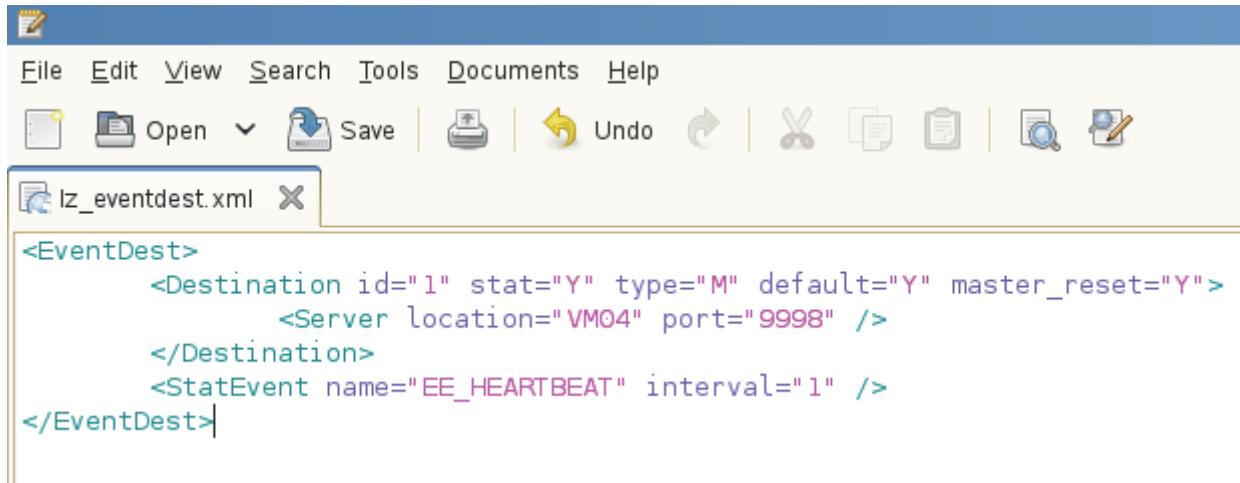
## Exercise 3. Agent autonomy heartbeats

The Event Integration Facility event destination XML file can include a StatEvent element. This element sends the online or offline status of the monitoring agent to the event server. You can customize the provided heartbeat rules or write your own to handle the EIF heartbeat events.

The destination server receives an EIF event with a class name of ITM\_Heartbeat. ITM\_Heartbeat contains a slot called *interval*, whose value is the heartbeat interval. The received SNMP events contain the attribute **AlertGroup**, whose value is **ITM\_Heartbeat** and an attribute

**HeartbeatInterval**, whose value is the heartbeat interval. The situation\_eventdata slot is also set to the heartbeat interval.

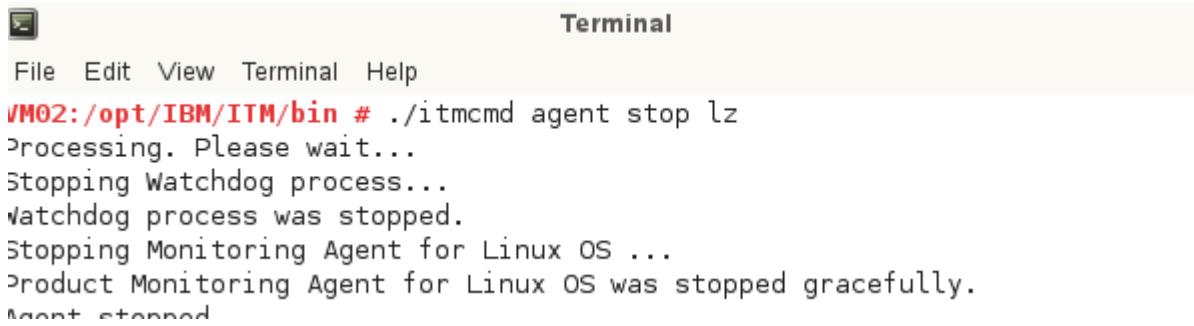
1. Observe that the **lz\_eventdest.xml** file has heartbeating enabled.



```
<EventDest>
    <Destination id="1" stat="Y" type="M" default="Y" master_reset="Y">
        <Server location="VM04" port="9998" />
    </Destination>
    <StatEvent name="EE_HEARTBEAT" interval="1" />
</EventDest>
```

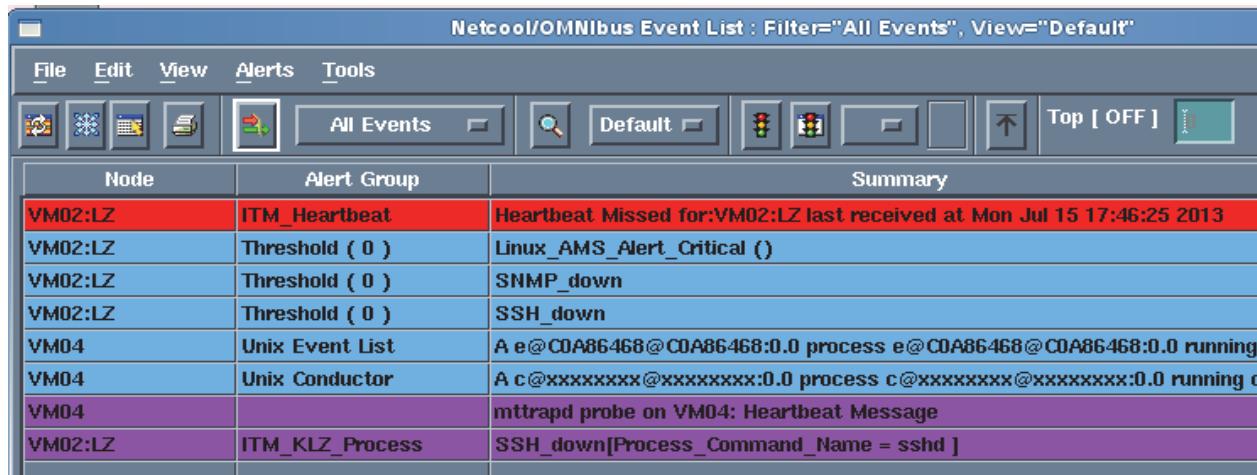
2. On VM02, close the Linux OS Agent by using the following command:

```
./itmcmd agent stop lz
```



```
VM02:/opt/IBM/ITM/bin # ./itmcmd agent stop lz
Processing. Please wait...
Stopping Watchdog process...
Watchdog process was stopped.
Stopping Monitoring Agent for Linux OS ...
Product Monitoring Agent for Linux OS was stopped gracefully.
Agent stopped
```

3. On VM04, look in the Event Viewer and notice the messages that arrive 2 - 3 minutes after the agent is stopped.

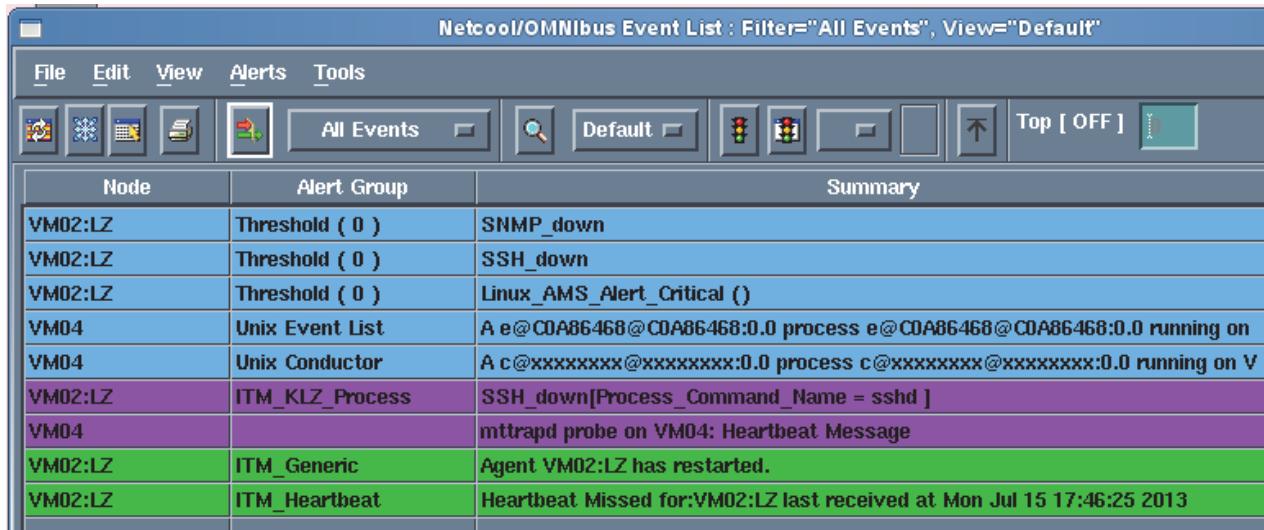


Node	Alert Group	Summary
VM02:LZ	ITM_Heartbeat	Heartbeat Missed for:VM02:LZ last received at Mon Jul 15 17:46:25 2013
VM02:LZ	Threshold ( 0 )	Linux_AMS_Alert_Critical ()
VM02:LZ	Threshold ( 0 )	SNMP_down
VM02:LZ	Threshold ( 0 )	SSH_down
VM04	Unix Event List	A e@C0A86468@C0A86468:0.0 process e@C0A86468@C0A86468:0.0 running
VM04	Unix Conductor	A c@xxxxxxxx@xxxxxxxx:0.0 process c@xxxxxxxx@xxxxxxxx:0.0 running c@xxxxxxxx@xxxxxxxx:0.0 running
VM04		mtrrapd probe on VM04: Heartbeat Message
VM02:LZ	ITM_KLZ_Process	SSH_down[Process_Command_Name = sshd ]

- On VM02, restart the Linux OS agent by using the following command:

```
./itmcmd agent start lz
```

When the heartbeat is received, the resolution is displayed in the Event Viewer on VM04.



The screenshot shows the Netcool/OMNIbus Event List interface. The title bar reads "Netcool/OMNIbus Event List : Filter="All Events", View="Default"". The menu bar includes File, Edit, View, Alerts, and Tools. Below the menu is a toolbar with icons for search, refresh, and other functions. The main area is a table with three columns: Node, Alert Group, and Summary. The table contains the following data:

Node	Alert Group	Summary
VM02:LZ	Threshold ( 0 )	SNMP_down
VM02:LZ	Threshold ( 0 )	SSH_down
VM02:LZ	Threshold ( 0 )	Linux_AMS_Alert_Critical ()
VM04	Unix Event List	A e@CDA86468@CDA86468:0.0 process e@CDA86468@CDA86468:0.0 running on
VM04	Unix Conductor	A c@xxxxxxxx@xxxxxxxx:0.0 process c@xxxxxxxx@xxxxxxxx:0.0 running on V
VM02:LZ	ITM_KLZ_Process	SSH_down[Process_Command_Name = sshd ]
VM04		mttrapd probe on VM04: Heartbeat Message
VM02:LZ	ITM_Generic	Agent VM02:LZ has restarted.
VM02:LZ	ITM_Heartbeat	Heartbeat Missed for:VM02:LZ last received at Mon Jul 15 17:46:25 2013

## Exercise 4. Implementing private historical data collection

Another important aspect of autonomous agents is the ability to collect data that can be retained at the agent for a user-specified amount of time. You can collect historical data by using the situation definitions as you did when you created the private situation in the earlier exercise.

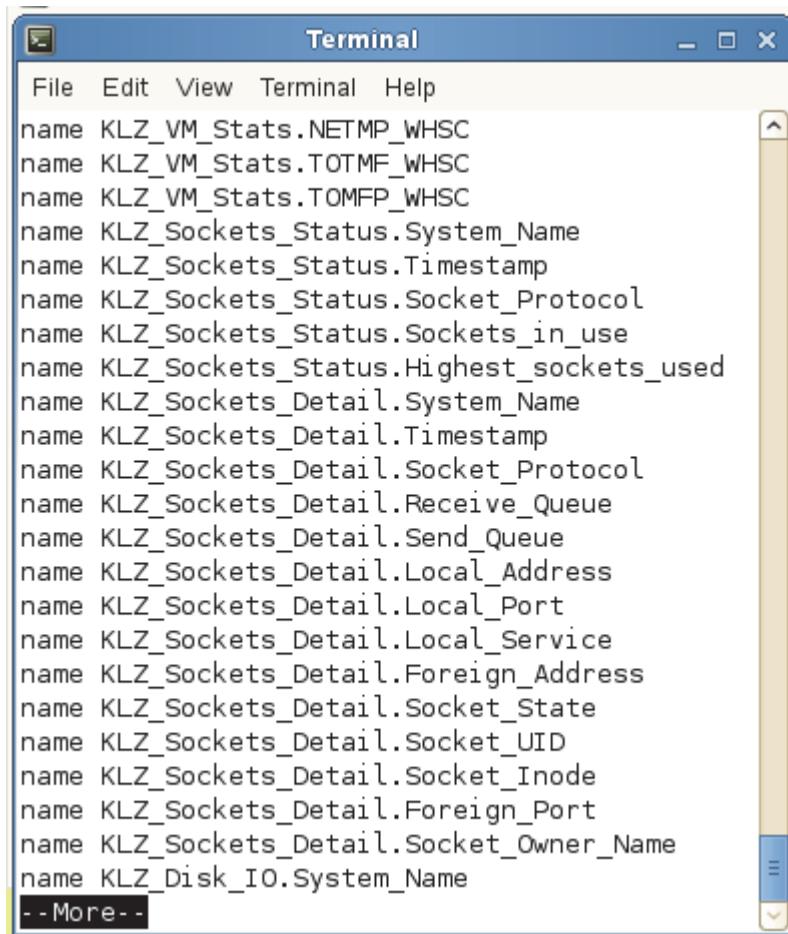
- On VM02, open a terminal session and change the directory:

```
/opt/IBM/ITM/li6263/lz/tables/ATTRLIB/
```

- Locate the **klz.atr** file in this directory. This file is the attribute file for the Linux OS agent. It defines which attributes are collected by the agent. Run the following command to see the attributes that are collected by the Linux OS agent:

```
cat klz.atr | grep KLZ_ | more
```

The result of this command shows you the attribute group names that can be used to collect historical data in the **lz\_situations.xml** file.



A screenshot of a terminal window titled "Terminal". The window contains a list of attribute group names, each starting with "name" followed by a specific name like "KLZ\_VM\_Status.NETMP\_WHSC", "KLZ\_VM\_Status.TOTMF\_WHSC", etc. At the bottom of the list, there is a "- More --" indicator.

```
File Edit View Terminal Help
name KLZ_VM_Status.NETMP_WHSC
name KLZ_VM_Status.TOTMF_WHSC
name KLZ_VM_Status.TOMFP_WHSC
name KLZ_Sockets_Status.System_Name
name KLZ_Sockets_Status.Timestamp
name KLZ_Sockets_Status.Socket_Protocol
name KLZ_Sockets_Status.Sockets_in_use
name KLZ_Sockets_Status.Highest_sockets_used
name KLZ_Sockets_Detail.System_Name
name KLZ_Sockets_Detail.Timestamp
name KLZ_Sockets_Detail.Socket_Protocol
name KLZ_Sockets_Detail.Receive_Queue
name KLZ_Sockets_Detail.Send_Queue
name KLZ_Sockets_Detail.Local_Address
name KLZ_Sockets_Detail.Local_Port
name KLZ_Sockets_Detail.Local_Service
name KLZ_Sockets_Detail.Foreign_Address
name KLZ_Sockets_Detail.Socket_State
name KLZ_Sockets_Detail.Socket_UID
name KLZ_Sockets_Detail.Socket_Inode
name KLZ_Sockets_Detail.Foreign_Port
name KLZ_Sockets_Detail.Socket_Owner_Name
name KLZ_Disk_IO.System_Name
-- More --
```



**Note:** Use Ctrl+C to end the command.

3. Open another terminal session and change to the following directory:

```
/opt/IBM/ITM/localconfig/lz
```

4. Edit the **lz\_situations.xml** file by inserting the following line inside the **<PRIVATECONFIGURATION>** stanza, immediately following the **</PRIVATESET>** statement:

```
<HISTORY TABLE="KLZ_Sockets_Detail" INTERVAL="5" RETAIN="48" />
```

This single line collects private historical data for the **KLZ\_Process** attribute group every five minutes. The data is retained for 48 hours.

The updated **lz\_situations.xml** file now looks like the following screen capture.

```

<PRIVATESIT>
<CONDITION>SSH_down</CONDITION>
<CRITERIA>
<! [CDATA[ *IF *MISSING KLZ_Process.Process_Command_Name
*EQ ( 'sshd' ) ]]>
</CRITERIA>
<INTERVAL>000030</INTERVAL>
<HISTORY TABLE="KLZ_Process" INTERVAL="5" RETAIN="48" />
</PRIVATESIT>
</PRIVATECONFIGURATION>

```

5. Stop and restart the Linux OS agent.
6. Locate the directory **/opt/IBM/ITM/logs** and run the following command:

more VM02:LZ.LG0



**Note:** The last character in the command is a zero, not the letter O.

7. Look at the output to see whether the XML files are parsed correctly. Locate a series of lines that look like the ones in the following example.

```

File Edit View Terminal Help
1130715152224921KRAA002I Agent Autonomous Mode initialized - Saved Events 50 Limit 2 Unit 1000000
1130715152224927KRAS023I SNMP trap configuration successful. 1 SNMP trap destinations and 3 situations defined
1130715152224928KRAS030I The SNMP heartbeat interval is set to 1.
1130715152224928KRAS025I SNMP trap emitter task started and ready
1130715152224968KRAF060I EIF destination <1> server <VM04> port <9998> SSL <N> successfully defined
1130715152224968KRAF043I EIF event configuration successful. 1 EIF destination(s) defined
1130715152224976KRAE054I The EIF heartbeat interval is set to 1.
1130715152224976KRAS045I EIF event emitter task started and ready
1130715152224976KRAX014E Threshold XML override document object name not defined
1130715152224977KRAX155E History data collection PVTHIST_KLZ_KLZSOCKD INTERVAL 1 minutes RETAIN 48 hours EXPORT 15 minutes USE Warehouse XML Parser
1130715152224981KRAIRA000 Starting Private situation PVTHIST_KLZ_KLZSOCKD <1,2377123811> for KLZ.KLZSOCKD.
1130715152224981KRAIRA000 Starting Private situation SNMP_down <1,2377123810> for KLZ.KLZPROC.
--More--(27%)

```

The private history is kept in the following directory:

/opt/IBM/ITM/li6263/lz/hist

Within a few minutes, the binary file **PVTHIST\_KLZSOCKD** is created.



**Note:** If you have trouble getting the XML to parse correctly, you can copy a solution file from the following location:

`/labfiles/adv/autonomous/PrivateHistory/lz_situations.xml`

## Exercise 5. Using the Agent Service interface

Even though you collected private history, you cannot see it in the Tivoli Enterprise Portal. To see the history that you collected, you must access it by using the SOAP interface. The Agent Service interface is an implementation of the SOAP interface that you can use to look at private history, agent information, and other items.

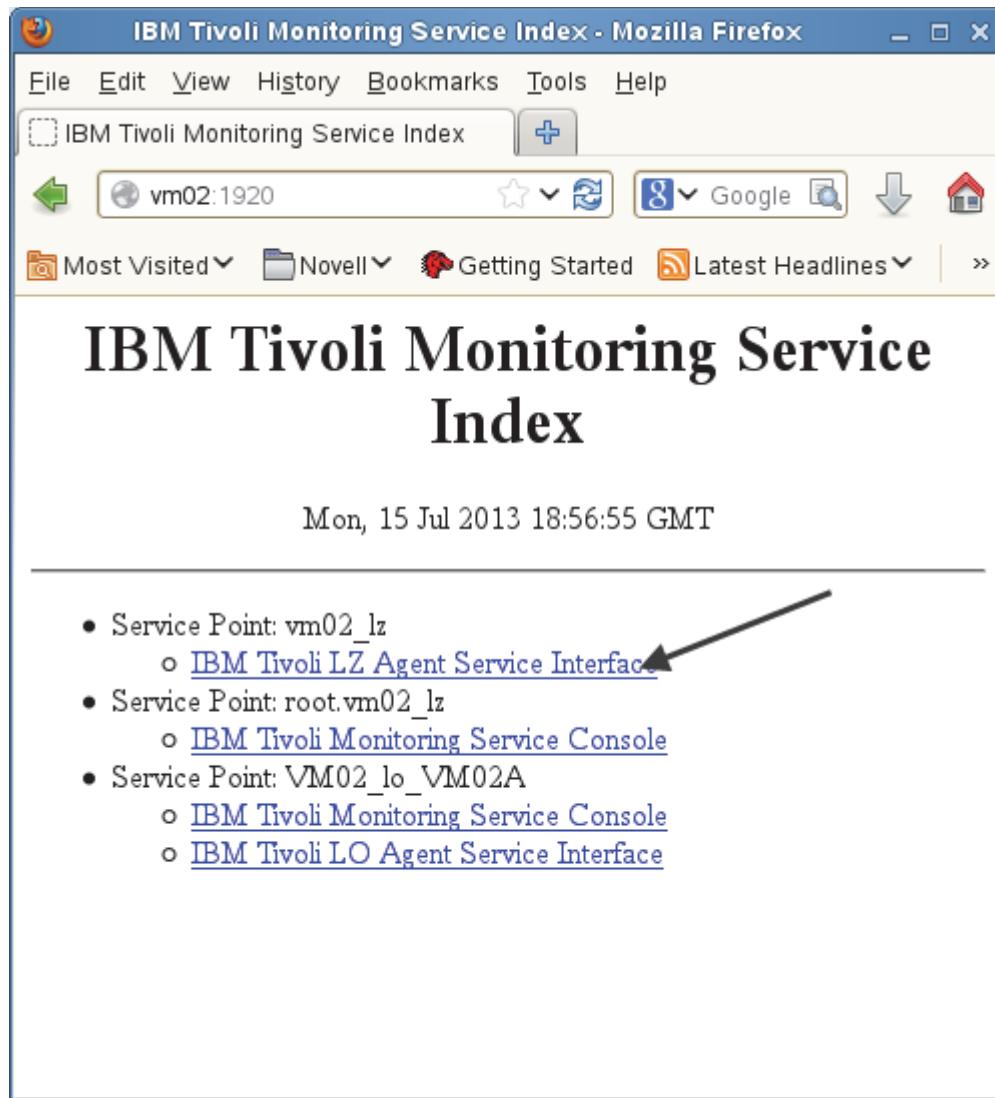
1. Return to VM01 and open a Firefox web browser to the following web address:

`http://VM02:1920`

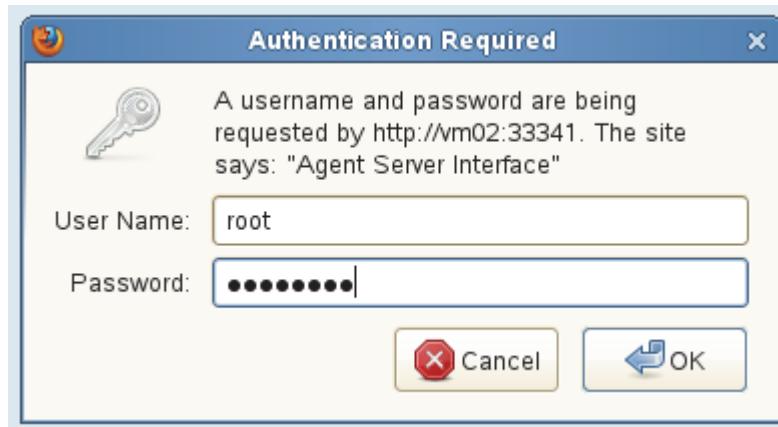


**Note:** Be sure to point the browser to VM02, and not VM01.

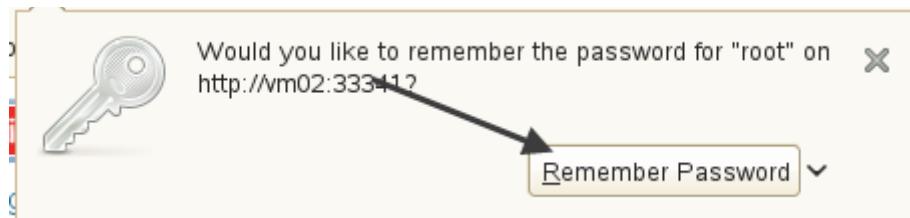
2. Click the **IBM Tivoli LZ Agent Service Interface** link.



3. Enter **root** and **object00** for the credentials and click **OK** when challenged for credentials.



4. Click **Remember Password**.



The following page is displayed.

5. Click the **Agent Information** link. Review the information provided.
6. Click **Back** in the browser.
7. Click the **Situations** link and browse through the enterprise and private situations.
8. Click **Back** in the browser.

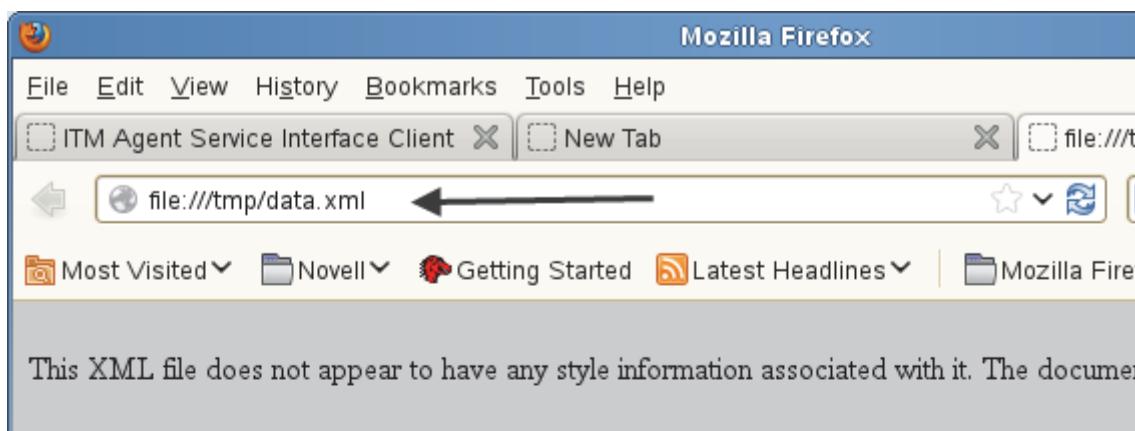
9. Click the **History** link.
10. Generate a history report by selecting the table, the attributes, and the time span. Click **Report**.

Selected	Column name	Column description
<input checked="" type="checkbox"/>	ORIGINNODE	System Name
<input checked="" type="checkbox"/>	TIMESTAMP	Timestamp
<input checked="" type="checkbox"/>	SCKPROTO	Socket Protocol
<input checked="" type="checkbox"/>	RECVQ	Receive Queue
<input checked="" type="checkbox"/>	SENDQ	Send Queue
<input checked="" type="checkbox"/>	LOCLADDR	Local Address
<input checked="" type="checkbox"/>	LOCLPORT	Local Port
<input checked="" type="checkbox"/>	LOCLSVC	Local Service
<input checked="" type="checkbox"/>	FORNADDR	Foreign Address
<input checked="" type="checkbox"/>	STATE	Socket State
<input checked="" type="checkbox"/>	SOCKUID	Socket UID
<input checked="" type="checkbox"/>	SCKINOD	Socket Inode
<input checked="" type="checkbox"/>	REMOTPORT	Foreign Port
<input checked="" type="checkbox"/>	RUSER	Socket Owner Name

**Note:** Check the local time on VM02 when you make the request. This step ensures that you get data.

11. Click **Back** in the browser.
12. Click any other links that you are interested in within the Agent Service interface.
13. Click **Back** in the browser.
14. Click the **Queries** link and explore the information provided.
15. Click **Back** in the browser.
16. Click **Service Interface Request**.
17. Type the string <SITSUMMARY> into the request area and click **Submit the Request**.
18. Examine the Agent Response Payload output.

19. Optional: Copy and paste the Agent Response Payload into a temporary file and view the output by using the Firefox browser.

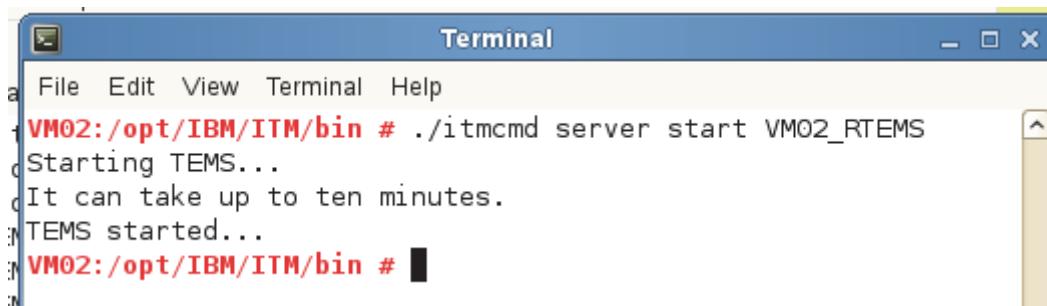


```

- <SITSUMMARY> ←
  <NUMBSIT>29</NUMBSIT>
  - <SITUATION>
    <NAME>Apache_Down_Linux</NAME>
    <SOURCE>E</SOURCE>
    <TYPE>Sampled</TYPE>
    <INTERVAL>30</INTERVAL>
    <ROWSIZE>1540</ROWSIZE>
    <FIRSTSTARTTIME>Mon Jul 15 13:25:13 2013</FIRSTSTARTTIME>
    <LASTSTARTTIME>Mon Jul 15 13:25:13 2013</LASTSTARTTIME>
    <LASTSTOPTIME>NA</LASTSTOPTIME>
    <FIRSTEVENTTIME>NA</FIRSTEVENTTIME>
    <LASTTRUETIME>NA</LASTTRUETIME>
    <LASTFALSETIME>Mon Jul 15 14:10:12 2013</LASTFALSETIME>
    <TIMESRECYCLED>0</TIMESRECYCLED>
    <TIMESAUTONOMOUS>0</TIMESAUTONOMOUS>
    <PREDICATE>Process_Command_Name = httpd2-prefork </PREDICATE>
  </SITUATION>
  - <SITUATION>

```

20. To prepare the systems for future exercises, restart the remote monitoring server on VM02.

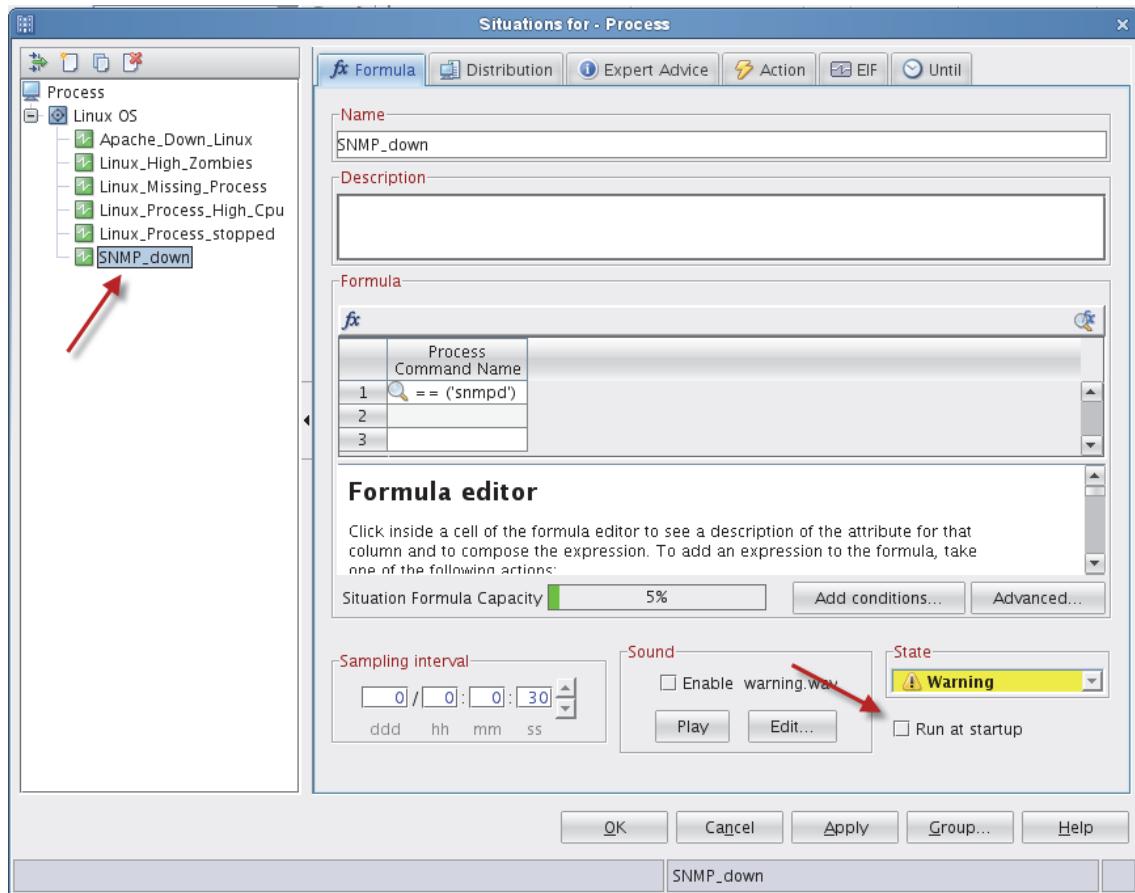


```

VM02:/opt/IBM/ITM/bin # ./itmcmd server start VM02_RTEMS
Starting TEMS...
It can take up to ten minutes.
TEMs started...
VM02:/opt/IBM/ITM/bin #

```

21. On VM01, stop the situation **SNMP\_down** and adjust the situation so that it does not automatically restart.





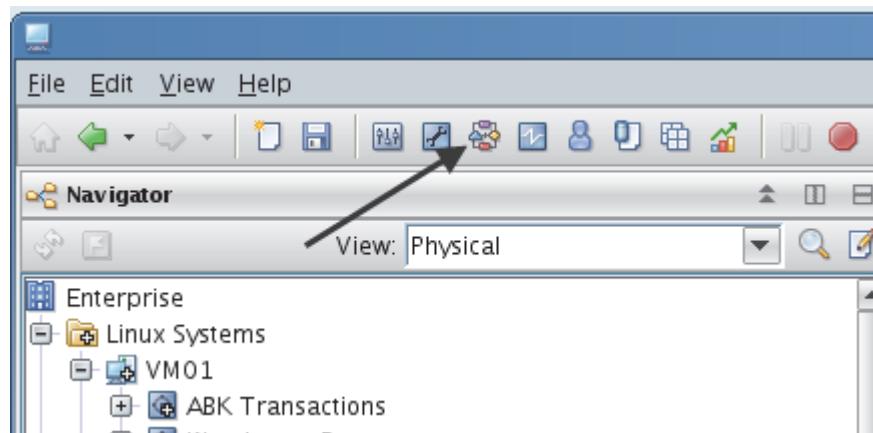
## 5 Working with policies exercises

When you use situation actions, no immediate feedback is available about the success or failure of the system command. The only indication of whether the command is successful is that the situation event is closed when the situation is re-evaluated. You typically need to either receive immediate notification after a problem is solved or issue additional commands. In the exercises for this unit, you issue commands to create a policy to use workflow automation.

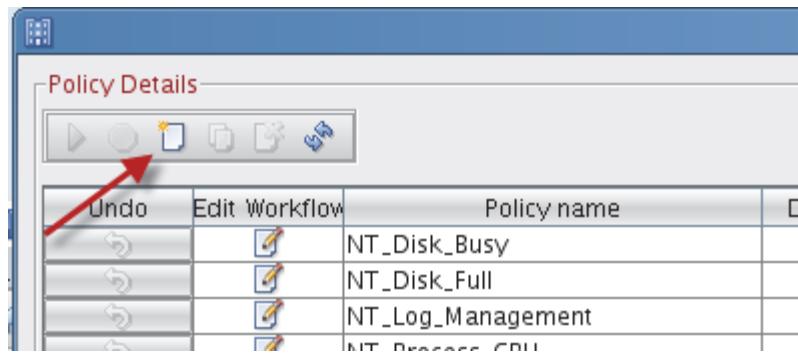
### Exercise 1. Creating a policy

Your policy example in this course is based on a missing process scenario. You began by solving problems from Take Actions. Then, you worked with situation actions. In this exercise, you use workflow automation.

1. From the portal, select **Enterprise > Linux Systems > VM01 > Linux OS**.
2. Right-click the **Process** Navigator and select **Situations**.
3. Check the **Action** tab for the Linux\_Missing\_Process situation. If an action exists, remove it.
4. Ensure that Linux\_Missing\_Process starts by clicking **Run at Startup**. Click **OK**.
5. To create your first policy, access the Workflow editor by pressing **Ctrl+W** on your keyboard. You can also click the **Workflow editor** icon in your portal toolbar.



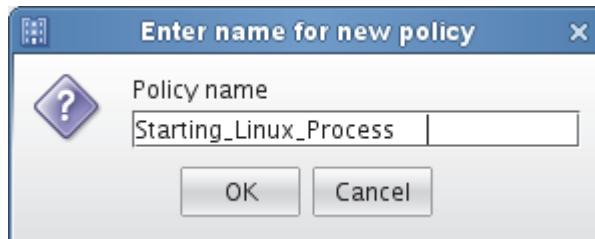
6. In the editor, select the **New Policy** icon to create a policy.



7. Enter **Starting\_Linux\_Process** as the name of the policy. Click **OK**.



**Important:** You cannot modify the policy name after the policy is committed to the hub monitoring server.

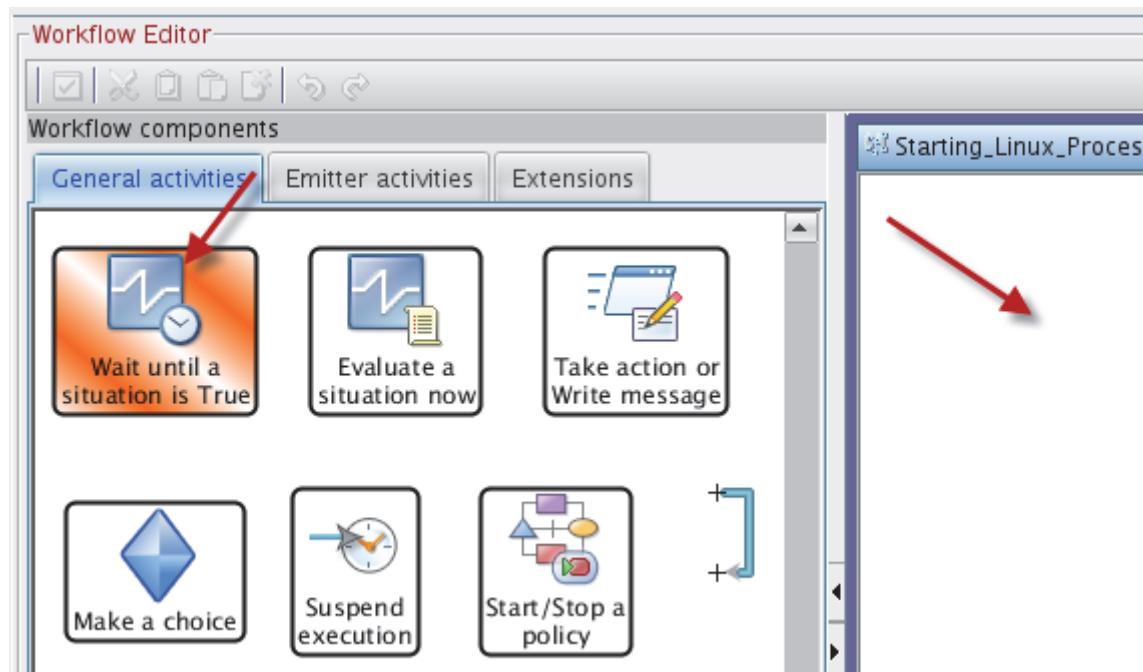


The Starting\_Linux\_Process policy is added to the Policy Details table.

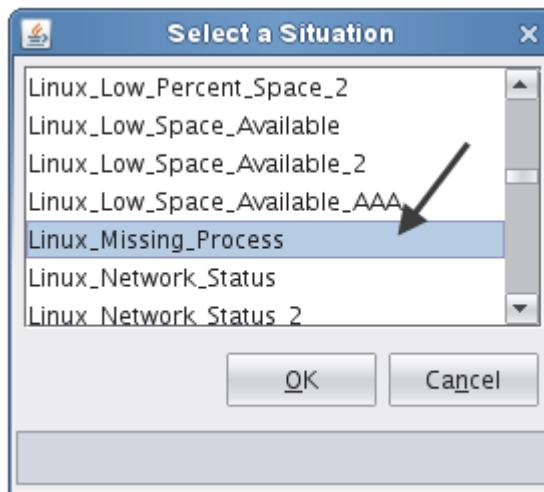


The bottom portion of your Workflow editor has two panes. In the left pane, you have several tabs with activities that you can include in your workflow. In the right pane, you have an empty Grapher view. You build your policy in this view. Most policies start with a situation. After that situation becomes true, the policy runs. The policy must run to solve the missing process issue. You place that situation into the workflow as the policy trigger situation.

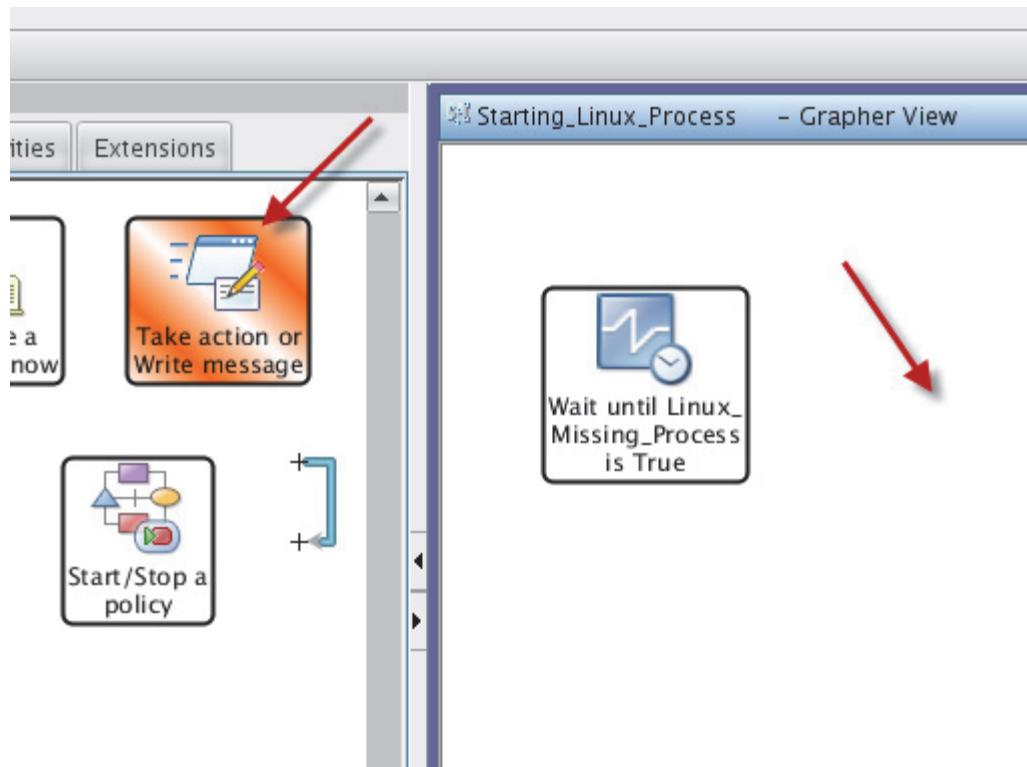
8. Select the **Wait until a situation is True** activity in the **General activities** tab. Click in the Grapher view to add that activity to the workflow.



9. From the displayed list of situations, select the **Linux\_Missing\_Process** situation to trigger your policy. Click **OK**.



10. Add a **Take action or Write message** activity to the workflow.



11. Enter the following command in the **System Command** field. Leave no spaces after the last forward slash (/).

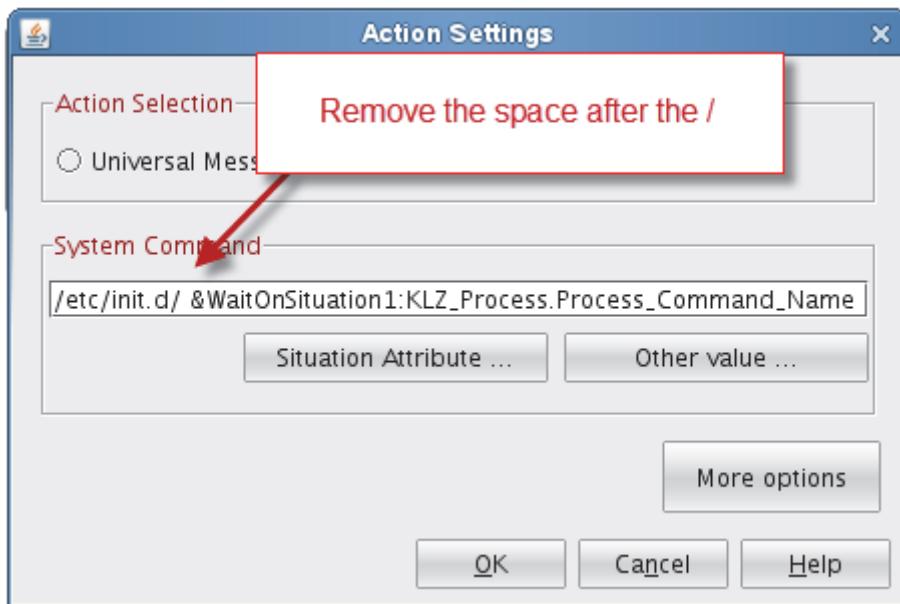
/etc/init.d/



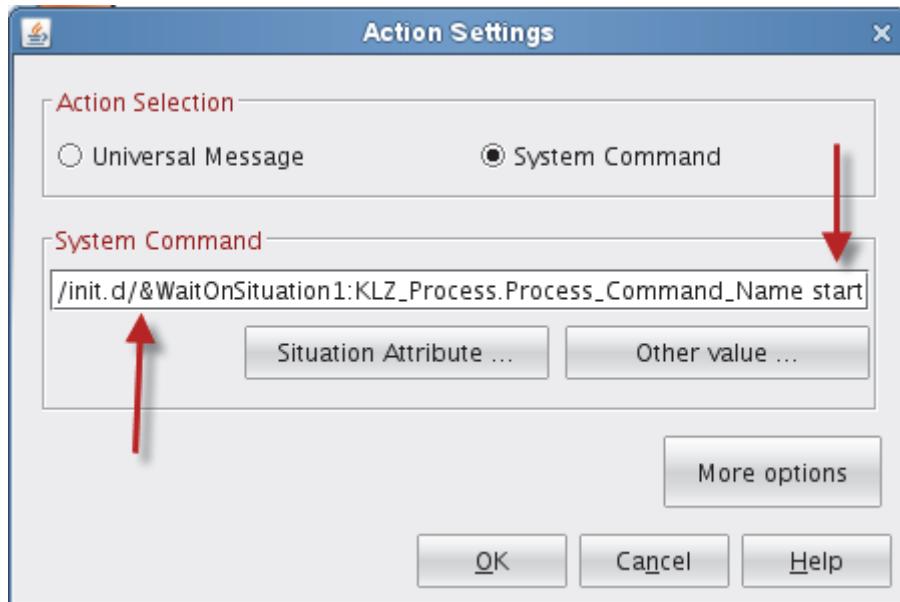
12. Click **Situation Attribute**. Select **Linux Process > Process Command Name** and click **OK**.

A space is automatically added before the process command name attribute.

13. Remove the space.

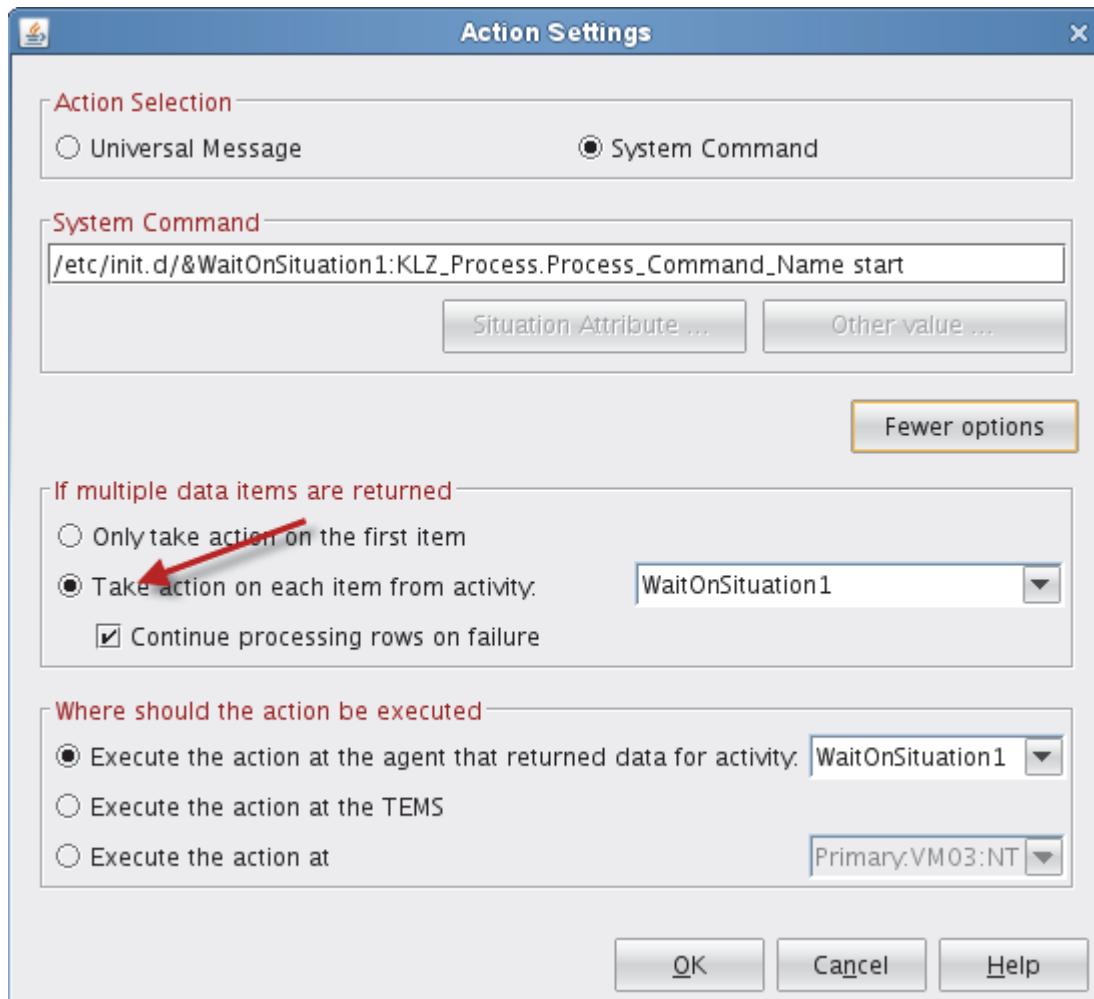


14. Add the word **start** to the end of the command.



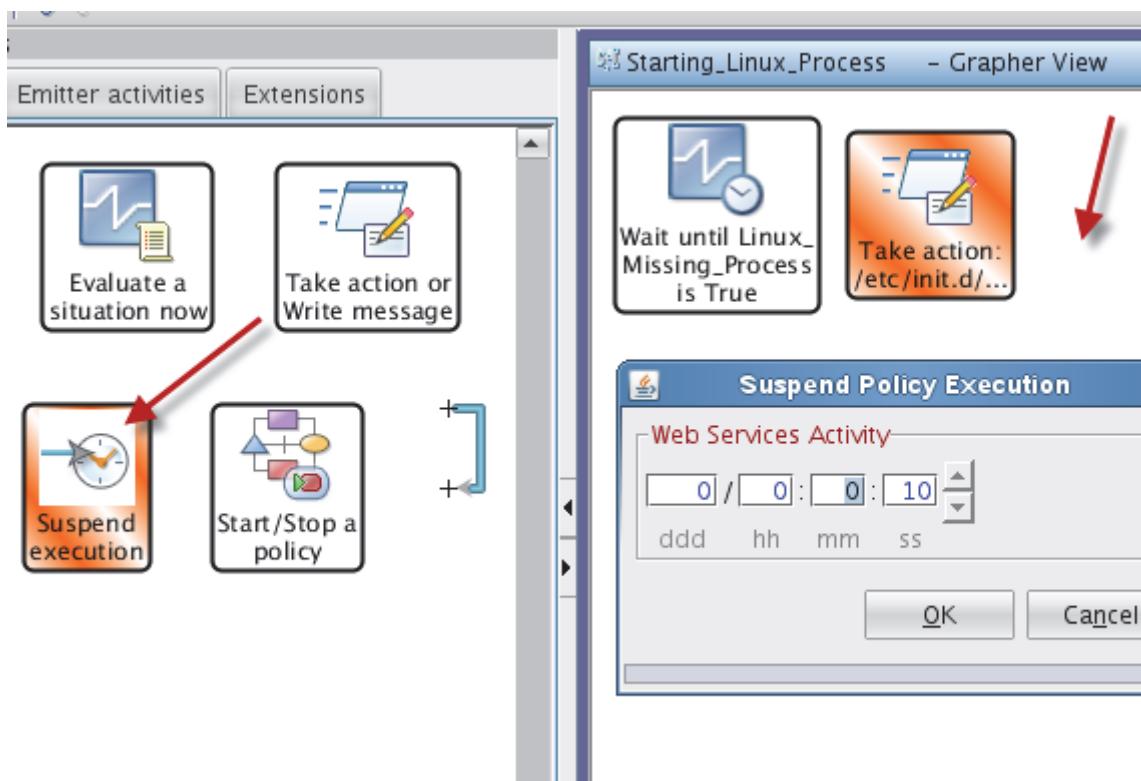
15. Click **More options**.

16. Click **Take action on each item from activity**. You have only one situation embedded in the workflow at this time, and only the **WaitOnSituation1** option is available.

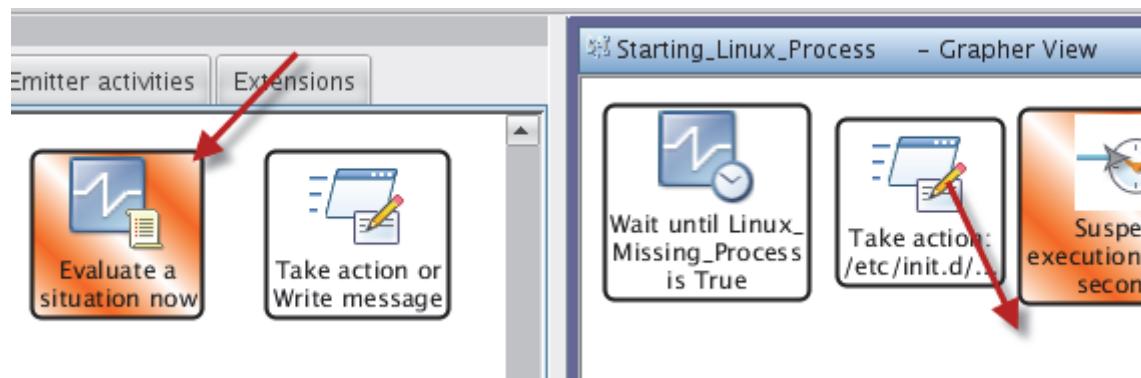


17. Click **OK** to save your action settings.

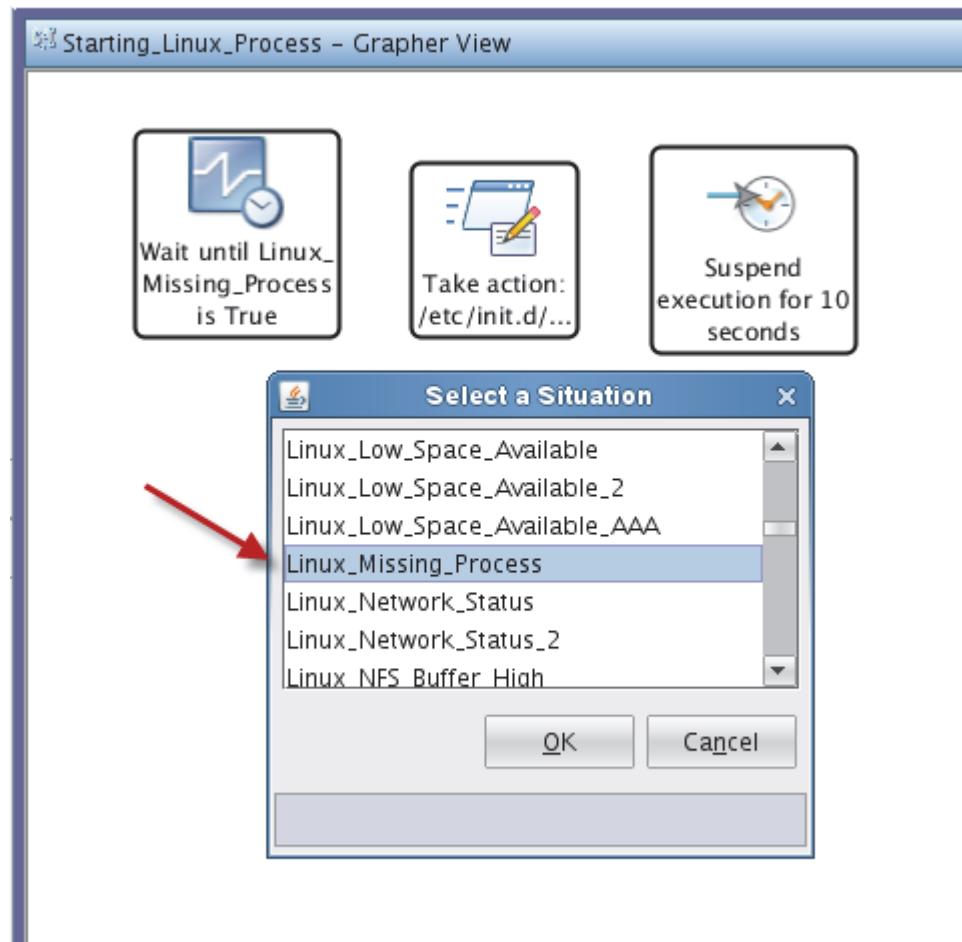
18. Add a **Suspend execution** activity to give the process time to start before evaluating the trigger situation again. Leave the default time at **10** seconds. Click **OK**.



19. Add the **Evaluate a Situation now** activity to immediately check the status of the situation expression. Select the activity and click inside the Grapher view.



20. When prompted, select the trigger situation **Linux\_Missing\_Process**. Click **OK**.



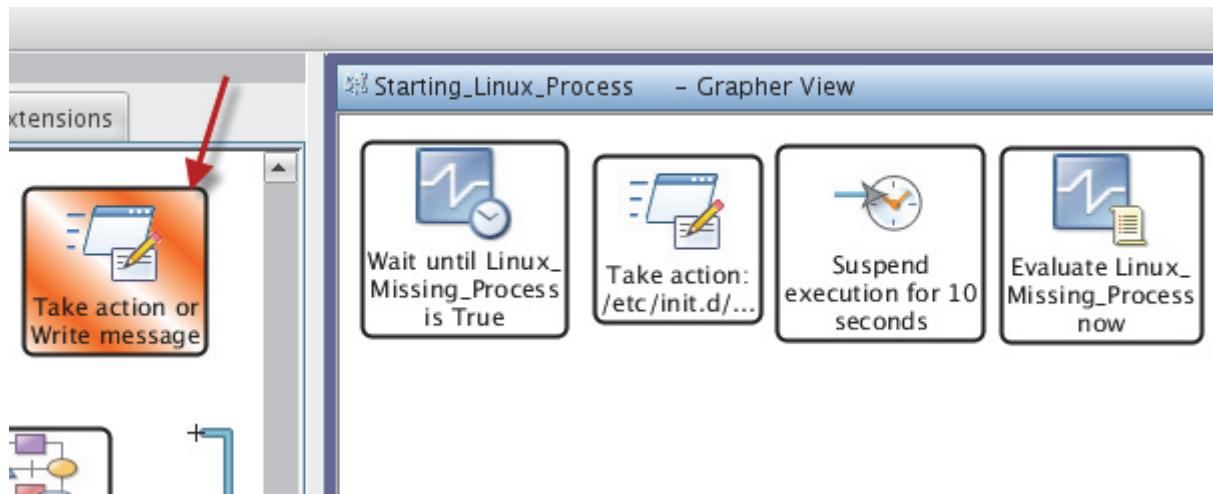
## Including a failure message in the workflow

When you validate the success or failure of automatically starting each process, you include system commands that send echo messages to terminal windows on your workstation. The messages describe the result of the situation evaluation.

21. Add a Take action or Write message activity to your workflow.

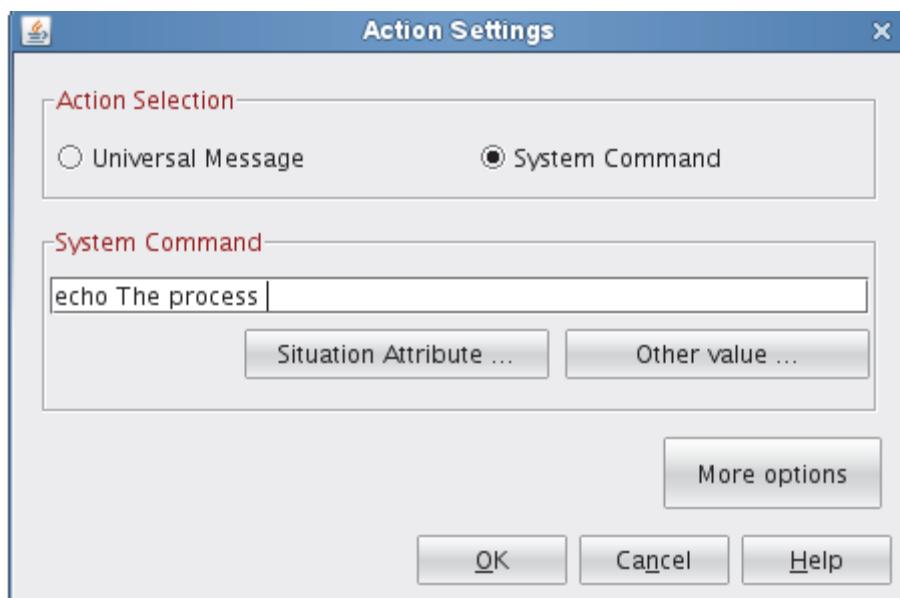
## Exercise 1. Creating a policy

If the problem persists, the process is not running and someone is notified. In the upcoming steps, you create an echo message. This message is written to all open terminal windows and indicates success or failure.



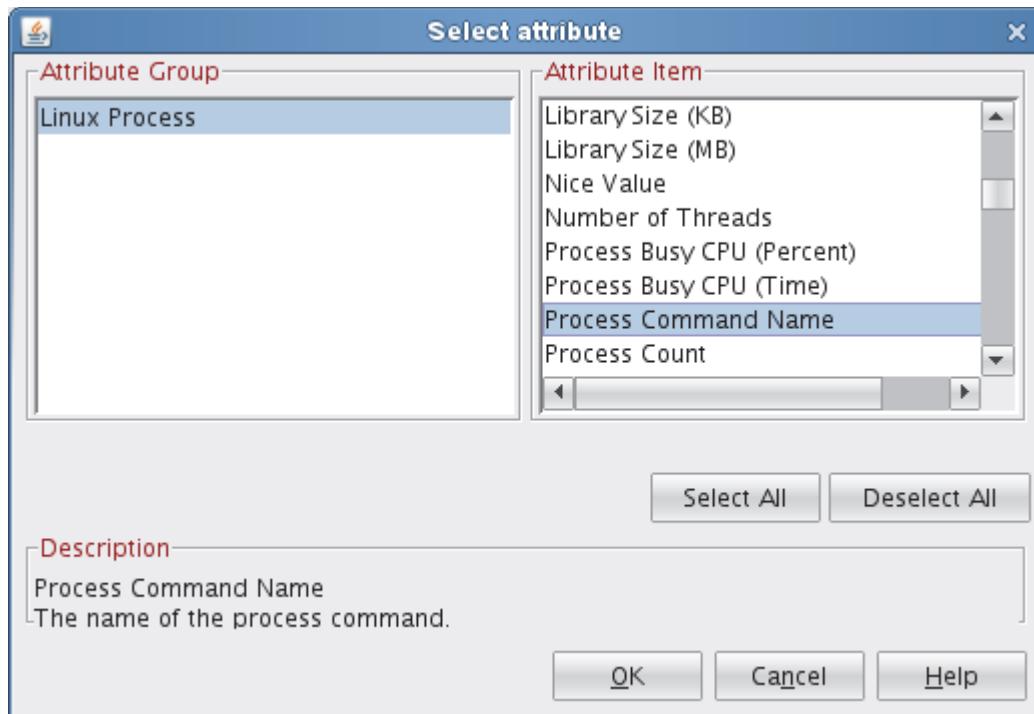
22. Enter the following command in the **System Command** field:

`echo The process`

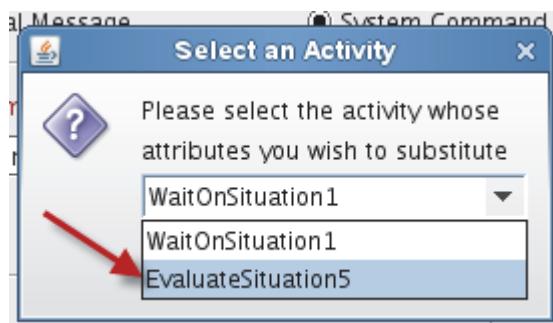


23. Click **Situation Attribute**.

24. Substitute the attribute **Process Command Name** in the command.

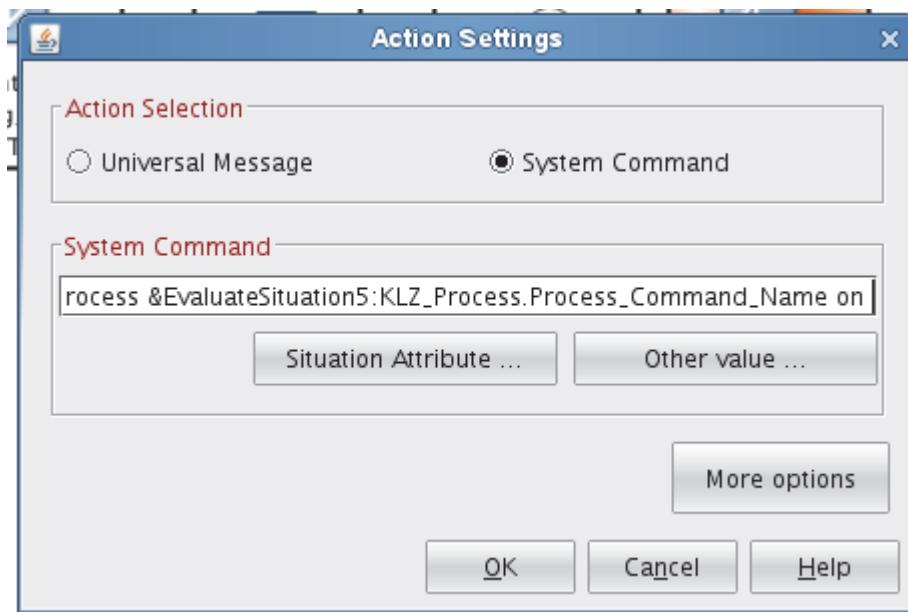


25. After confirming the attribute selection, ensure that you substitute the attribute from the situation results of the **EvaluateSituation5** situation. Select it from the pull-down list. Click **OK**.

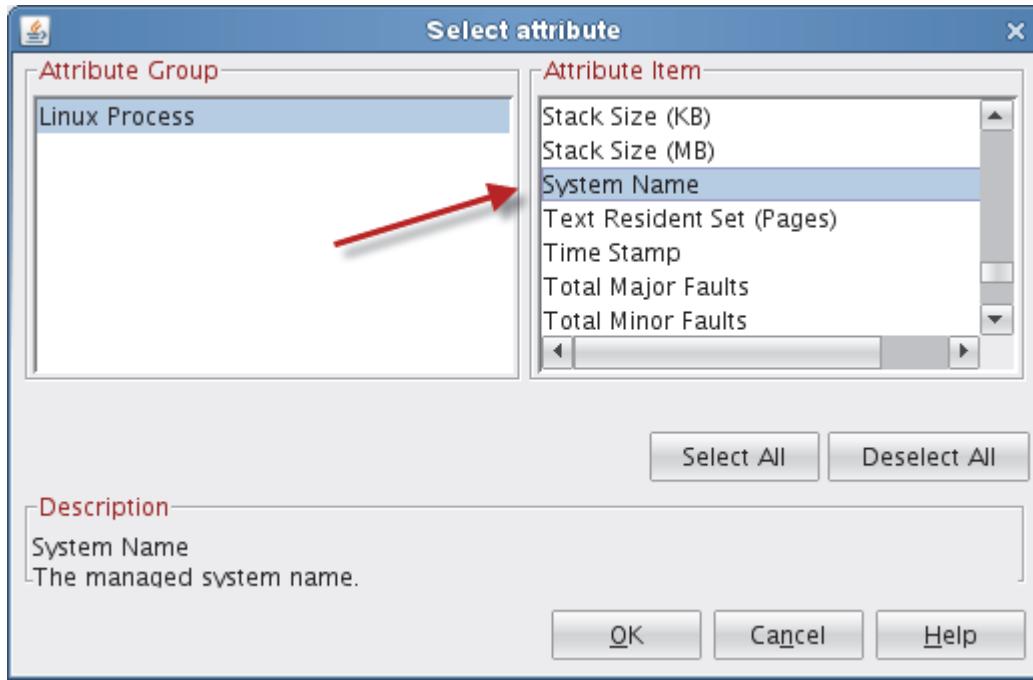


**Note:** Your situation names can be different. For instance, you might see EvaluateSituation4 instead of EvaluateSituation5.

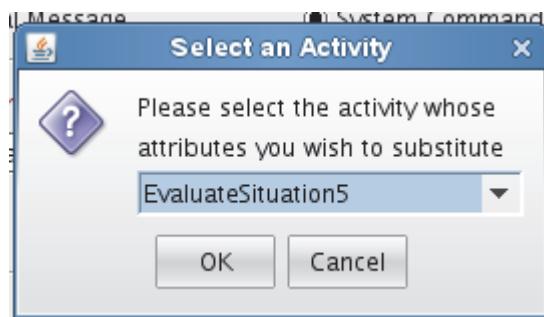
26. Continue the command by adding a space followed by the word **on**.



27. Click **Situation Attribute**. Select **System Name** as the attribute item and click **OK**.



28. Select the **EvaluateSituation5** activity (or whichever activity applies) and click **OK**.



29. Add the following words to complete the command:

was not running and could NOT be automatically started. Take additional action.  
| wall

The purpose of the command is to send a broadcast message to all open terminal windows.  
Your entire system command looks like the following example:

```
echo The process &EvaluateSituation5:KLZ_Process.Process_Command_Name on  
&EvaluateSituation5:KLZ_Process.System_Name was not running and could NOT be  
automatically started. Take additional action. | wall
```



30. Click **More options**. Ensure that **Only take action on the first item** is selected.

31. Click **OK** in the Action Settings window to save it.

## Including a success message in the workflow

Include a second Take action or Write message activity, which contains a message indicating that the problem was solved.

32. Create a new Take action or Write Message activity. Select the box and click in the Grapher view. Place this box *under* the previous **Take action or Write Message** box that you created.
33. Specify the system command by using the technique that you learned when creating the failure message:

```
echo The process &WaitOnSituation1:KLZ_Process.Process_Command_Name on
&WaitOnSituation1:KLZ_Process.System_Name was not running and could be
automatically started. | wall
```

Ensure that the attributes are substituted from the WaitOnSituation1 activity and not from the EvaluateSituation5 activity. You can see part of the command in the following example.



34. Click **More options**. Ensure that **Only take action on the first item** is selected.
35. Click **OK** in the Action Settings window to save it.

## Connecting the workflow

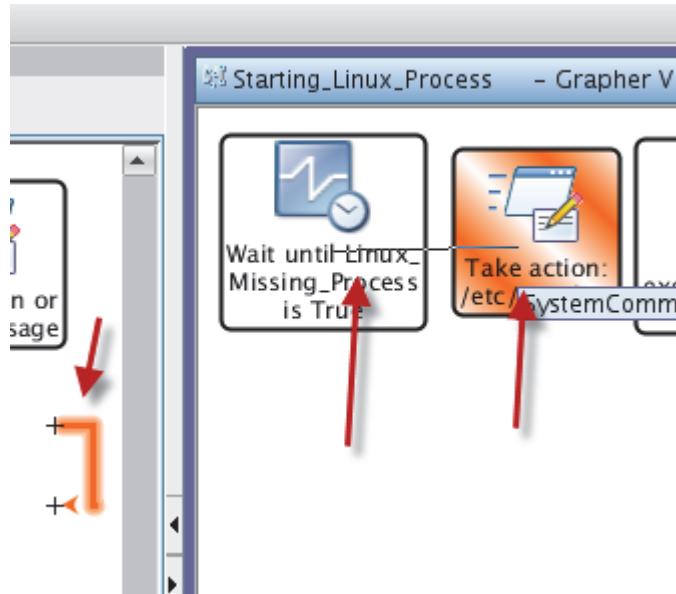
To define the workflow logic, use the connector tool. This tool is in the Workflow editor. To connect all activities, start with the Wait until Linux\_Missing\_Process is True activity. The connector tool looks like either of the following examples.





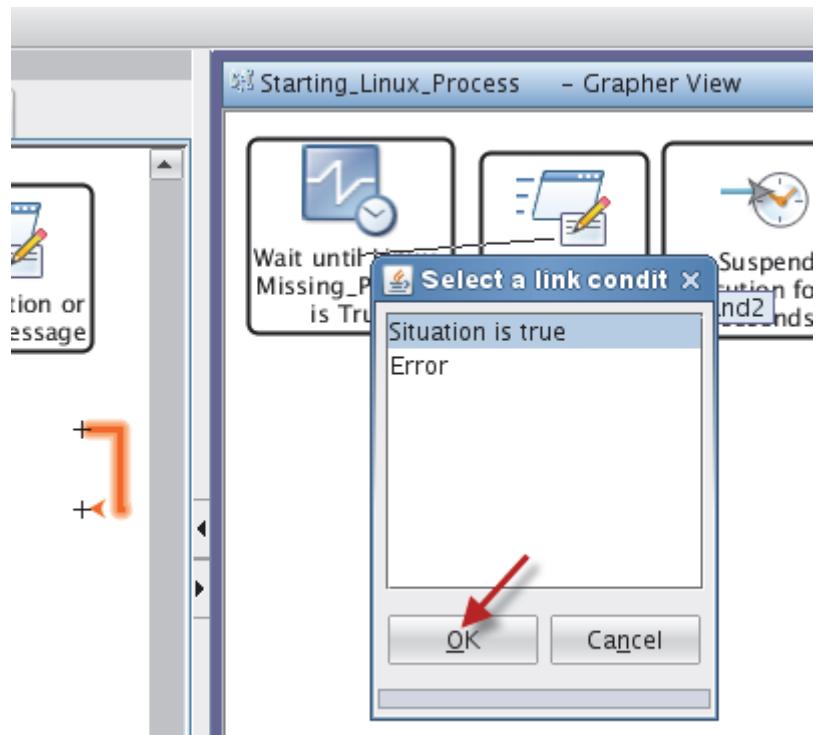
**Note:** You can double-click the connector tool to make it sticky. Then, you do not have to select it every time you want to create another connection. Ensure that you select both activities to establish the connection.

36. Double-click the connector to make it sticky. Click the **Wait until Linux\_Missing\_Process is True** activity and then click the **Take action** activity that restarts the process.

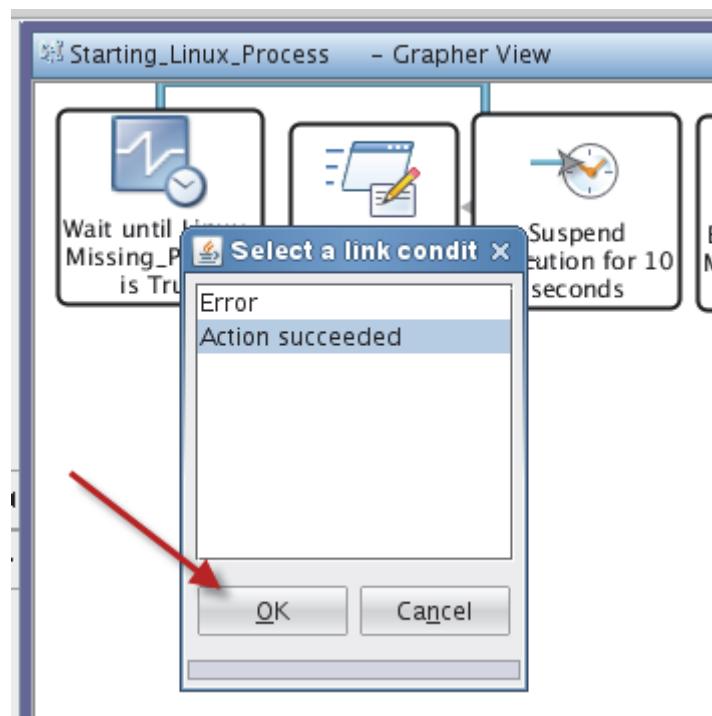


A Select a Link Condition window opens.

37. Select **Situation is true** and click **OK**.



38. Connect the **Take action** and **Suspend execution** activities. Select **Action succeeded** in the **Select a link condition** window. Click **OK**.

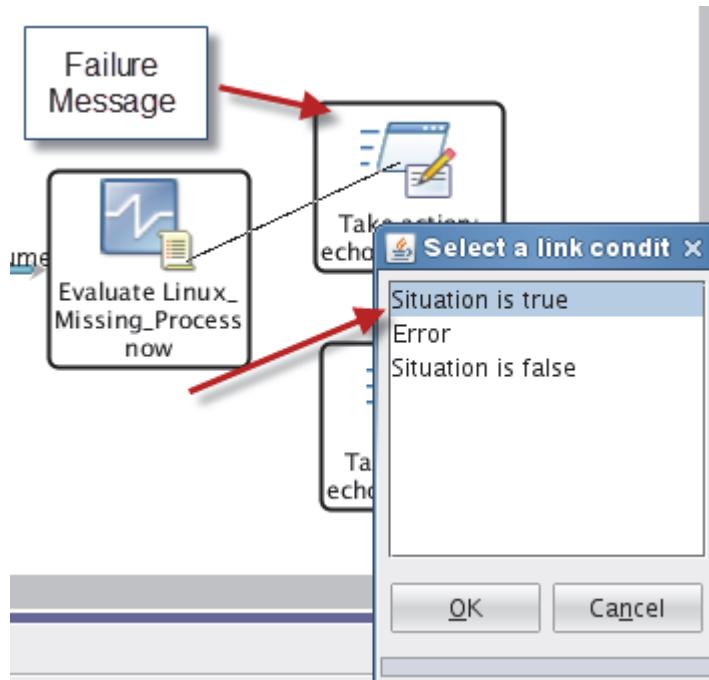


39. Connect the **Suspend Execution for 10 Seconds** activity to the **Evaluate Linux\_Missing\_Process now** activity.



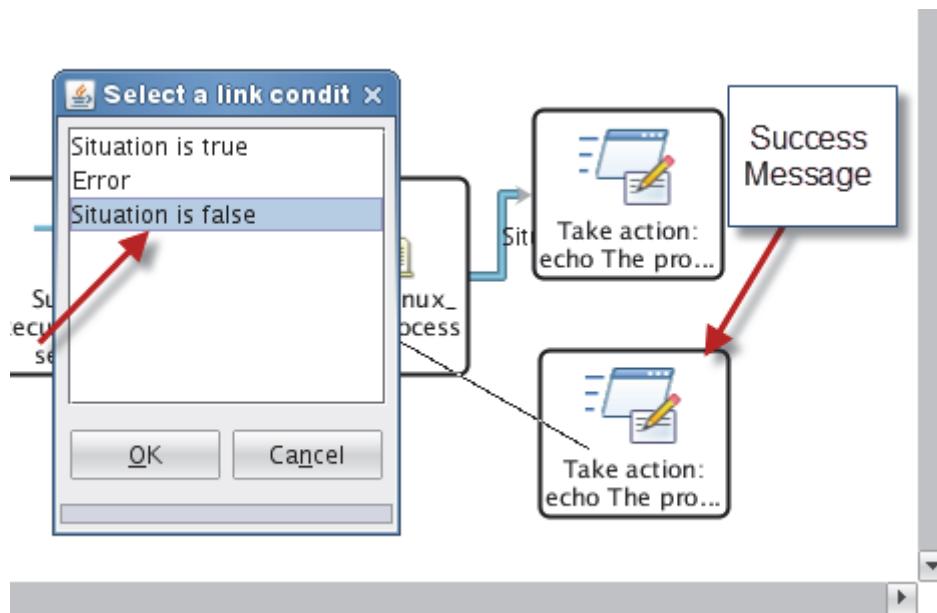
40. Connect the last two **Take action** activities to the **Evaluate Linux\_Missing\_Process now** activity, depending on the result of the evaluation. Use the following steps to make the connections. One connector goes to the upper **Take action** box, which is the first Take action that you created. This box contains the failure messages.

- a. Select **Situation is true** as the link condition. If the situation is true at this time, the attempt to restart the process did not occur within the time frame specified. The Linux process is missing. Click **OK**.

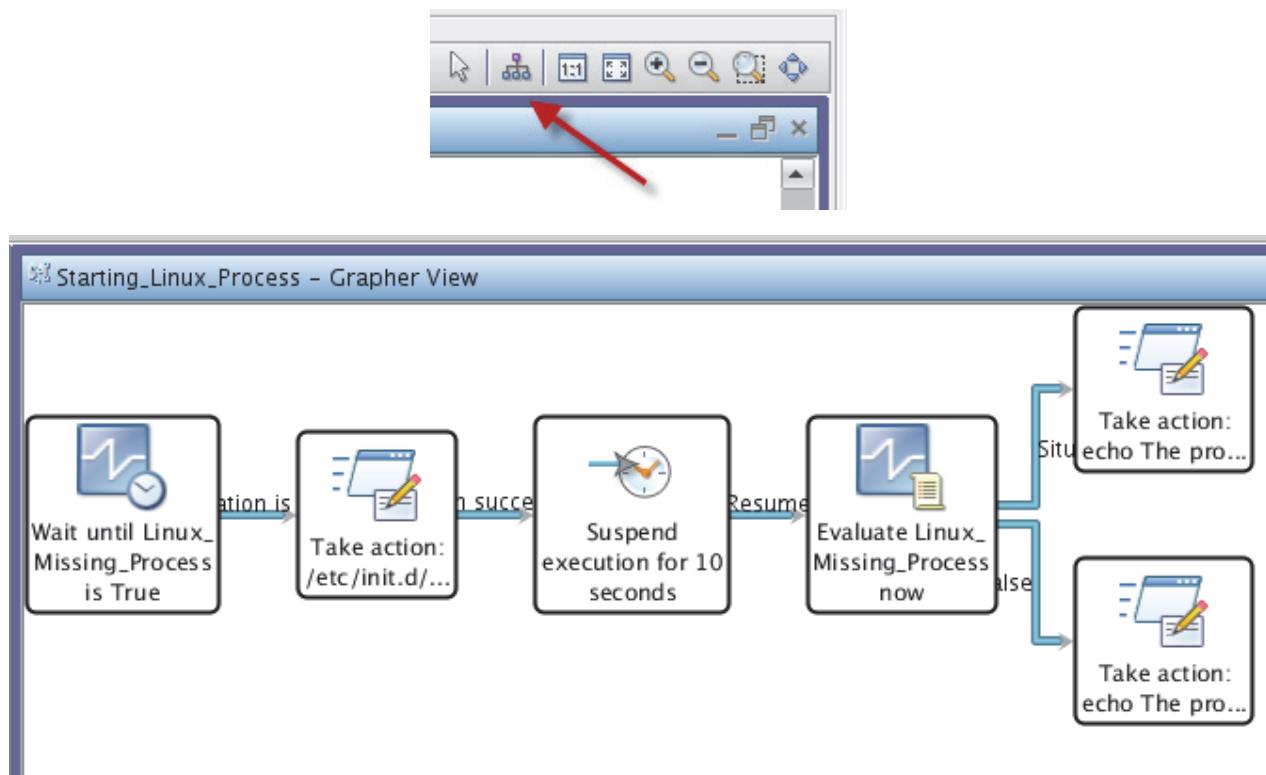


The connector, which goes to the lower or last Take action box that you created, contains the success message.

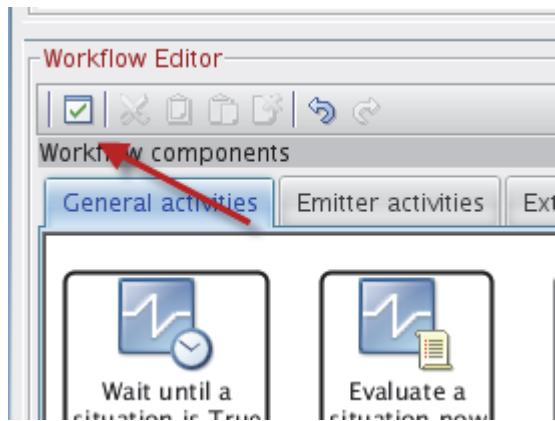
- b. Select **Situation is false** as the link condition. The situation is false at this time because the Linux process is started. Click **OK**.



Your workflow looks like the following example. You can click the **Perform Node Layout** icon to space everything evenly.



41. Validate your policy by using the Validate tool, which is in the upper-left corner of the Workflow editor.



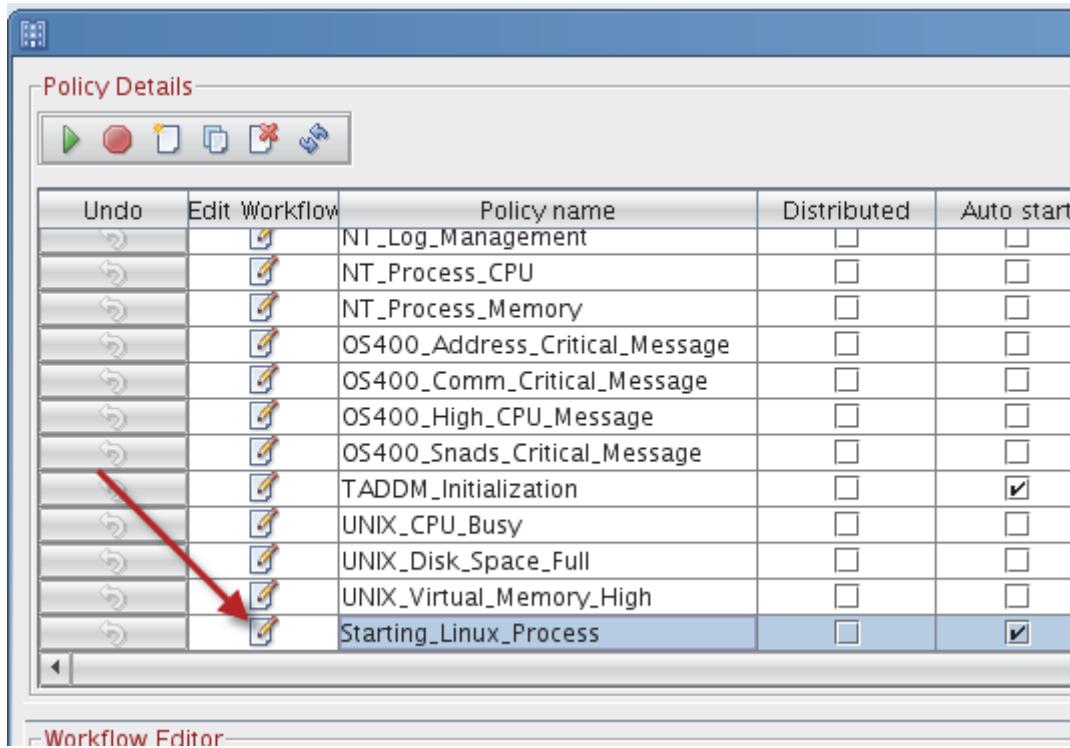
You can also validate automatically when you close and save your policy or when you click **Apply** to confirm the settings.



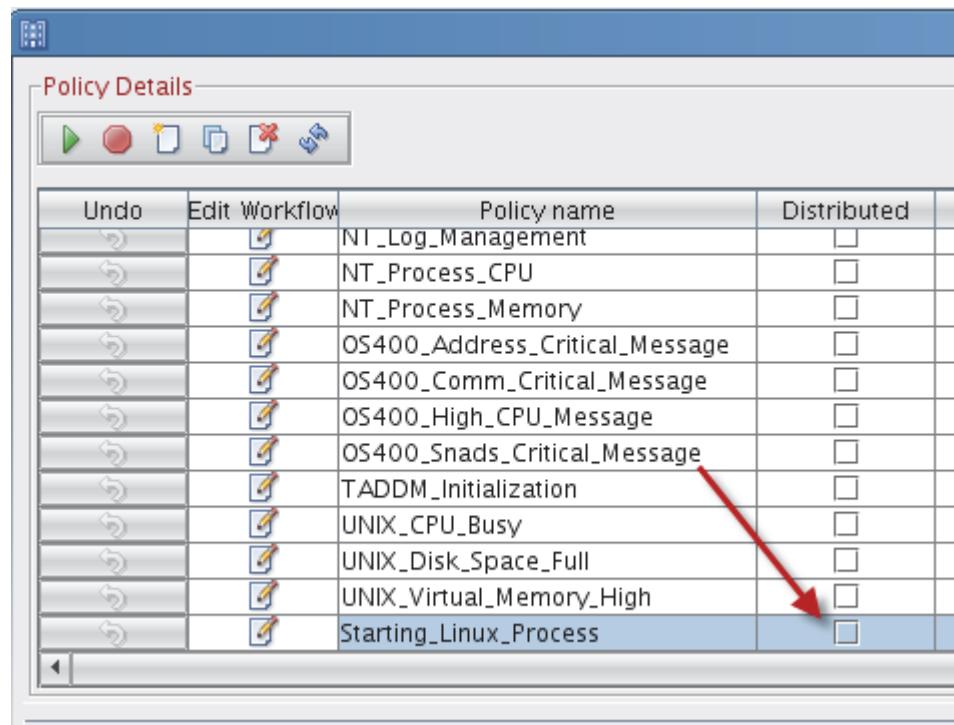
**Note:** This policy works for all processes that are monitored in the trigger situation. You can expand to other processes later.

## Exercise 2. Modifying policy settings

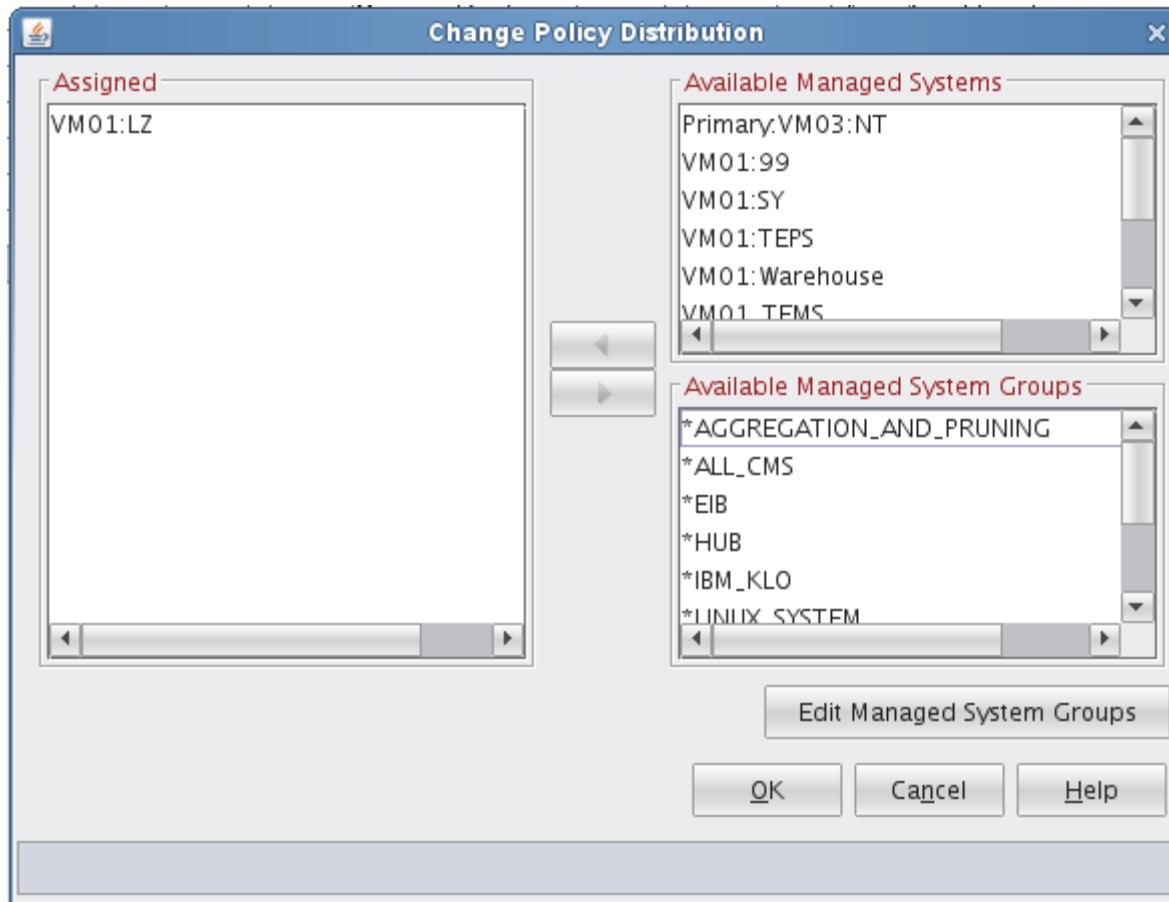
1. Access the Policy Details pane of the Workflows window. Select the **Starting\_Linux\_Process** policy.



2. Distribute the Starting\_Linux\_Process policy to the Linux OS agent on VM01. Select the check box in the Distributed column for **Starting\_Linux\_Process**.



3. Move the VM01:LZ agent to the Assigned pane. Confirm the setting and click OK.



4. Examine the Policy Details pane. After the policy is distributed, you see a check mark in the Distributed column. You can leave the policy correlation mode in the Correlate by column as **Host Name**, because you are running all activities against the same agent. Clear the **Auto Start** check box.

Undo	Edit Workflow	Policy name	Distributed	Auto start	Save results	Correlate by
undo	edit	UNIX_CPU_Busy	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Managed System
undo	edit	UNIX_Disk_Space_Full	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	Managed System
undo	edit	UNIX_Virtual_Memory_High	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Managed System
undo	edit	Starting_Linux_Process	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Host Name

5. Click **Apply** to confirm your settings.



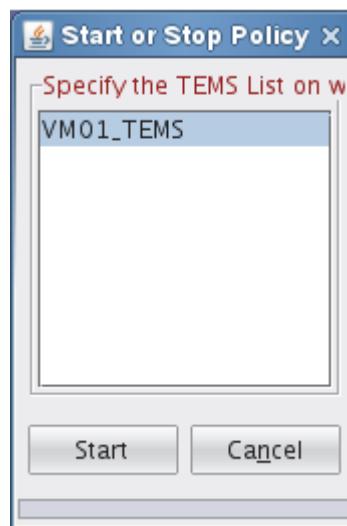
**Note:** If the **Undo** button is highlighted, you can use it to undo any changes that you make after you last confirmed the policy.

## Exercise 3. Starting the policies

1. To start your policy, click the **Start** button.



2. When prompted, select the monitoring server where you want the policy to be evaluated. For these lab exercises, select **VM01\_TEMS**. Click **Start**. If prompted to save the policy, click **OK**.



3. Open a terminal window. Test that your policy functions by first closing the SNMP daemon.

```
cd /etc/init.d  
. ./snmpd stop
```

```
a VM01:~/Desktop # cd /etc/init.d  
VM01:/etc/init.d # ./snmpd stop  
Shutting down snmpd:  
VM01:/etc/init.d #
```

## 5 Working with policies exercises

### Exercise 3. Starting the policies

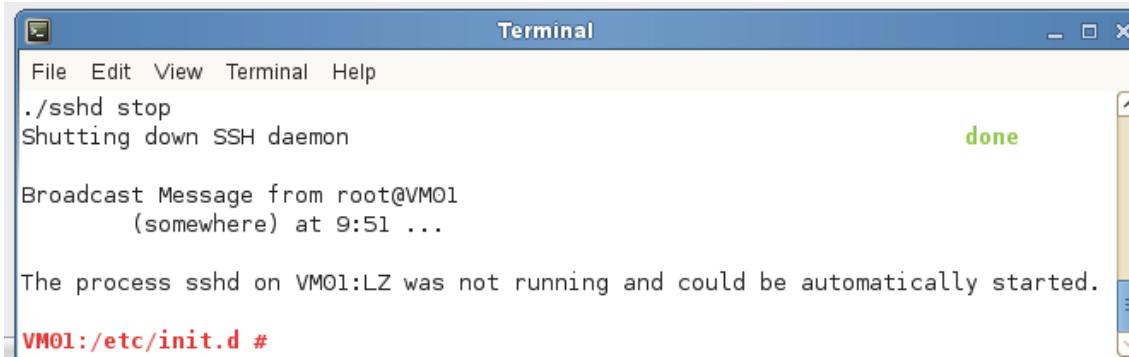
4. Wait for the Starting\_Linux\_Process policy to restart the SNMP daemon. Look for the message that indicates success or failure.

```
VM01:~/Desktop # cd /etc/init.d
VM01:/etc/init.d # ./snmpd stop
Shutting down snmpd:                                         done

Broadcast Message from root@VM01
(somewhere) at 9:42 ...

The process snmpd on VM01:LZ was not running and could be automatically started
.
```

5. Repeat the test by stopping the SSH daemon.



The screenshot shows a terminal window with a blue title bar labeled "Terminal". The window contains the following text:

```
File Edit View Terminal Help
./sshd stop
Shutting down SSH daemon                                         done

Broadcast Message from root@VM01
(somewhere) at 9:51 ...

The process sshd on VM01:LZ was not running and could be automatically started.

VM01:/etc/init.d #
```

6. On VM01, close the SSH daemon. When it restarts, close it again within 10 seconds. Look for the broadcast message that the SSH process cannot be automatically restarted. The policy is set to restart. It checks sshd again in the next interval and successfully restarts it.

The screenshot shows a terminal window with the following content:

```
VM01:/etc/init.d # ./sshd stop
Shutting down SSH daemon
VM01:/etc/init.d # ./sshd status
Checking for service sshd
VM01:/etc/init.d # ./sshd status
Checking for service sshd
VM01:/etc/init.d # ./sshd stop
Shutting down SSH daemon
```

On the right side of the terminal window, there are four status indicators:

- done (next to the first shutdown command)
- unused (next to the first status check)
- running (next to the second status check)
- done (next to the second shutdown command)

Below the terminal output, there are two broadcast messages:

Broadcast Message from root@VM01  
(somewhere) at 11:30 ...

The process sshd on VM01:LZ was not running and could NOT b  
e automatically star  
ted. Take additional action.

Broadcast Message from root@VM01  
(somewhere) at 11:31 ...

The process sshd on VM01:LZ was not running and could be au  
tomatically started.

The policy is written so that if the failure to start the ssh daemon happens multiple times, multiple failure messages are generated. In the next exercise, you make the policy wait for a condition to become false. *False* in this exercise means that the process is not missing. *True* means that process is missing.

## Exercise 4. Atomizing policies

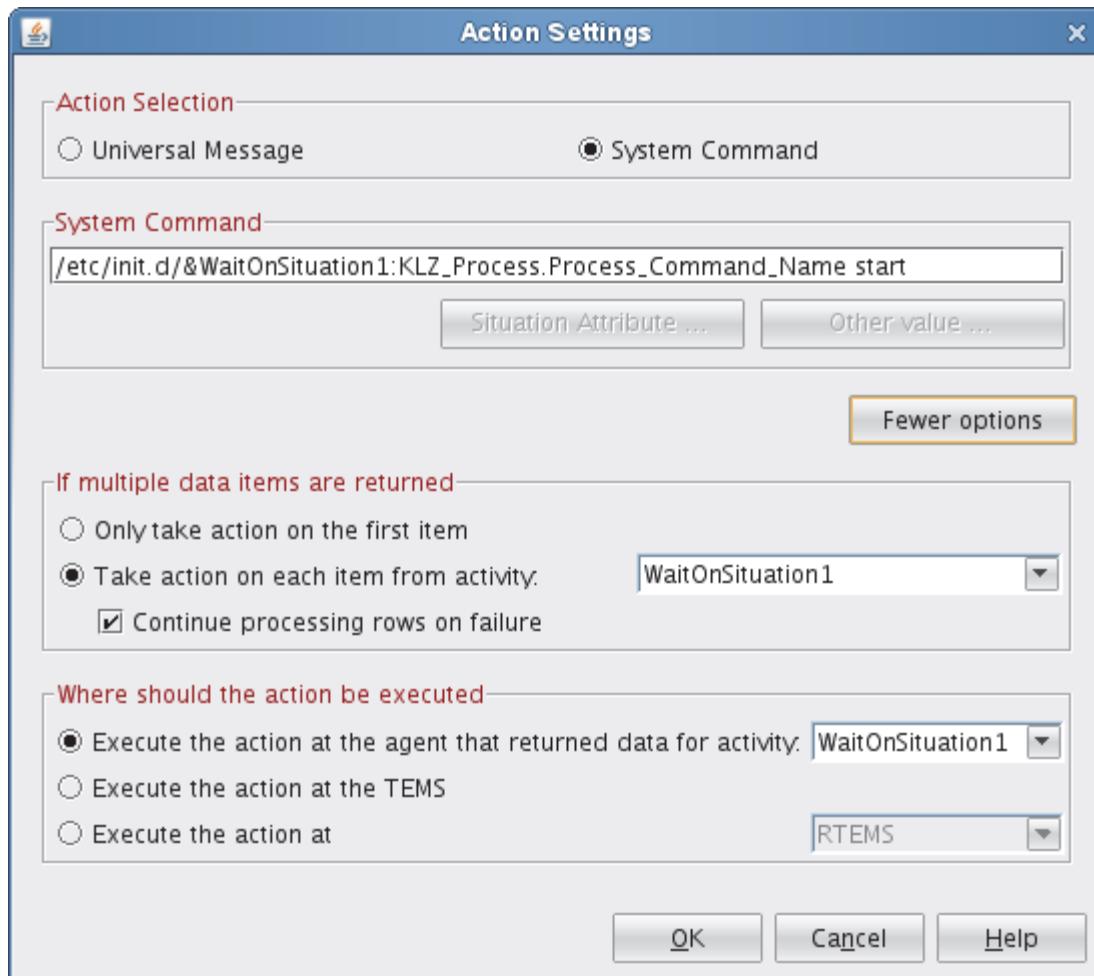
A policy starts multiple execution threads if you atomize the situation. A policy uses a display item to trigger a separate situation event for every process that is not running. Policies can run a separate thread of operation for every situation event. You can include a Take action or Write message activity in the workflow. The policy continues its operation as a separate thread for every process that is not running.

One issue with this policy is that it can continue trying to restart any process that it cannot start automatically. In other words, the policy can loop. If your last activity is a message or an email, the activity is continuously sent to the recipient, which is a problem. To prevent this problem, you can include a Wait until a situation is False activity. You place this activity behind the system command that sends a message that the restart failed.

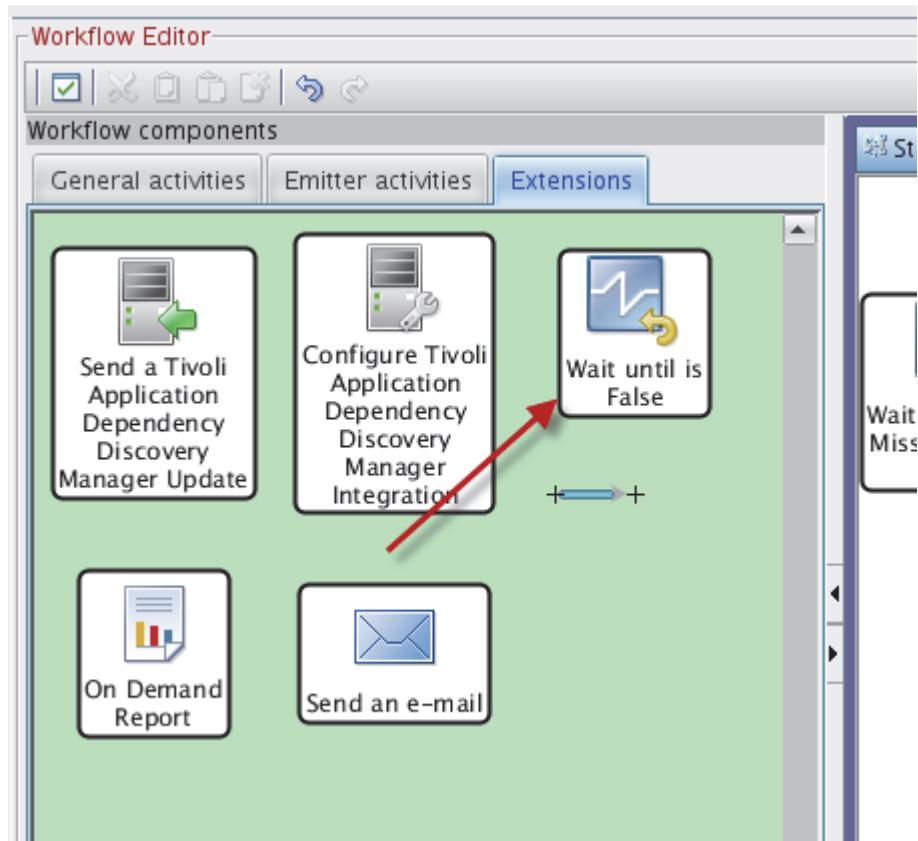
1. Open the Workflow editor for the **Starting\_Linux\_Process** policy by selecting the icon in the Edit Workflow column of the Policy Details pane.
2. Double-click the first Take action. Select **More options** to expand the **Action Settings**. Review the action settings.

The actions indicate **Take action on each item from activity**. Multiple threads are created for this policy, one for each process.

3. After you review this activity, click **Cancel** to exit.

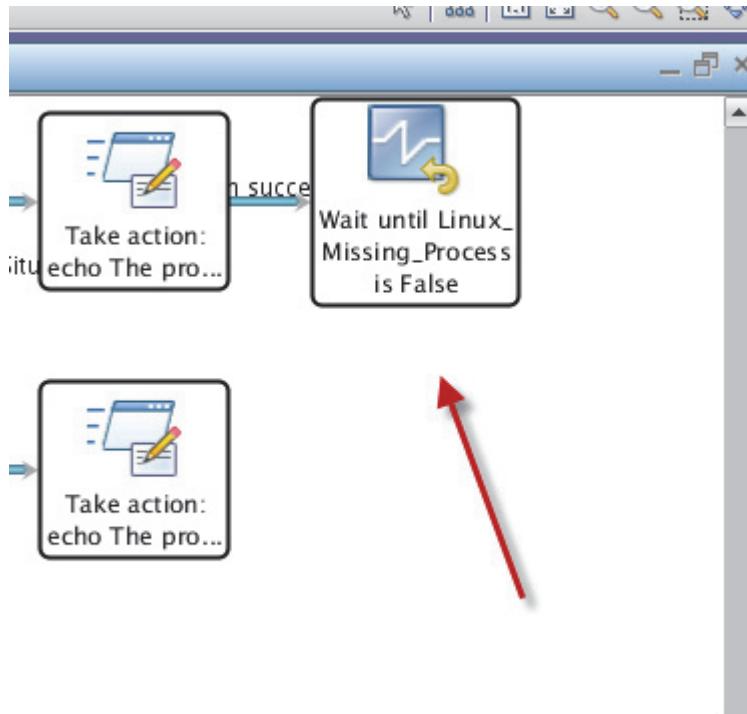


4. Click the **Extensions** tab under the Workflow Components pane in the Workflow editor and add a **Wait until is False** activity. Select **Linux\_Missing\_Process** when prompted.



5. Place the new activity after the upper right true Take action, which emits the **process could NOT be started** message.

6. Add a connector between the Take action and the new activity. Select **Action succeeded** for the link condition and click **OK**. Save the policy. Stop and start the policy.



7. Test the policy by closing the ssh daemon. Within 10 seconds after the process is automatically started, close the ssh daemon again.

To determine whether the ssh daemon is automatically started by the policy, you repeatedly issue the **sshd status** command. An example is shown in the following screen capture. If you close the ssh daemon after it starts, you can simulate that the ssh daemon was not

automatically started and had a problem. Look for the message that the sshd cannot be automatically restarted. This time, the process stays closed until it is restarted by other means.

The screenshot shows a terminal window with the following text:

```
VM01:/etc/init.d # ./sshd stop
Shutting down SSH daemon
VM01:/etc/init.d # ./sshd status
Checking for service sshd
VM01:/etc/init.d # ./sshd status
Checking for service sshd
VM01:/etc/init.d # ./sshd status
Checking for service sshd
VM01:/etc/init.d # ./sshd stop
Shutting down SSH daemon

Broadcast Message from root@VM01

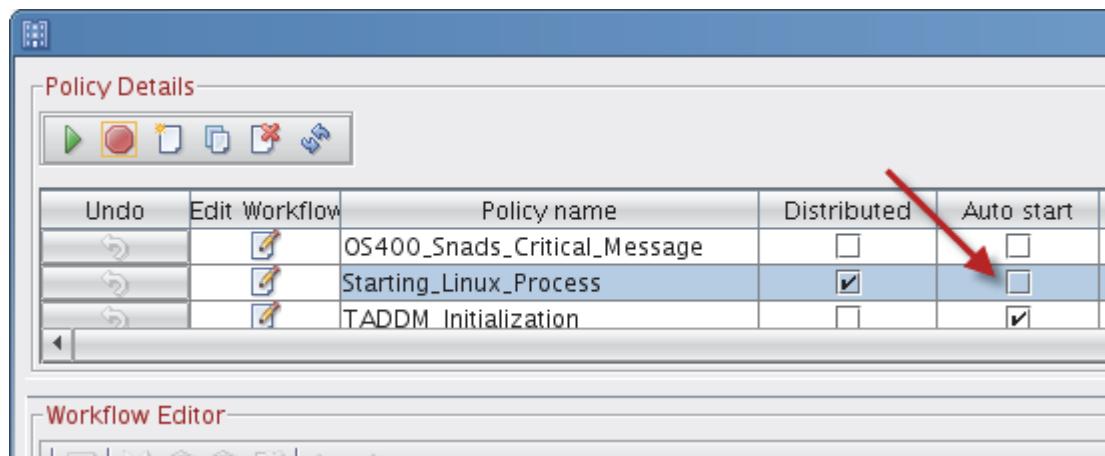
(somewhere) at 11:38 ...

The process sshd on VM01:LZ was not running and could NOT b
e automatically star
ted. Take additional action.
```



**Note:** The policy never tries to restart the process again until you manually start the application and the trigger situation is evaluated as false. Keep the application open for at least 30 seconds. After you stop the process again, the policy restarts it.

8. Stop the policy **Starting\_Linux\_Process** and ensure that the **Auto start** feature on the policy is cleared so that future exercises function correctly.





## 6 Agentless monitoring exercises

Using agentless monitoring, you can monitor an IT environment from a small set of centralized servers. This type of monitoring is useful for the following reasons:

- Limiting the amount of software installed on managed servers reduces administration, maintenance, and roll-out costs.
- Lightweight OS monitoring targets key metrics, and basic situations satisfy many monitoring needs.
- Standard protocols are used to gather metrics.
- Traditional enterprise agents are available for core business services.

*Agent-based* technology is on the system being monitored. It collects data that is based on local policy or by the management server. *Agentless* technology is primarily on the management server and receives its data by using a remote programming interface. *Agentless* does not mean that nothing is running on the monitored system. Some basic operating system function or base application function is running to provide the data as requested over the network. Services must be running on the managed system, and resources are being used. Examples include SNMP, JMX, CIM, and WMI.

Agentless monitoring provides the following key operating system metrics:

- Logical and physical disk use
- Network use
- Virtual and physical memory
- System-level information
- Individual and aggregate processor use
- Process availability

In the exercises for this unit, you learn how to use *agentless* monitoring. You also learn to configure IBM Tivoli Monitoring to monitor systems that do not have an IBM Tivoli Monitoring agent, and explain the agentless monitoring features and functions.

# Exercise 1. Agentless monitoring overview

In this exercise, you configure agentless monitoring on VM01 to gather information about VM02, which has a Linux OS agent installed on it. To compare the information that is collected, you use the agentless monitoring functions and the Linux OS agent.

1. Open a terminal session on VM02 and edit the `/etc/snmp/snmpd.conf` file by changing `rocommunity secret 127.0.0.1` to this line:

```
rocommunity tivoli
```



A screenshot of a gedit window titled '\*snmpd.conf (/etc)'. The file contains the following configuration:

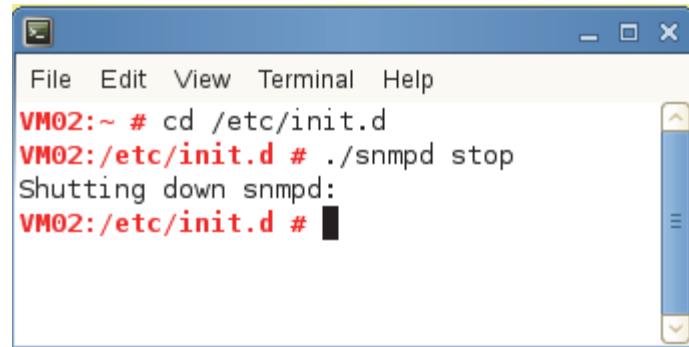
```
# Please see /usr/share/doc/packages/net-snmp/EXA
# more complete example and snmpd.conf(5).
#
# Writing is disabled by default for security rea
# to enable it uncomment the rwcommunity line and
# name to something nominally secure (keeping in
# transmitted in clear text).

# don't use '< >' in strings for syslocation or s
# Note that if you define the following here you '
# them with snmpset
syslocation Server Room
syscontact Sysadmin (root@localhost)

# These really aren't meant for production use.
# and can use considerable resources. See snmpd.
# on setting up groups and limiting MIBS.
rocommunity tivoli
# rwcommunity mysecret 127.0.0.1
```

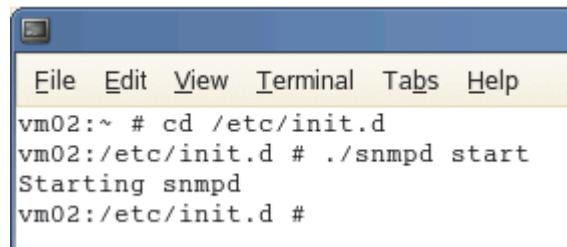
A red arrow points to the line 'rocommunity tivoli'.

2. Stop the SNMP daemon on VM02.



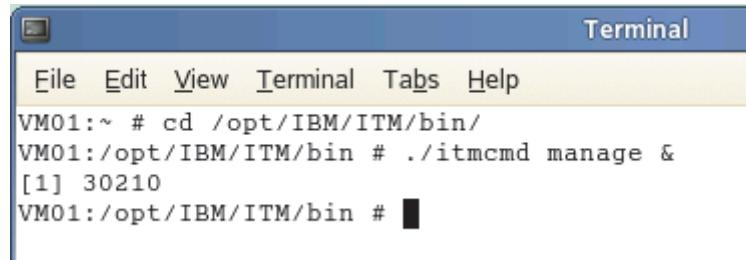
```
File Edit View Terminal Help
VM02:~ # cd /etc/init.d
VM02:/etc/init.d # ./snmpd stop
Shutting down snmpd:
VM02:/etc/init.d #
```

3. Restart the SNMP daemon on VM02.



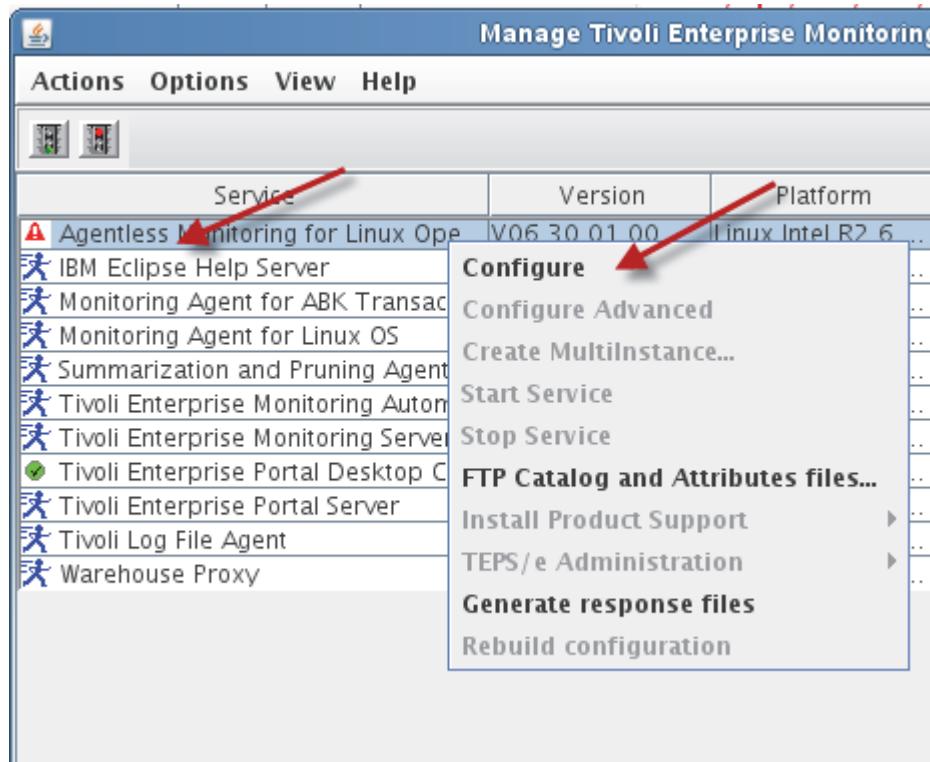
```
File Edit View Terminal Tabs Help
vm02:~ # cd /etc/init.d
vm02:/etc/init.d # ./snmpd start
Starting snmpd
vm02:/etc/init.d #
```

4. On VM01, open a terminal session and start the Manage Tivoli Enterprise Monitoring Services application.



```
File Edit View Terminal Tabs Help
Terminal
VM01:~ # cd /opt/IBM/ITM/bin/
VM01:/opt/IBM/ITM/bin # ./itmcmd manage &
[1] 30210
VM01:/opt/IBM/ITM/bin #
```

5. Right-click **Agentless Monitoring for Linux Operating Systems** and select **Configure**.



6. Click **Add Instance**.  
7. Enter **LinuxWebServers** as the instance name and click **OK**.



8. Click **Next** to accept the default settings.

- Set **tivoli** as the Community Name and enter it again in the Confirm Community Name field, and click **Next**. The community name matches the update that you made to the **snmpd.conf** file on VM02.

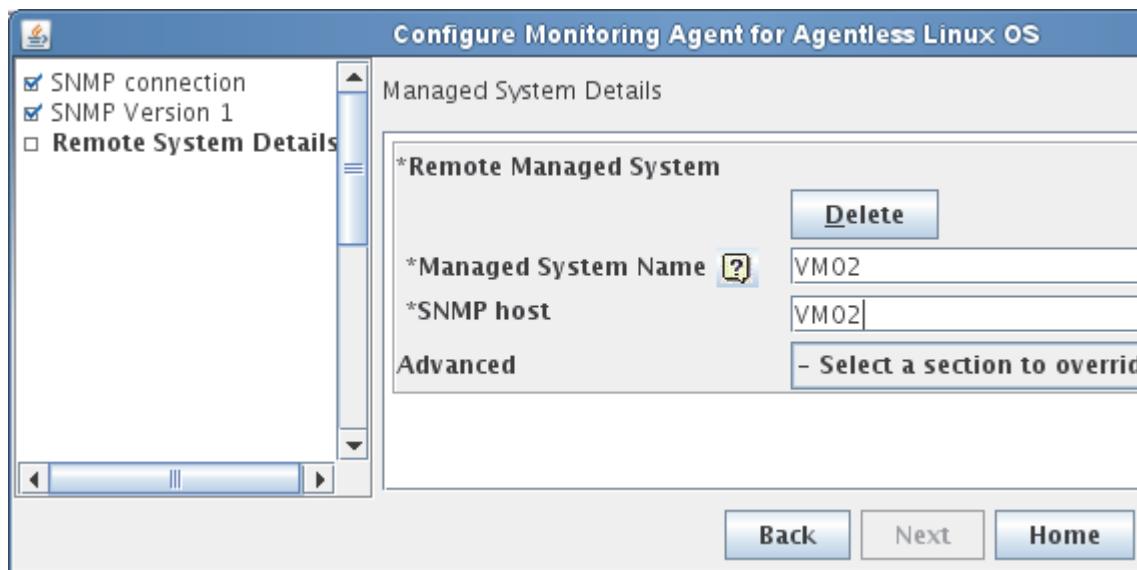


Your results should resemble the following screen capture.

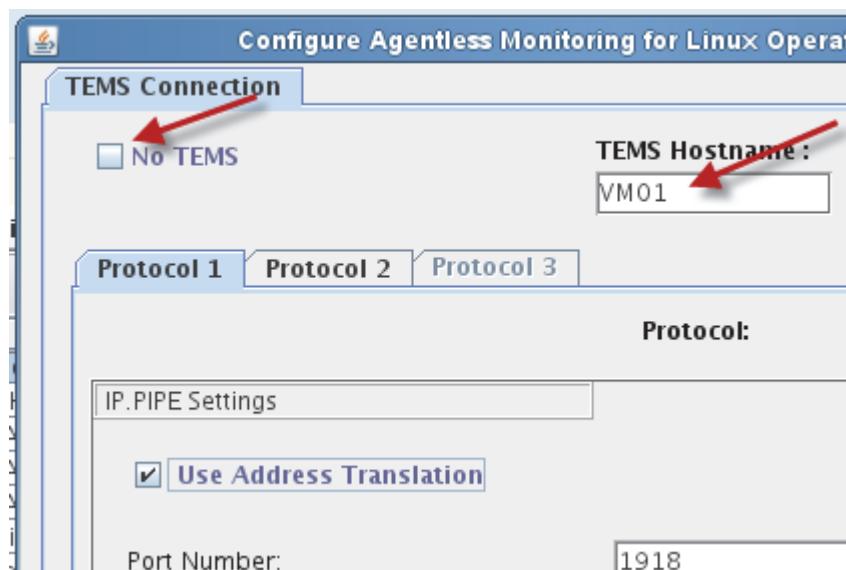


- Click **New**.

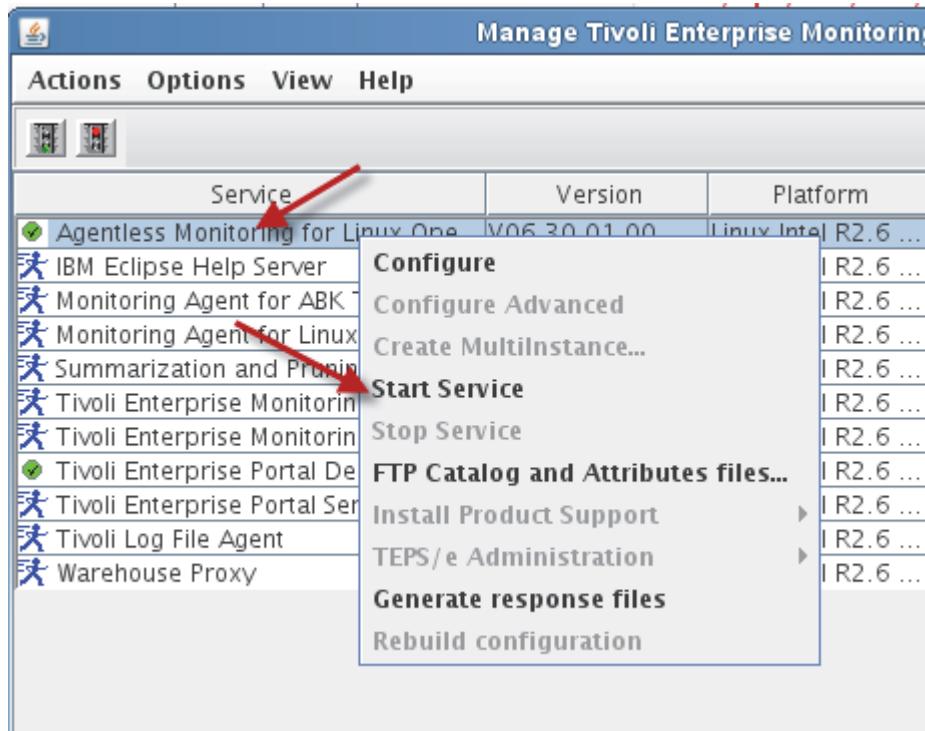
11. Provide the value **VM02** for both the Managed System name and the SNMP host name, and click **OK**.



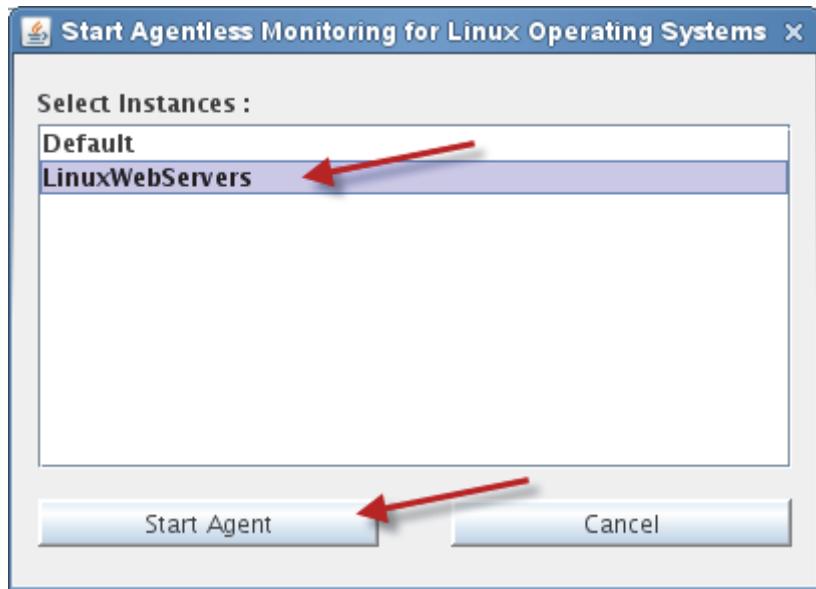
12. Clear the **No TEMS** check box and provide the Tivoli Enterprise Monitoring Server name of **VM01** in the **TEMs Hostname** field. Click **Save**.



13. Using the Manage Tivoli Enterprise Monitoring Services application, start the Agentless Monitoring for Linux Operating Systems service.

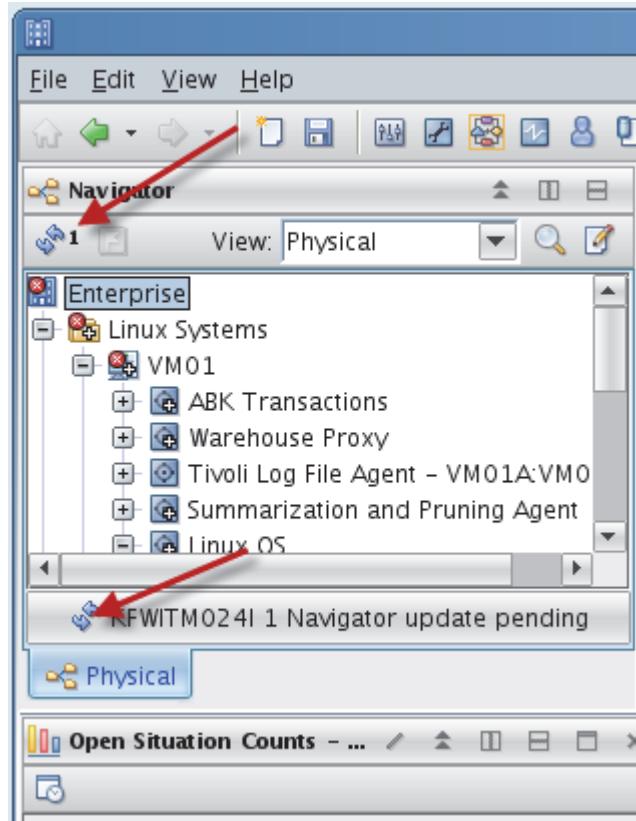


14. Select the **LinuxWebServers** instance and click **Start Agent**.

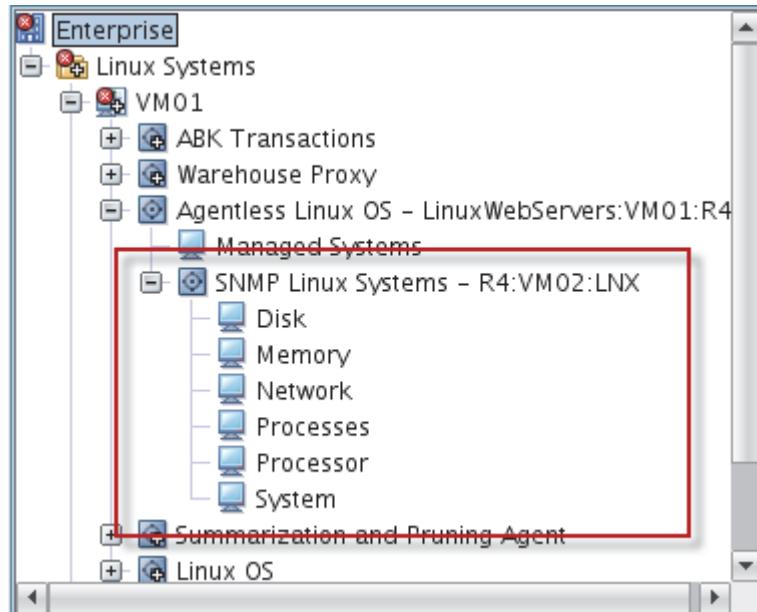


15. Wait a few minutes for the agent to start, connect, and discover resources.

16. In the Tivoli Enterprise Portal, accept the pending Navigator update by clicking one of the two Accept navigator update pending icons.



17. Locate the new Navigator item in the Tivoli Enterprise Portal for the agentless monitoring instance for VM02.

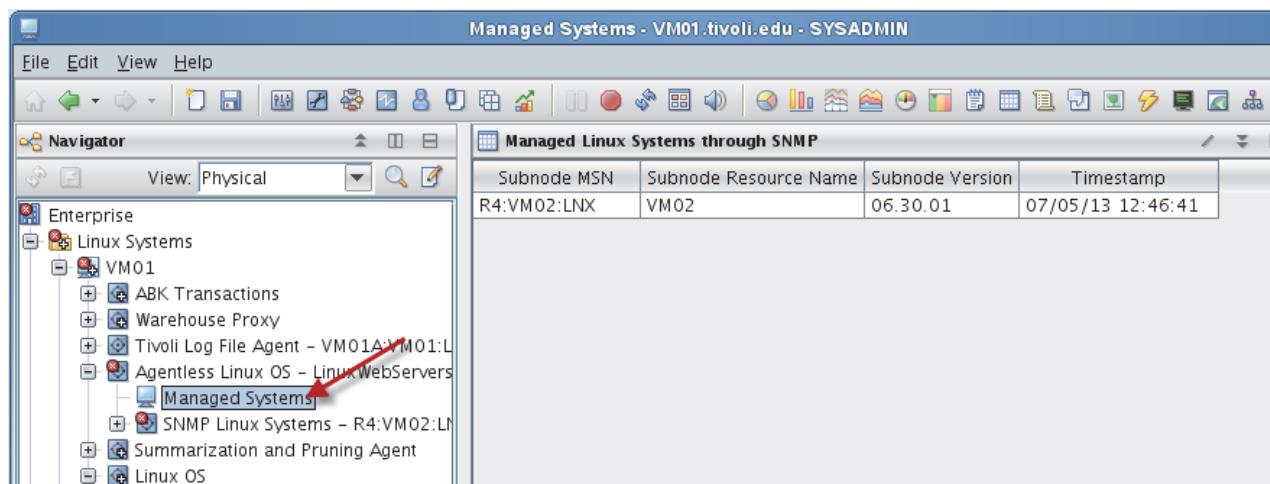


## Exercise 2. Agentless monitoring workspaces

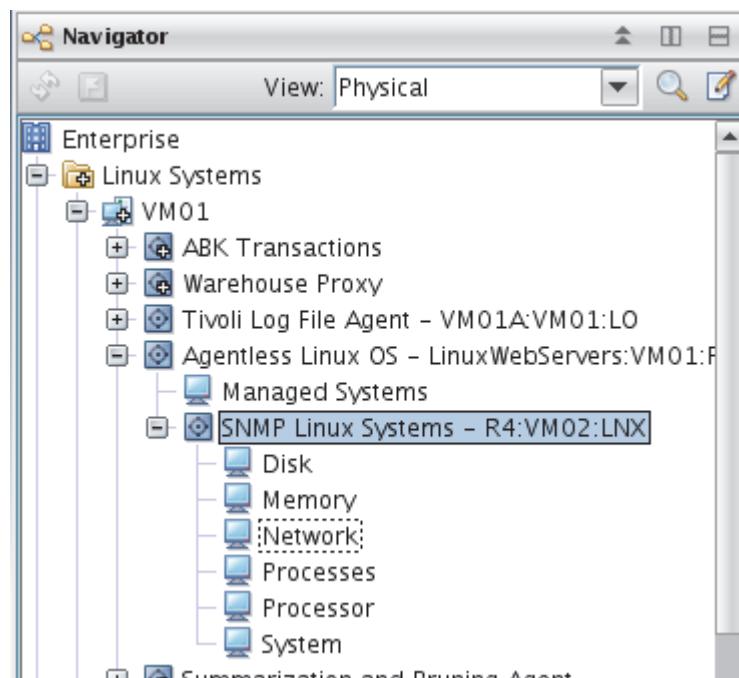
In this exercise, you explore the workspaces that are supplied with the agentless monitor. You can see what data is provided and compare that data to the data that is collected by the Linux OS agent that is also running on this system.

1. Click the **Managed Systems** workspace under **Agentless Linux OS - Linux WebServers**.

This workspace lists all systems monitored by this instance of the agentless monitor. At this point, you are monitoring a single system, VM02.

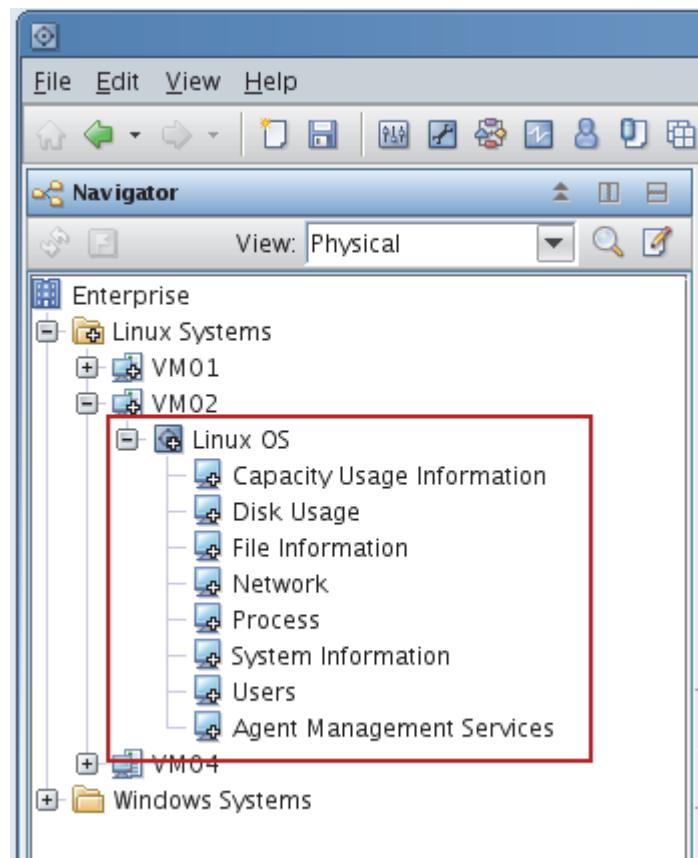


2. Review each of the workspaces for VM02, which are managed by the instance of the agentless monitor.



The navigator items shown in the following example link to the workspaces that show the data collected by the traditional Linux OS agent.

3. Review these workspaces and compare them to the data collected by the agentless monitor. For example, you can review the disk usage, physical memory, and network interfaces.



## Exercise 3. Configuring a second system for agentless monitoring

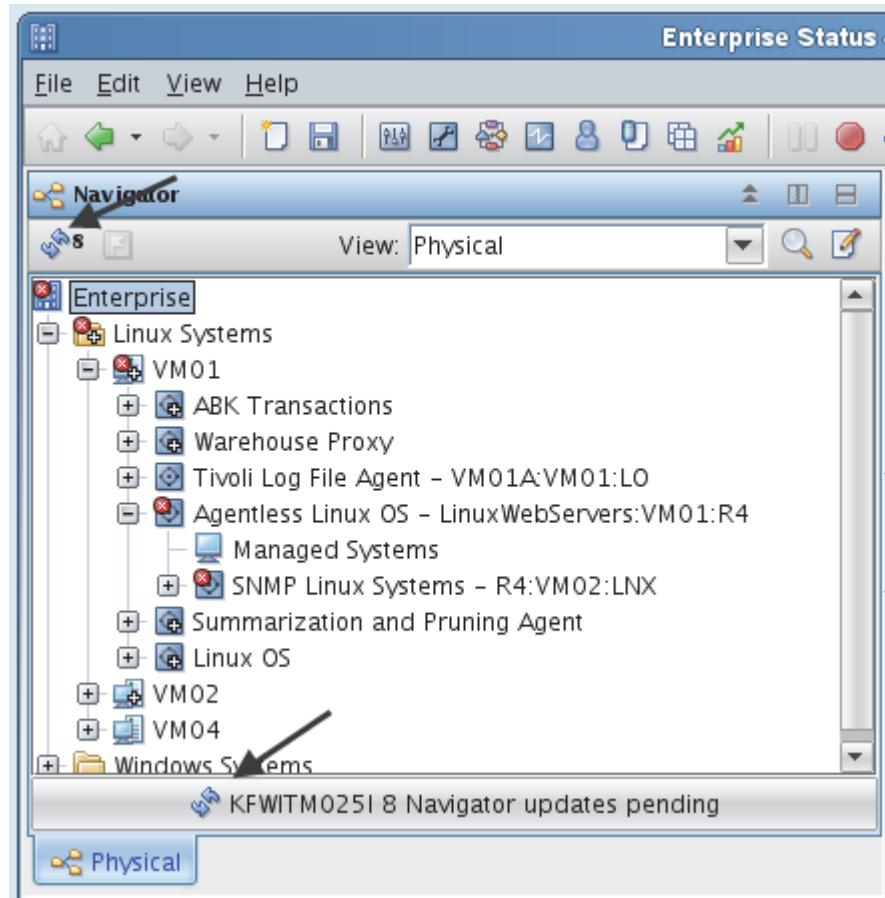
In this exercise, you configure a second Linux endpoint so that you can monitor VM05.

1. Using what you learned in the prior exercises, configure the Agentless Monitoring for Linux Operating Systems instance LinuxWebServers to also monitor **VM05**. Do not add another instance. Instead, in the **Remote Systems Details** dialog box, select **New** to add another Remote Managed System to the existing instance LinuxWebServers. Remember to stop and restart the instance after the configuration changes are made.

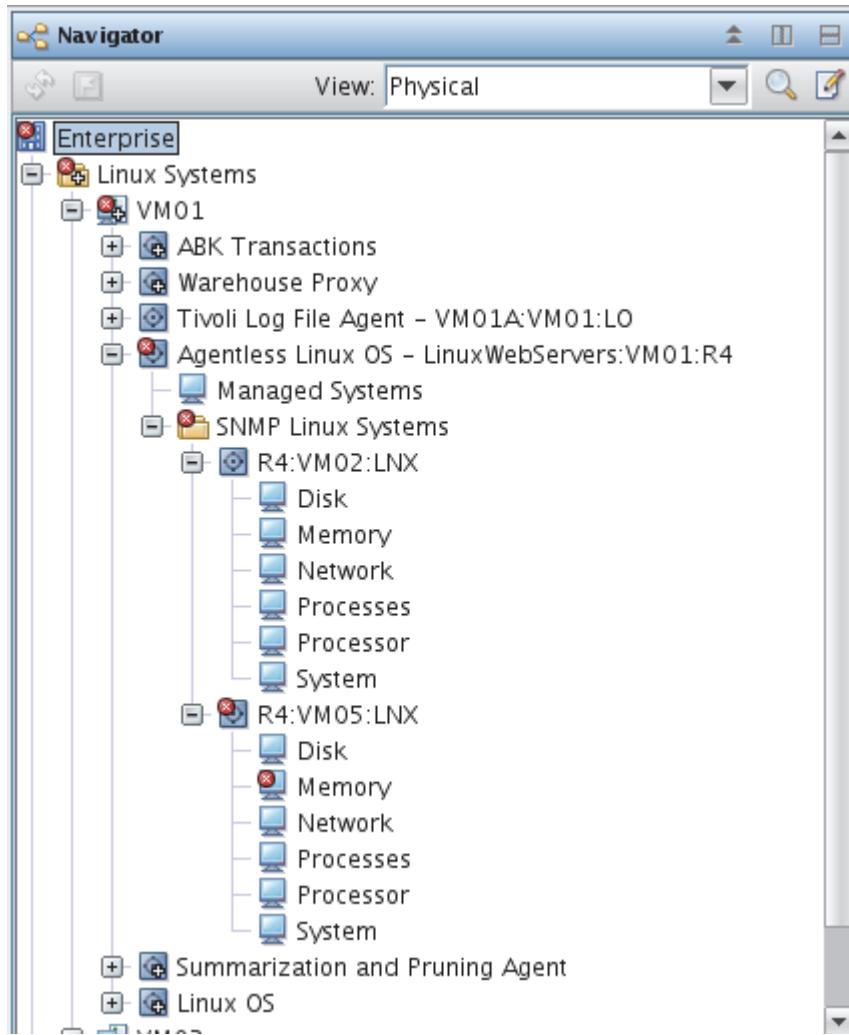


**Note:** If the new workspaces are empty, ensure that the community name on VM05 is set correctly as you did in the prior exercises. Ensure that you added a new managed system, and not a new instance.

2. Accept the pending Navigator updates after you reconfigure and restart the Agentless Monitoring for Linux Operating Systems instance.



When complete, your updated system should look like the following example.



The second system, VM05 is now configured.

Managed Linux Systems through SNMP			
Subnode MSN	Subnode Resource Name	Subnode Version	Timestamp
R4:VM02:LNX	VM02	06.30.01	07/05/13 13:20:23
R4:VM05:LNX	VM05	06.30.01	07/05/13 13:20:23

3. Review the new workspaces.



# 7 Agent Management Services exercises

## Agent Management Services overview

IBM Tivoli Agent Management Services is a strategic approach to agent management. The purpose of these services is to provide the following capabilities:

- Monitor the availability of other agents that run under the context of the operating system and respond automatically to abnormalities according to user policy.
- Monitor the availability of the Agent Management Services watchdog process and respond automatically to abnormalities with it by the operating system agent restarting this process.
- Provide a programmatic and graphical interface to start, stop, restart, manage, and unmanage an agent manually.
- Enable the deployment of new agents to the operating system.

You can use Tivoli Agent Management Services to monitor the availability of agents. Tivoli Agent Management Services can respond automatically (such as with a restart) if the agent operates abnormally or ends unexpectedly. By using these services, you can see improved agent availability ratings. As a result, you can benefit from increased use of your IBM Tivoli products and spend less time on agent management, thus reducing your operational costs.

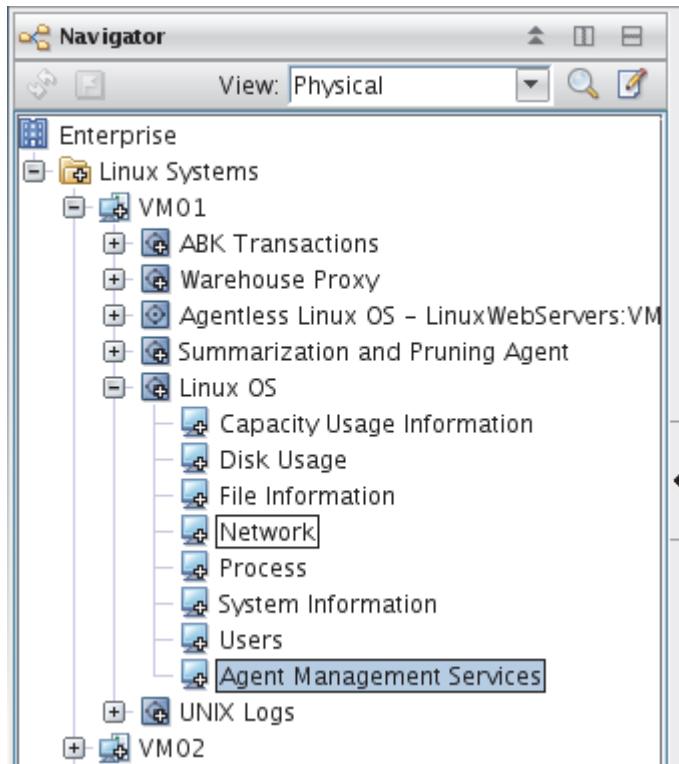
There are two components of Agent Management Services. One is called the Physical Watchdog. In Linux, this process is kcawd. This process monitors the operating system agent, and if the operating system agent stops, the physical watchdog restarts it. The second component is the Agent Watchdog. It runs inside the operating system agent. The purpose of the Agent Watchdog is to keep managed agents running by restarting the agents if they fail. It also watches the Physical Watchdog and can restart it if it fails.

You can use the user interface for the services to start, stop, and restart an agent manually. You can also use the interface to stop noncritical processes that are running on your servers. As a result, there is an increase in your ability to meet Service Level Agreements during peak utilization times. By having the option to manage or unmanage an agent, you can decide whether a Proxy Agent Service instance takes control of a particular agent.

# Exercise 1. Exploring Agent Management Services workspaces

In this exercise, you explore the Agent Management Services workspaces.

1. Examine the Agent Management Services workspace on VM01 and its views.

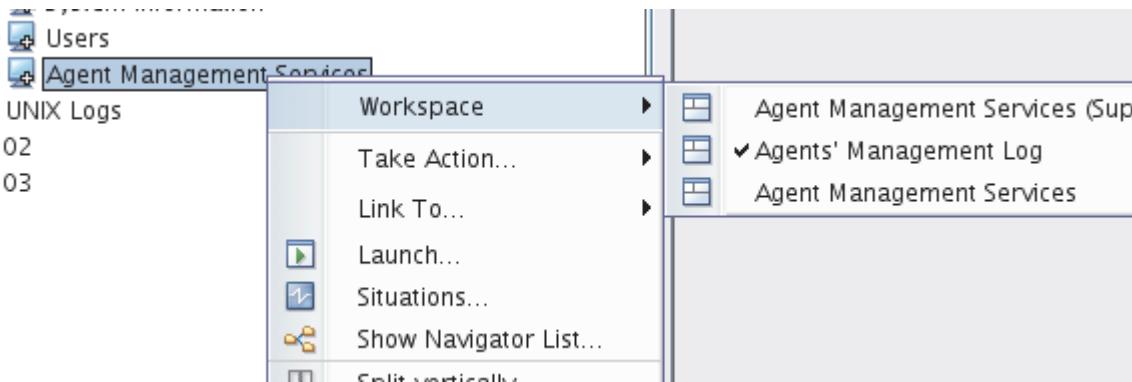


The following four views are in this workspace:

- Agents' Management Status
- Agents' Alerts
- Agents' Management Definitions
- Agents' Runtime Status

Specifically note the entries for Tivoli Log File Agent, which is the agent that you manage in this exercise.

2. Right-click the **Agent Management Services** workspace and select the **Agents' Management Log** workspace. If more than one page exists, examine each one.



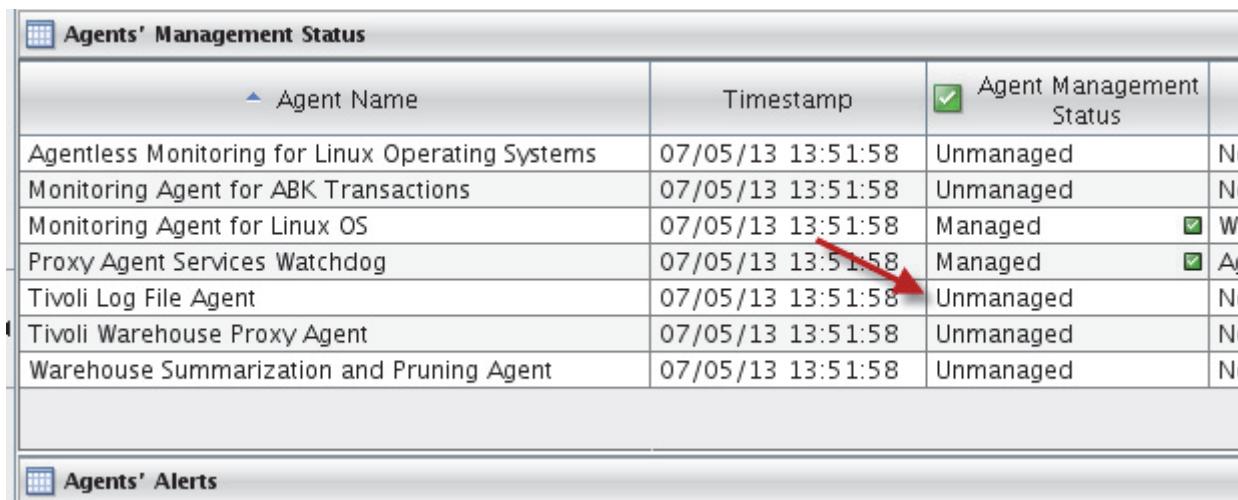
You probably see several CAP file and start messages. Review any messages that you see there and then return to the main Agent Management Services workspace by clicking **Backward**.

Agents' Management Log		
Global Timestamp	Message Number	Message Text
07/05/13 10:57:31	KLZAMS007	Proxy Agent Services Watchdog Operational Event: Agent start
07/05/13 10:57:25	KLZAMS022	Proxy Agent Services Watchdog Operational Event: Agent initia
07/05/13 10:55:25	KLZAMS000	/opt/IBM/ITM/config/CAP/ksy_default.xml CAP File Event: Age
07/05/13 10:55:22	KLZAMS000	/opt/IBM/ITM/config/CAP/kr4_default.xml CAP File Event: Age
07/05/13 10:55:19	KLZAMS000	/opt/IBM/ITM/config/CAP/klz_default.xml CAP File Event: Age
07/05/13 10:55:17	KLZAMS000	/opt/IBM/ITM/config/CAP/klo_default.xml CAP File Event: Age
07/05/13 10:55:13	KLZAMS000	/opt/IBM/ITM/config/CAP/khd_default.xml CAP File Event: Age
07/05/13 10:55:09	KLZAMS000	/opt/IBM/ITM/config/CAP/kca_default.xml CAP File Event: Age
07/05/13 10:55:05	KLZAMS000	/opt/IBM/ITM/config/CAP/k99_default.xml CAP File Event: Ag

## Exercise 2. Exploring Agent Management Services features

In this exercise, you explore the Agent Management Services features. You start by managing an agent. Then, you simulate a failure by using the supplied Sample Linux Kill Process Take Action command and watch the agent be restarted. You do this several times to exceed the daily restart count and thus trigger a situation to alert you about the excessive restarts. Then, you restart the agent and reset the daily count, as you do after resolving an issue.

1. Return to the Agent Management Services workspace. Observe that the status of the Tivoli Log File Agent is **Unmanaged** in the Agents Management Status view. The status is Unknown in the Agents' Runtime Status view.



Agent Name	Timestamp	Agent Management Status
Agentless Monitoring for Linux Operating Systems	07/05/13 13:51:58	Unmanaged
Monitoring Agent for ABK Transactions	07/05/13 13:51:58	Unmanaged
Monitoring Agent for Linux OS	07/05/13 13:51:58	Managed
Proxy Agent Services Watchdog	07/05/13 13:51:58	Managed
Tivoli Log File Agent	07/05/13 13:51:58	Unmanaged
Tivoli Warehouse Proxy Agent	07/05/13 13:51:58	Unmanaged
Warehouse Summarization and Pruning Agent	07/05/13 13:51:58	Unmanaged

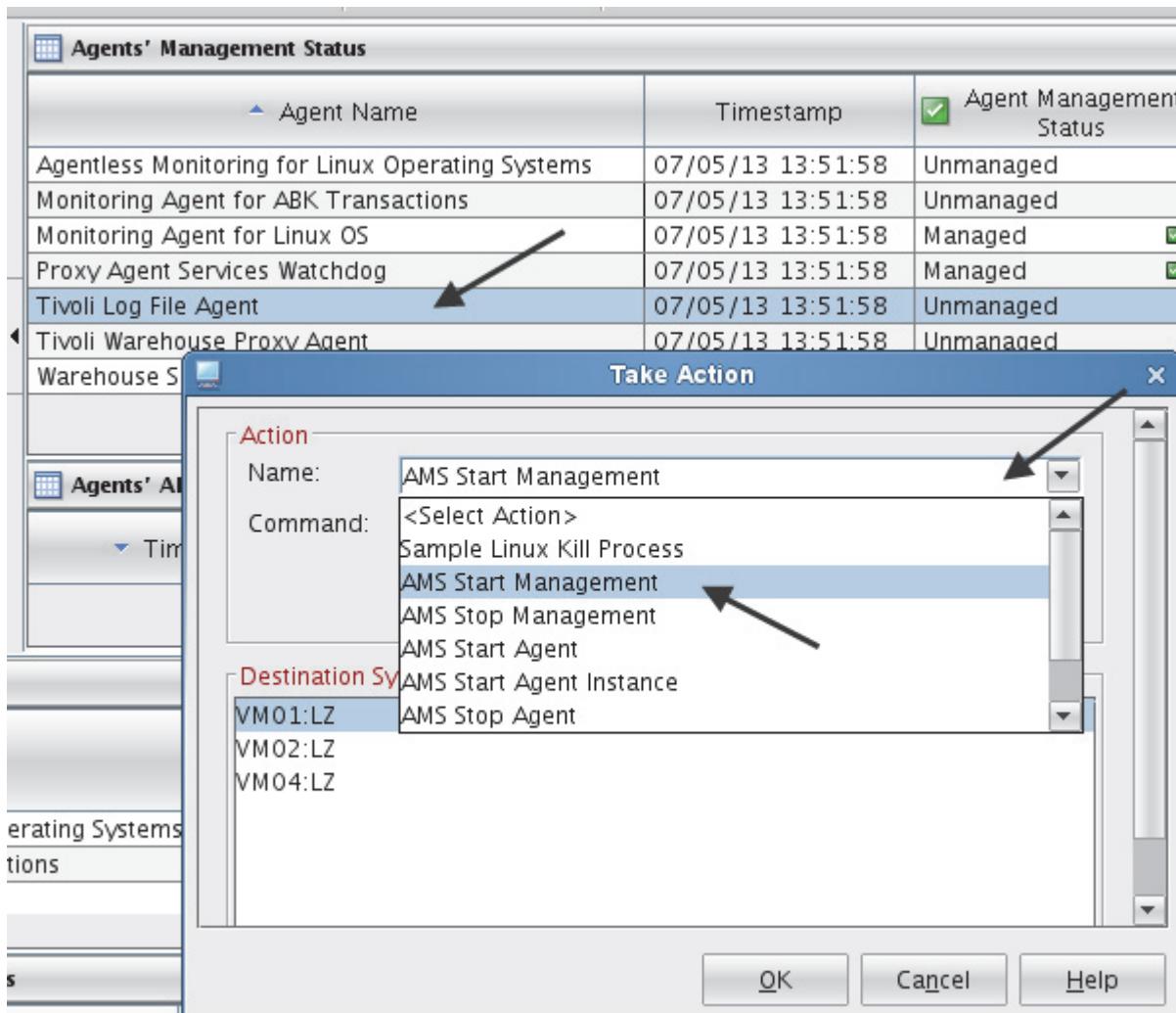
The Manager Type column in the Agents' Management Status view can have the following values:

- **Unknown**
- **Not\_Managed**: The agent is not under availability monitoring by any application
- **Agent\_Management\_Services**: Agent Management Services is responsible
- **Watchdog**
- **External**: An application other than Agent Management Services is responsible for availability monitoring of the agent, for example Tivoli System Automation

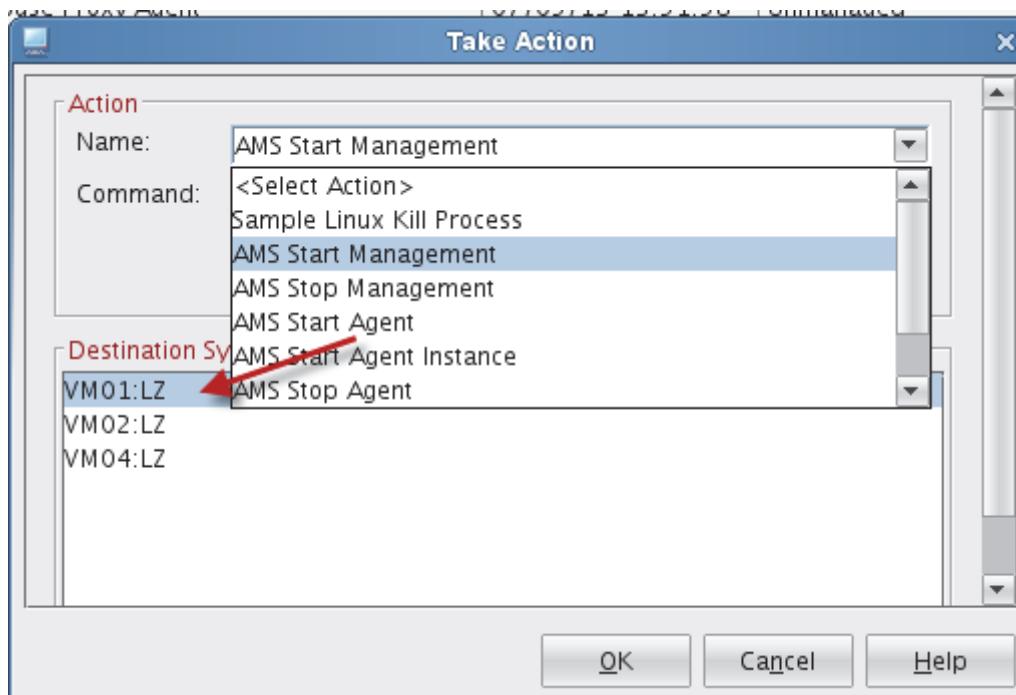
The valid Agent Management Status values are as follows:

- **Unmanaged**: The agent is known, but not under the management of Agent Management Services
- **Managed**: The agent is under the management of Agent Management Services

- **Watchdog**
2. Change the status of the Tivoli Log File Agent to Managed.
    - a. From the Agents' Management Status view, right-click **Tivoli Log File Agent**.
    - b. In the drop-down menu, click **Take Action > Select**.
    - c. From the Take Action window, in the **Name** menu, select **AMS Start Management**.



- d. Select **VM01:LZ** as the destination system.



- e. Click **OK** to exit the Take Action window.  
f. Click **OK** to exit the Action Status window.  
g. Refresh the portal and note that the status is now **Managed** in the Agents' Management Status view.

Agent Name	Timestamp	Agent Management Status
Agentless Monitoring for Linux Operating Systems	07/05/13 13:56:12	Unmanaged
Monitoring Agent for ABK Transactions	07/05/13 13:56:12	Unmanaged
Monitoring Agent for Linux OS	07/05/13 13:56:12	Managed
Proxy Agent Services Watchdog	07/05/13 13:56:12	Managed
Tivoli Log File Agent	07/05/13 13:56:12	Managed
Tivoli Warehouse Proxy Agent	07/05/13 13:56:12	Unmanaged
Warehouse Summarization and Pruning Agent	07/05/13 13:56:12	Unmanaged

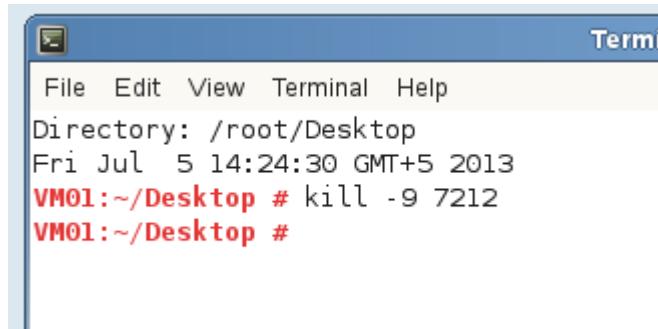
The status on the Agents' Runtime Status view is **Running**.

Agent Name	Timestamp	Agent Availability Status	Process Name	Instance Name	Process ID	A
Agentless Monitoring for Linux Operating Systems	07/05/13 13:56:12	Unknown	kr4agent		0	ITM
Monitoring Agent for ABK Transactions	07/05/13 13:56:12	Unknown	k99agent	None	0	ITM
Monitoring Agent for Linux OS	07/05/13 13:56:12	Running	klzagent	None	7638	ITM
Proxy Agent Services Watchdog	07/05/13 13:56:12	Running	kcawd		25807	Cor
Tivoli Log File Agent	07/05/13 13:56:12	Running	kloagent	VM01A	7212	ITM
Tivoli Warehouse Proxy Agent	07/05/13 13:56:12	Unknown	khdprtj	None	0	ITM
Warehouse Summarization and Pruning Agent	07/05/13 13:56:12	Unknown	ksy610	None	0	ITM

#### Agents' Management Definitions

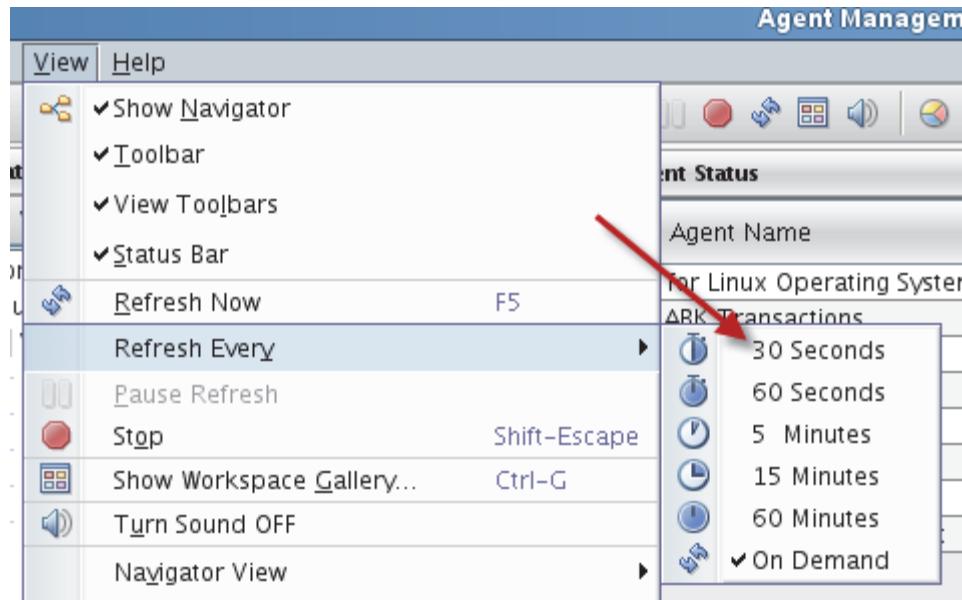
There is a process ID associated with the agent.

3. Make note of that process ID. In this case, the value is 7212. Your value is probably different. You can use a Take Action called Sample Linux Kill Process to stop a process. However, to save time during this exercise, do not use that take action. Instead, open a terminal window and use it for stopping processes.
4. Open a terminal window.
5. From the terminal window, kill the process by supplying the process ID for the Tivoli Log File Agent. This process ID must match the process ID noted earlier.



The screenshot shows a terminal window titled "Terminal". The window has a menu bar with "File", "Edit", "View", "Terminal", and "Help". Below the menu is a status line showing "Directory: /root/Desktop" and the date and time "Fri Jul 5 14:24:30 GMT+5 2013". The main area of the terminal shows the command "VM01:~/Desktop # kill -9 7212" being entered, followed by a red error message "VM01:~/Desktop #".

- Set the Agent Management Services workspace to refresh every 30 seconds.



- Note the alert message in the Agents' Alerts view.

Agents' Alerts			
Timestamp	Alert Message	Alert Details	Agent Name
07/05/13 14:25:25	Agent abnormally stopped	N/A	Tivoli Log File Agent

When an agent is stopped, the process ID of the agent is zero (0). While an agent is being restarted, the status is Start Pending. Look at the process ID for the Tivoli Log File Agent. The update messages and alerts can take several refresh cycles before being displayed.

- Wait for the **Start Pending** status to be displayed for the Tivoli Log File Agent.

Agent Name	Timestamp	Agent Availability Status	Process Name	Instance Name	Process ID	
Agentless Monitoring for Linux Operating Systems	07/05/13 14:28:29	Unknown	kr4agent		0	IT
Monitoring Agent for ABK Transactions	07/05/13 14:28:29	Unknown	k99agent	None	0	IT
Monitoring Agent for Linux OS	07/05/13 14:28:29	Running	lizagent	None	7638	IT
Proxy Agent Services Watchdog	07/05/13 14:28:29	Running	kcawd		25807	Co
Tivoli Log File Agent	07/05/13 14:28:29	Start Pending	kloagent	VM01A	0	IT
Tivoli Warehouse Proxy Agent	07/05/13 14:28:29	Unknown	khdxprtj	None	0	IT

The process ID number changed, indicating that the agent was successfully restarted.

9. Scroll this view to the right and examine the Daily Restart Count column and the entry for this agent.

Daily Restart Count
0
0
0
0
1
0
0

By default, agents controlled by Agent Management Services can restart five times within 24 hours. Messages in the Agents' Management Log workspace indicate that the agent stopped abnormally and was restarted.

Message Text
Tivoli Warehouse Proxy Agent - None - Operational Event: Agent started successfully.
Tivoli Warehouse Proxy Agent - None - Operational Event: Agent start command received.
Tivoli Warehouse Proxy Agent - None - Operational Event: Agent stopped abnormally.
Tivoli Warehouse Proxy Agent - None - Operational Event: Agent started successfully.
Tivoli Warehouse Proxy Agent - None - Operational Event: Agent start command received.
Tivoli Warehouse Proxy Agent - None - Operational Event: Agent stopped abnormally.
Tivoli Warehouse Proxy Agent - None - Operational Event: Agent started successfully.
Tivoli Warehouse Proxy Agent - None - Operational Event: Agent start command received.
Tivoli Warehouse Proxy Agent - None - Operational Event: Agent stopped abnormally.
Tivoli Warehouse Proxy Agent - None - Management Event: Agent now managed.

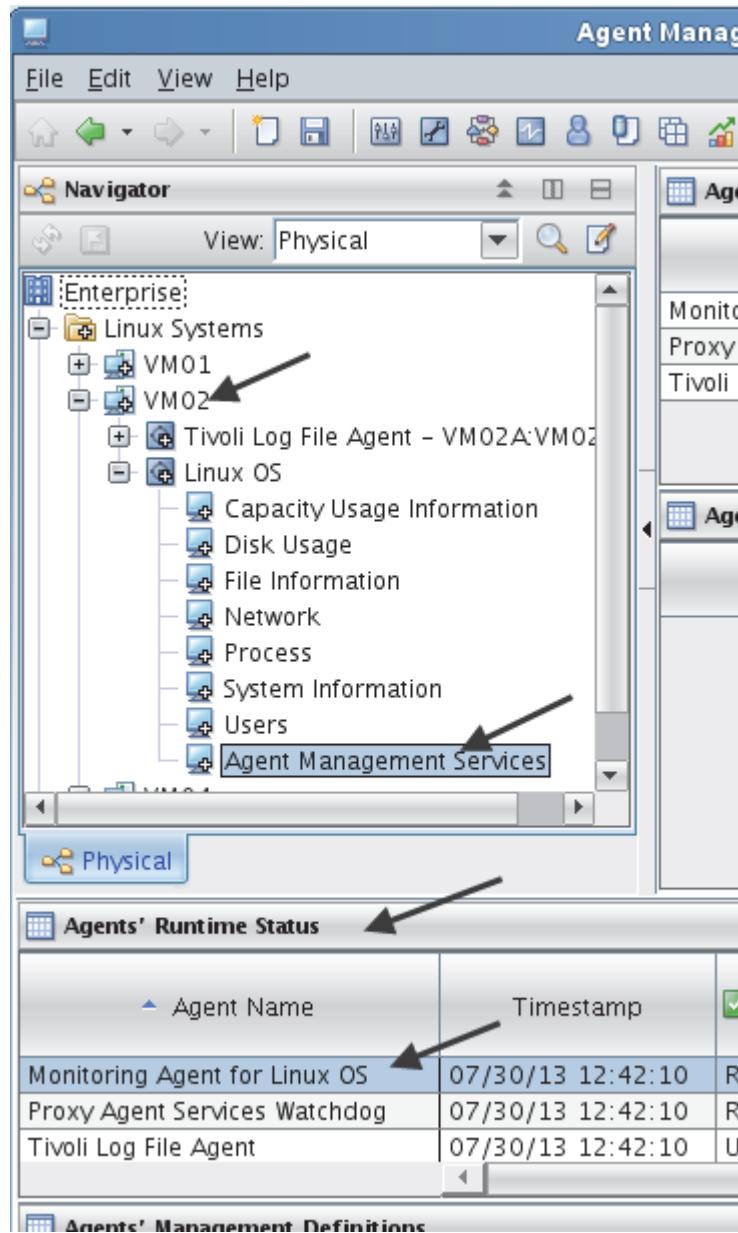


**Note:** If you repeat the process of killing the process ID for the Tivoli Log File Agent five times, you see a situation event on the situation event console.

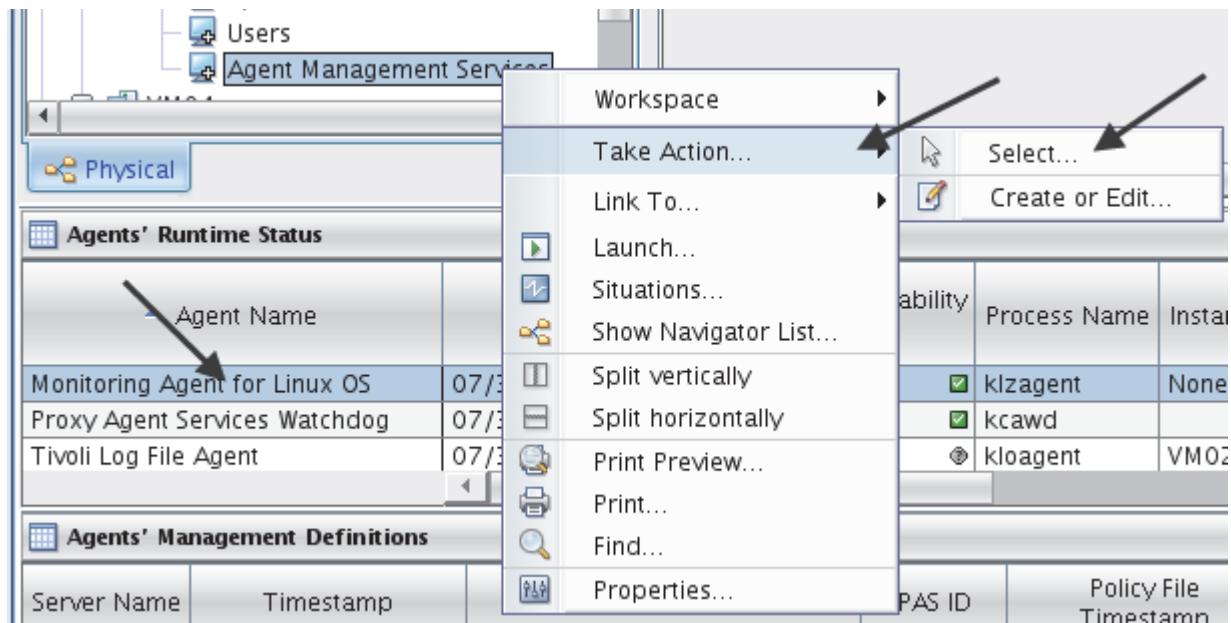
## Exercise 3. Restarting the Monitoring Agent for Linux OS by using Agent Management Services

Using Agent Management Services, you can restart operating system agents from the Tivoli Enterprise Portal.

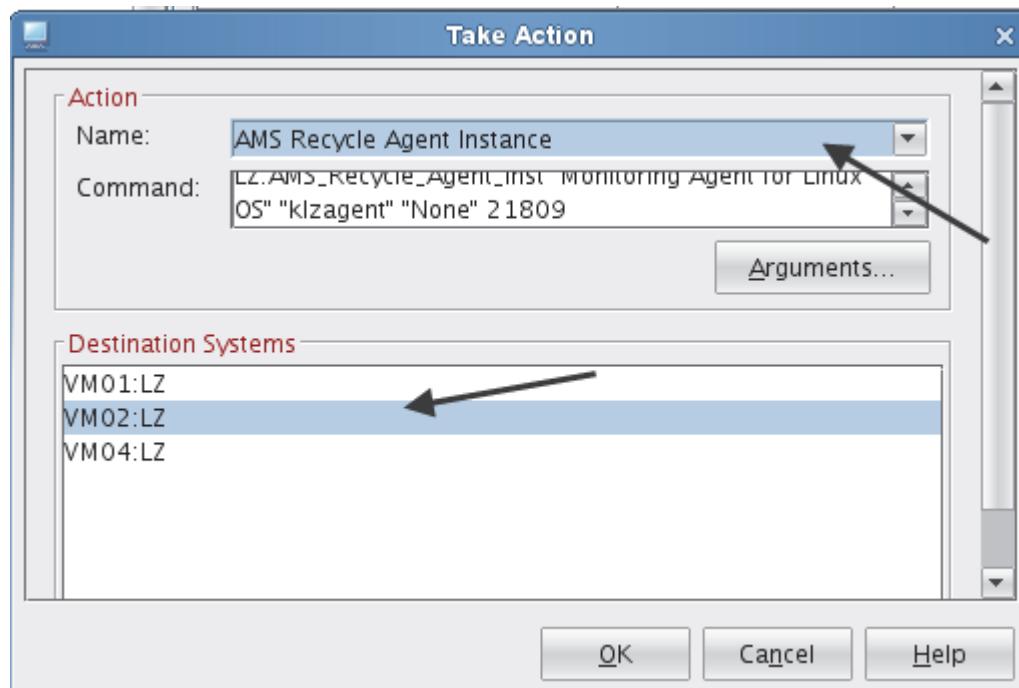
1. Locate the Agents' Runtime Status view for Linux OS on VM02 in the Navigator Physical view.



2. Right-click **Monitoring Agent for Linux OS** and select **Take Action > Select**.



3. Scroll down and select **AMS Recycle Agent Instance**. Select **VM02:LZ**.



4. Click **OK**.

5. Observe the Monitoring Agent go offline and return in the Tivoli Enterprise Portal.



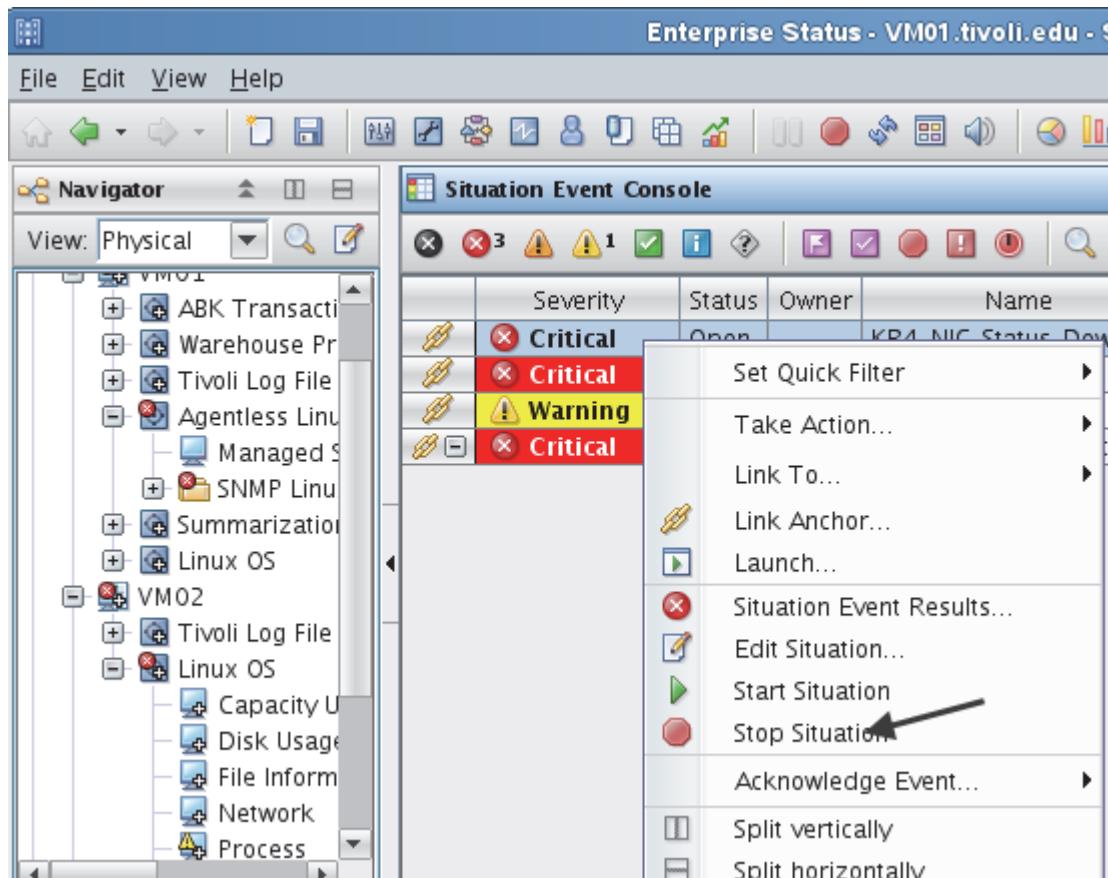
## 8 Managing situation event integration exercises

In these exercises, you configure the Tivoli Enterprise Portal, Tivoli Enterprise Portal Server, and the Tivoli Enterprise Monitoring Server to forward situation events to Netcool/OMNibus. You also customize the Event Integration Facility (EIF) slots that are sent in the situation events.

# Exercise 1. Configuring IBM Tivoli Monitoring to forward situation events to Netcool/OMNibus

In this exercise, you configure the Tivoli Enterprise Monitoring Server and the Tivoli Enterprise Portal to forward situation events to Netcool/OMNibus.

1. On VM01, stop any open situations in the situation event console.



2. On VM04, delete all events in the Netcool/OMNibus Event List. Select **Edit > Select All**, right-click one of the events, and select **Delete**.



3. On VM01, open a terminal session and run the following commands. When prompted for a user ID and password, use **sysadmin** with no password.



**Note:** The **tacmd createEventDest** command is providing the number one (1) as input to the first parameter.

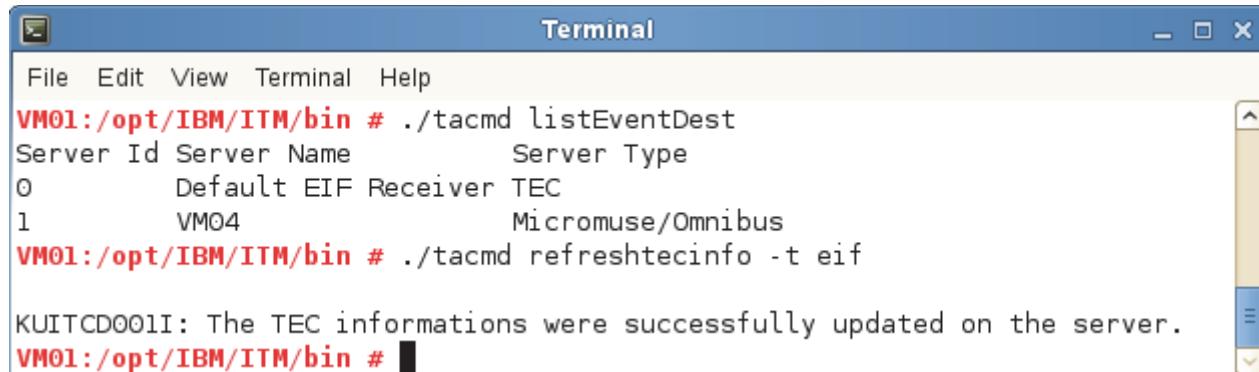
```
cd /opt/IBM/ITM/bin/
./tacmd login -s VM01 -u sysadmin
./tacmd listEventDest
./tacmd createEventDest -i 1 -p host1=VM04:9998 name=VM04 type=M
```

The screenshot shows a terminal window titled "Terminal". The session starts with the user navigating to the ITM bin directory and logging in as "sysadmin". The user then runs the "listEventDest" command to check existing destinations. Finally, the user runs the "createEventDest" command to add a new destination named "VM04" with server ID 1 and type M, pointing to host "VM04:9998". The terminal prompts for confirmation before creating the destination.

```
File Edit View Terminal Help
Directory: /root/Desktop
Tue Jul 30 09:26:43 GMT+5 2013
VM01:~/Desktop # cd /opt/IBM/ITM/bin/
VM01:/opt/IBM/ITM/bin # ./tacmd login -s VM01 -u sysadmin
Password?
Validating user...
KUIC00007I: User sysadmin logged into server on https://VM01:3661.
VM01:/opt/IBM/ITM/bin # ./tacmd listEventDest
Server Id Server Name           Server Type
0      Default EIF Receiver TEC
VM01:/opt/IBM/ITM/bin # ./tacmd createEventDest -i 1 -p host1=VM04:9998 name=VM04 type=M
KUICCE004I: Are you sure you want to create the event destination server definition VM04 with server ID 1 on the server?
Enter Y for yes or N for no: y
KUICCE007I: The event destination server definition VM04 with server ID 1 was successfully created on the server at https://VM01:3661. Note that Hub TEMS needs to be recycled or refreshed for this to take effect.
VM01:/opt/IBM/ITM/bin #
```

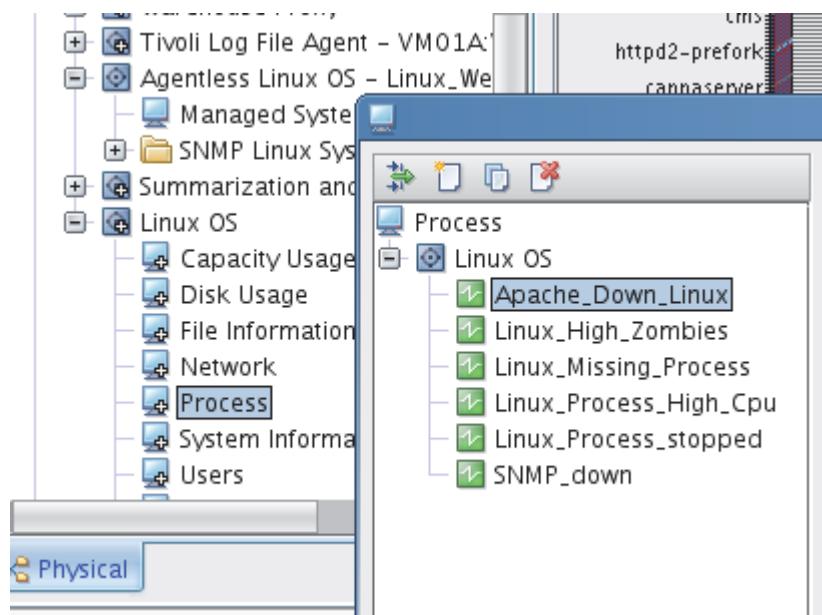
4. Run the **./tacmd listEventDest** command again and notice the update.

5. Run the `./tacmd refreshtecinfo -t eif` command to refresh the monitoring server.

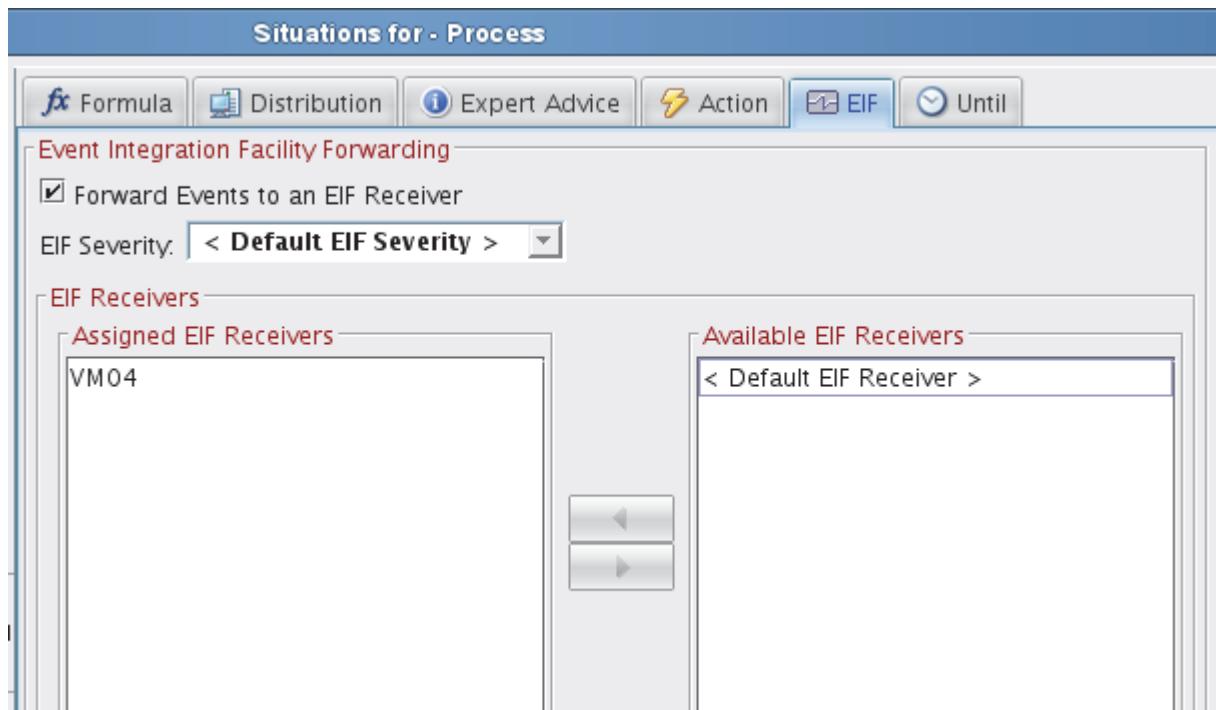


```
VM01:/opt/IBM/ITM/bin # ./tacmd listEventDest
Server Id Server Name           Server Type
0          Default EIF Receiver TEC
1          VM04                 Micromouse/Omnibus
VM01:/opt/IBM/ITM/bin # ./tacmd refreshtecinfo -t eif
KUITCD001I: The TEC informations were successfully updated on the server.
VM01:/opt/IBM/ITM/bin #
```

6. On VM01, open the situation, **Apache\_Down\_Linux** (in the Situation editor under the Linux OS>Process folder).



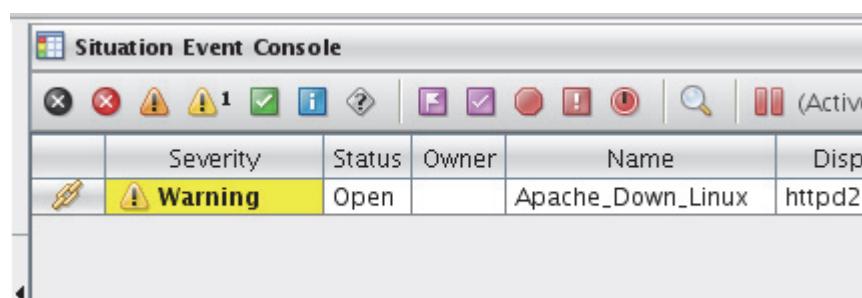
7. Open the EIF notebook page in the Situation editor and select **Forward Events to an EIF Receiver**. Move VM04 to be the Assigned EIF Receivers for the situation, and move <Default EIF receiver> to the Available EIF Receivers.



You created the event receiver of VM04, and it is displayed in the Available EIF receivers window.

8. Click **OK** to close the Situation editor.
9. On VM01, ensure that the Apache web server is stopped by entering the following command:  

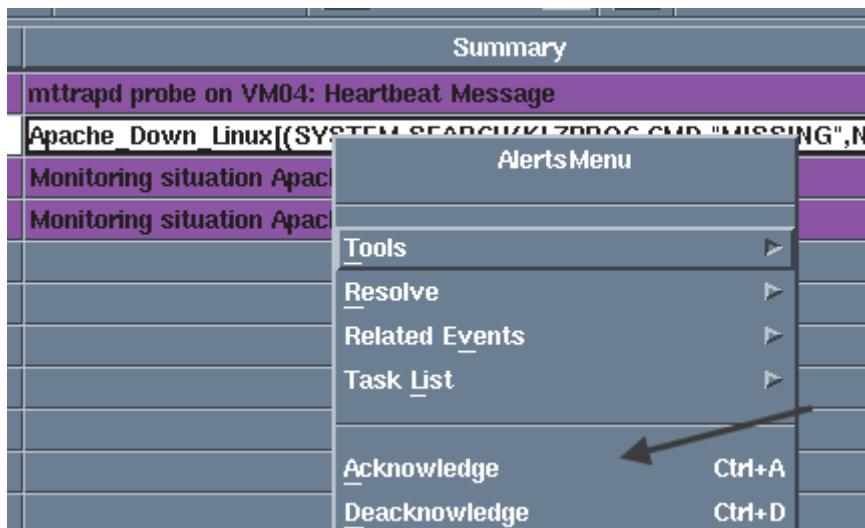
```
cd /etc/init.d
./apache2 stop
```
10. Locate the situation event in the situation event console on VM01.



11. Locate the event in Netcool/OMNibus.

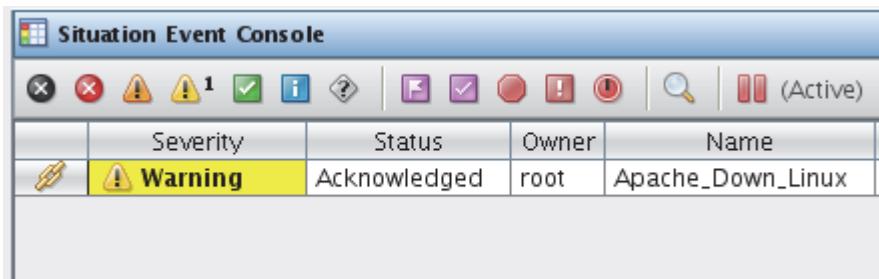
Node	Alert Group	Summary
VM01:LZ	ITM_KLZ_Process	Apache_Down_Linux[(SYSTEM.SEARCH(KLZPROC.CMD,"MISSING",N)]
VM02:LZ	ITM_StatEvent	Monitoring situation Apache_Down_Linux stopped
VM02:LZ	Status ( 3 )	Monitoring situation Apache_Down_Linux stopped
VM04		mttrapd probe on VM04: Heartbeat Message

12. Right-click the event in Netcool/OMNibus, and select **Acknowledge**.

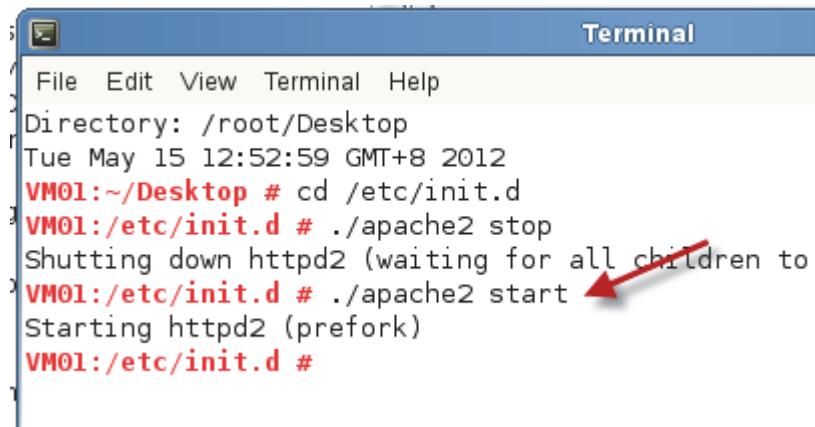


**Note:** Event synchronization was installed and configured on VM04. This setup keeps the situation events synchronized with events that they are forwarded to. These events are the ones that are acknowledged or closed on VM01 within IBM Tivoli Monitoring.

13. Return to VM01 and see that the situation is acknowledged in the situation event console.



14. Restart the Apache web server on VM01.



A terminal window titled "Terminal" showing the command line interface. The directory is /root/Desktop. The user runs "cd /etc/init.d" and then "../apache2 start". A red arrow points to the "start" command.

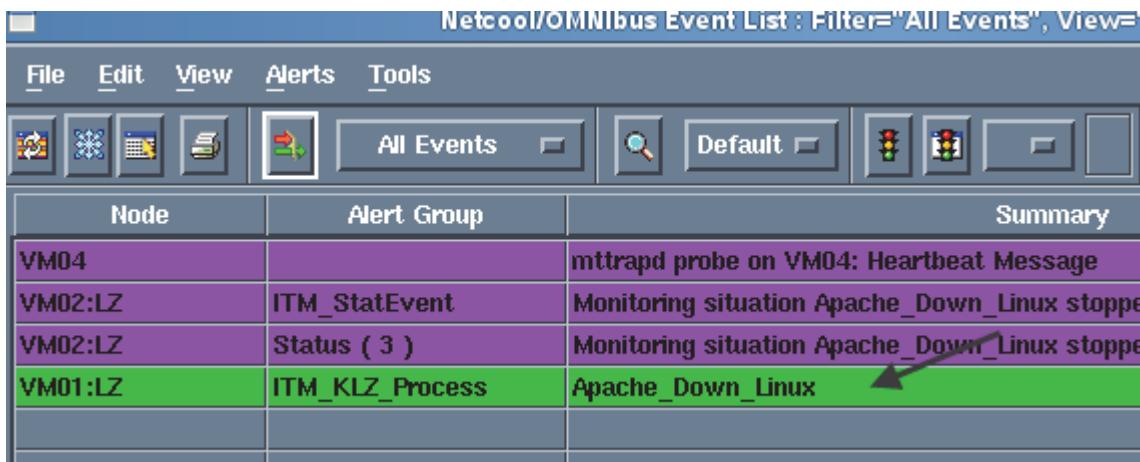
```

File Edit View Terminal Help
Directory: /root/Desktop
Tue May 15 12:52:59 GMT+8 2012
VM01:~/Desktop # cd /etc/init.d
VM01:/etc/init.d # ./apache2 stop
Shutting down httpd2 (waiting for all children to
VM01:/etc/init.d # ./apache2 start
Starting httpd2 (prefork)
VM01:/etc/init.d #

```

15. Wait until the situation event clears.

16. Observe the resolution event arrive and eventually close in Netcool/OMNibus.



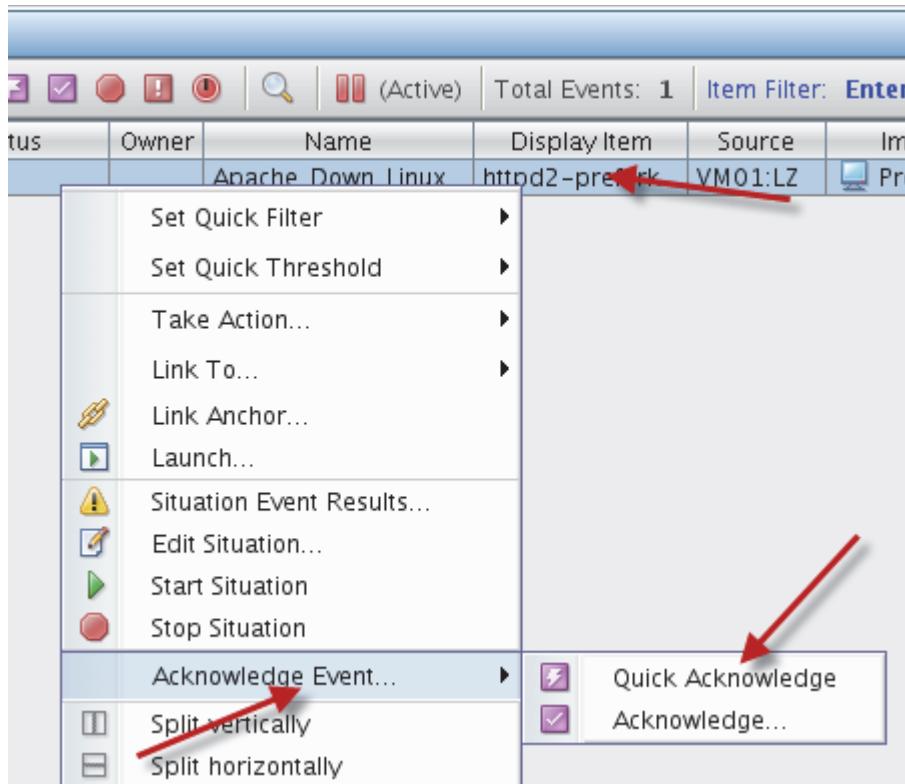
The screenshot shows the "Netcool/OMNibus Event List" window. The title bar says "Netcool/OMNibus Event List : Filter="All Events", View=". The menu bar includes File, Edit, View, Alerts, and Tools. Below the menu is a toolbar with various icons. The main area is a table with columns: Node, Alert Group, and Summary. There are four rows of data:

Node	Alert Group	Summary
VM04		mttrapd probe on VM04: Heartbeat Message
VM02:LZ	ITM_StatEvent	Monitoring situation Apache_Down_Linux stoppe
VM02:LZ	Status ( 3 )	Monitoring situation Apache_Down_Linux stoppe
VM01:LZ	ITM_KLZ_Process	Apache_Down_Linux

A red arrow points to the last row's "Summary" column for VM01:LZ.

17. Stop the Apache web server again on VM01.

18. Acknowledge the Apache\_Down\_Linux situation event from the situation event console.



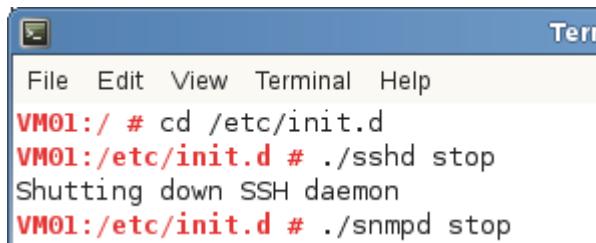
19. Return to VM04. Wait for a few minutes. Locate the Apache\_Down\_Linux event and note that it is a different color now. The change in color indicates that an acknowledged event is available.

Netcool/OMNibus Event List : Filter="All Events", View="Default"		
File	Edit	View
Alerts	Tools	
All Events	Default	
Node	Alert Group	Summary
VM04		mttrapd probe on VM04: Heartbeat Message
VM01:LZ	ITM_KLZ_Process	Apache_Down_Linux
VM02:LZ	ITM_StatEvent	Monitoring situation Apache_Down_Linux stopped
VM02:LZ	Status ( 3 )	Monitoring situation Apache_Down_Linux stopped

20. Restart the Apache web server again on VM01.

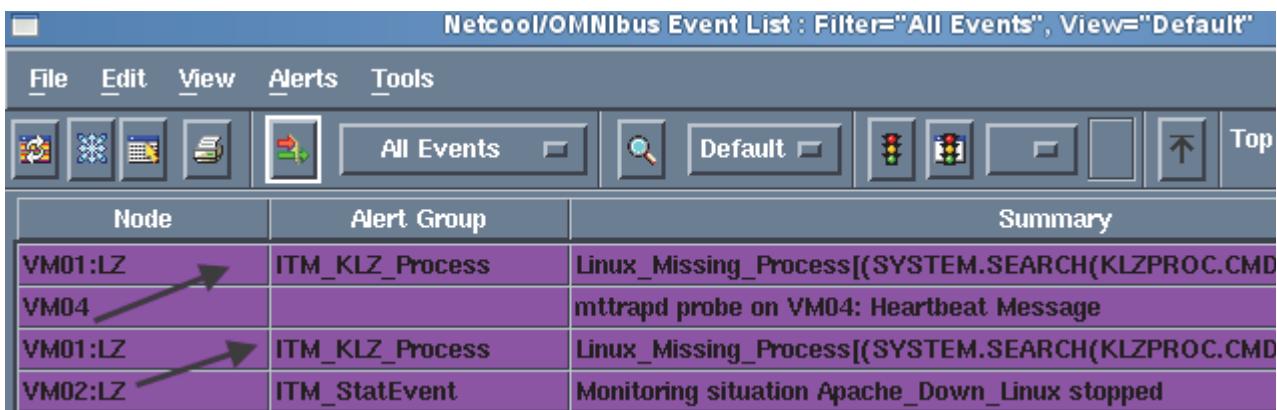
## Exercise 2. Customizing the EIF slot

1. Using the Situation editor, update the Linux\_Missing\_Process situation to forward to the EIF receiver on VM04, and start the situation by selecting Run at Startup. Ensure that the System Command in the Action tab has been cleared.
2. If the snmpd and sshd daemons are started, stop them.



```
VM01:/ # cd /etc/init.d
VM01:/etc/init.d # ./sshd stop
Shutting down SSH daemon
VM01:/etc/init.d # ./snmpd stop
```

3. Return to VM04 and locate the two events created by this situation.



Netcool/OMNIBus Event List : Filter="All Events", View="Default"			
File	Edit	View	Alerts
		All Events	
Node	Alert Group	Summary	
VM01:LZ	ITM_KLZ_Process	Linux_Missing_Process[(SYSTEM.SEARCH(KLZPROC.CMD	
VM04		mttrapd probe on VM04: Heartbeat Message	
VM01:LZ	ITM_KLZ_Process	Linux_Missing_Process[(SYSTEM.SEARCH(KLZPROC.CMD	
VM02:LZ	ITM_StatEvent	Monitoring situation Apache_Down_Linux stopped	

4. Observe that, from this message slot, you cannot tell which application is down without opening the event or increasing the width of the summary column.
5. Return to VM01 and run the following commands:

```
cd /opt/IBM/ITM/bin
./tacmd login -s VM01
./tacmd viewsit -s Linux_Missing_Process
```

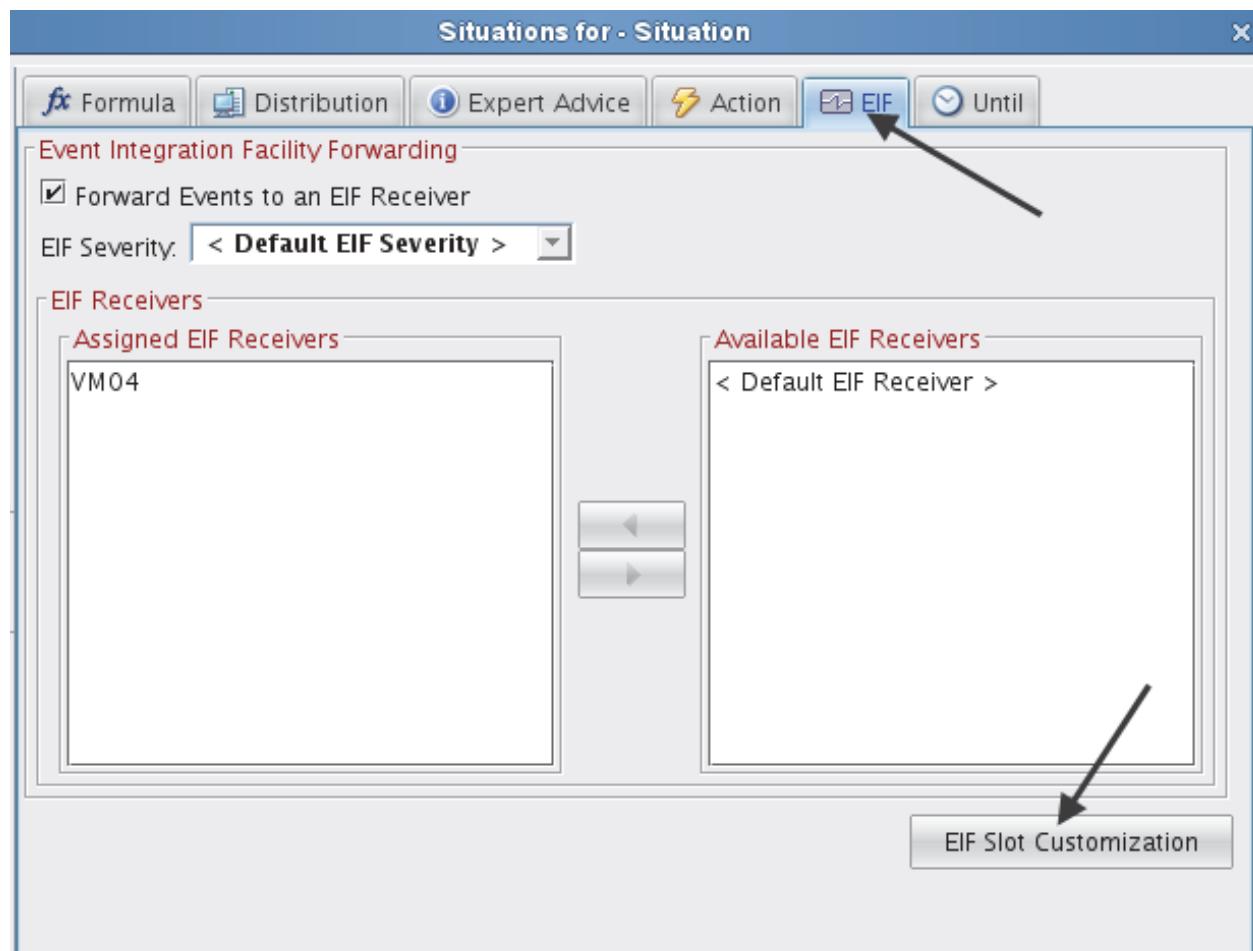
## Exercise 2. Customizing the EIF slot

Look at the formula string. The **KLZ\_Process.Process\_Command\_Name** is the attribute table and the attribute name. This string is used in the slot value.

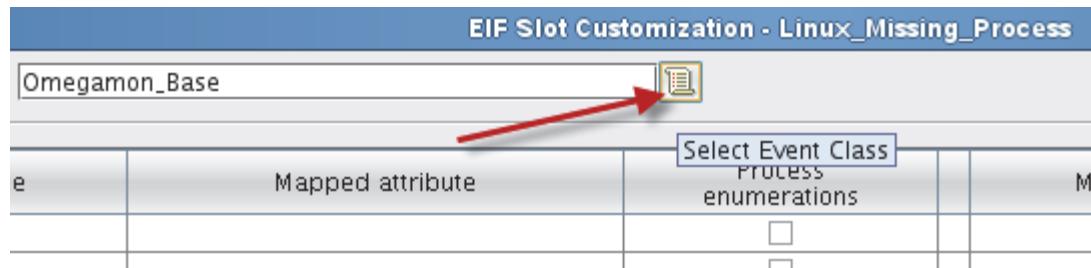
Formula: \*IF \*MISSING KLZ\_Process.Process\_Command\_Name \*EQ ( 'snmpd', 'sshd' )

```
VM01:/opt/IBM/ITM/bin # ./tacmd viewsit -s Linux_Missing_Process
Name : Linux_Missing_Process
Full Name :
Description : Check for a missing process on Linux
Type : Linux OS
Formula : *IF *MISSING KLZ_Process.Process_Command_Name *EQ ( 'snmpd', 'sshd' )
)
Sampling Interval : 0/0:0:30
Run At Start Up : Yes
Distribution : VM01:LZ
Text :
Action Location : Agent
Action Selection : System Command
System Command : *NONE
True For Multiple Items: Action on First Item only
TFC Severity :
```

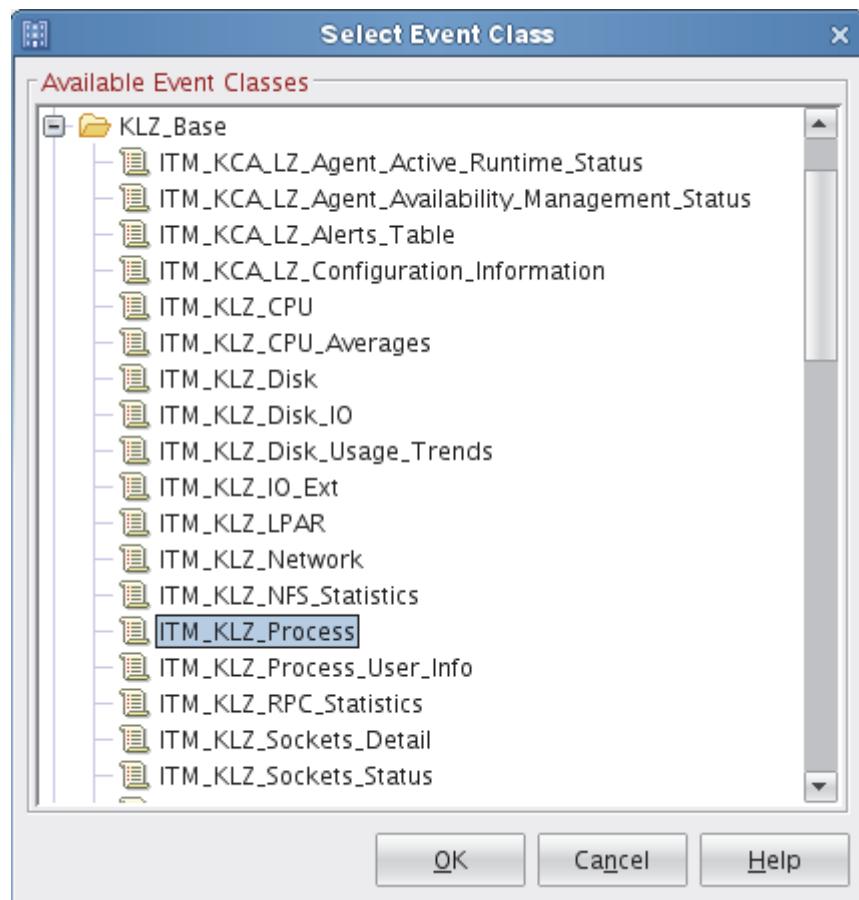
6. Open the Situation editor and edit the situation `Linux_Missing_Process`.
7. Click the EIF notebook page, click **Forward Event to EIF Receiver**, and click **EIF Slot Customization**.



8. Click the **Select Event Class** icon.



9. Expand **KLZ\_Base**, select **ITM\_KLZ\_Process**, and click **OK**.



10. Select **Map all attributes**.

The screenshot shows the EIF Slot Customization interface. At the top, there is a table with two rows: 'SEVERITY' and 'source'. Below this is a section titled 'Extended slots' containing a table with the following data:

Slot name
busy_cpu_pct
data_set_size
data_set_size_enum
dirty_pages
dirty_pages_enum
nice
parent_process_id
parent_process_id_enum
priority
proc_busy_cpu_norm

At the bottom of this section is a checkbox labeled 'Map all attributes' which is checked. A large black arrow points from the text 'busy\_cpu\_norm' in the 'Extended slots' table towards this checked checkbox.

11. In the Base slots view, locate the **msg** slot name, scroll right, and double-click in the Literal value column.

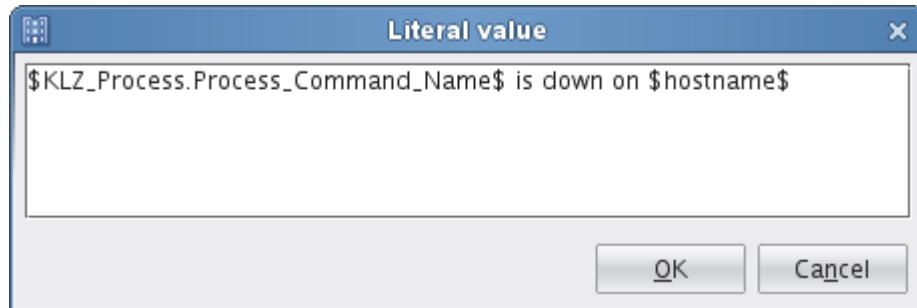
The screenshot shows the 'EIF Slot Customization - Linux\_Missing\_Process' interface. At the top, it displays the 'Event class name: ITM\_KLZ\_Process'. Below this is a table titled 'Base slots' with the following columns: Slot name, Mapped attribute, Process enumerations, Multiplier, Time format, and Literal value. The table contains the following data:

Slot name	Mapped attribute	Process enumerations	Multiplier	Time format	Literal value
adapter_host		<input type="checkbox"/>			
appl_label		<input type="checkbox"/>			
date		<input type="checkbox"/>			
fqhostname		<input type="checkbox"/>			
hostname		<input type="checkbox"/>			
msg		<input type="checkbox"/>			→
msg_catalog		<input type="checkbox"/>			
msg_index		<input type="checkbox"/>			
origin		<input type="checkbox"/>			
repeat_count		<input type="checkbox"/>			
severity		<input type="checkbox"/>			
source		<input type="checkbox"/>			

A thick black arrow points from the 'Literal value' column of the 'msg' row towards the right edge of the screen.

12. Enter the following value:

\$KLZ\_Process.Process\_Command\_Name\$ is down on \$hostname\$



13. Click **OK** several times to save the updated situation.

14. Restart the situation Linux\_Missing\_Process.

15. Examine the new events on VM04.

Netcool/OMNibus Event List : Filter="All Events"		
File	Edit	View
Alerts	Tools	
		All Events
		Default
Node	Alert Group	
VM04		mttrapd probe on VM04: Heartbeat
VM01:LZ	ITM_KLZ_Process	snmpd is down on VM01
VM01:LZ	ITM_KLZ_Process	sshd is down on VM01
VM03:LZ	ITM_StatusEvent	Monitoring situation Apache_Down

16. Notice that the slot values are updated appropriately.



## 9 Command-line interfaces exercises

### Command-line interfaces overview

The command-line interfaces (CLIs) are an alternative to using functions in the Tivoli Enterprise Portal user interface. Commands are especially useful when a function is repeated many times. You can use these commands when you deploy the same agent to multiple systems, or you can include them inside a script that is repeatedly run.

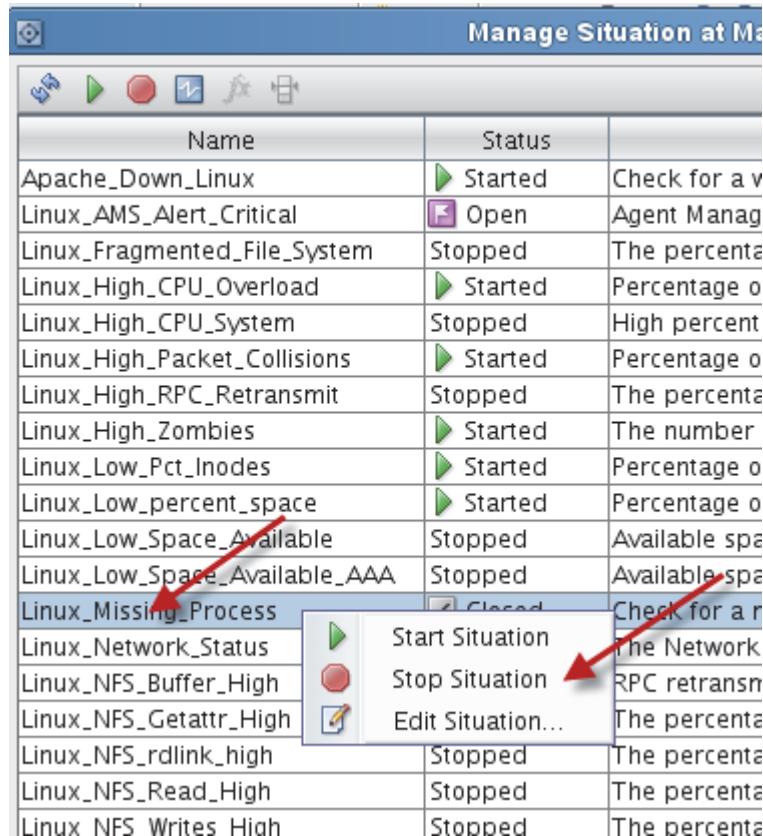
In this set of exercises, you perform the following tasks:

- Use SOAP commands
- Export and import data

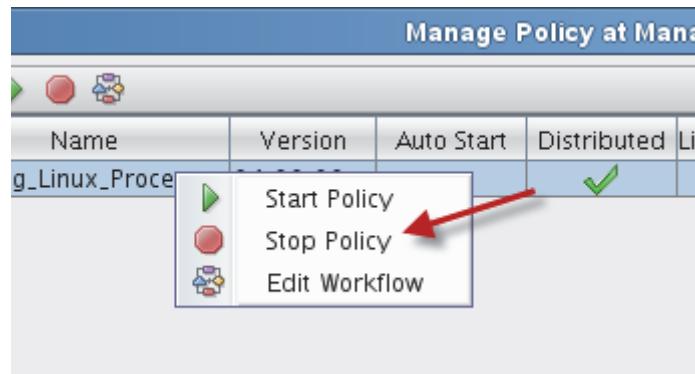
# Exercise 1. Using the SOAP command interface to work with a situation and a policy

In this example, you create an event in the Situation Event Console by using the SOAP CT\_Alert request.

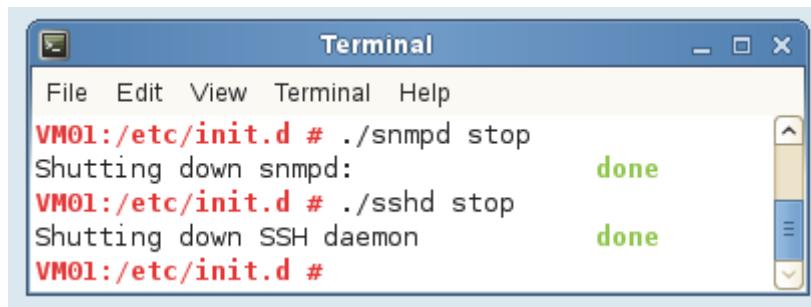
1. Open the portal interface. Click the **Enterprise** navigator and view the Situation Event Console. Throughout this exercise, messages are displayed in the Situation Event Console and are closed.
2. Stop the situation **Linux\_Missing\_Process**. You can stop this situation by right-clicking the **Linux OS** navigator found under **VM01**, selecting **Manage Situations**, highlighting **Linux\_Missing\_Process**, and clicking **Stop**.



3. Stop the policy **Starting\_Linux\_Process**. You can stop this policy by right-clicking the **Linux OS** navigator found under **VM01**, selecting **Manage Policies**, highlighting **Starting\_Linux\_Process**, and selecting **Stop Policy**.



4. If the snmpd and sshd daemons are running on VM01, close these two applications.

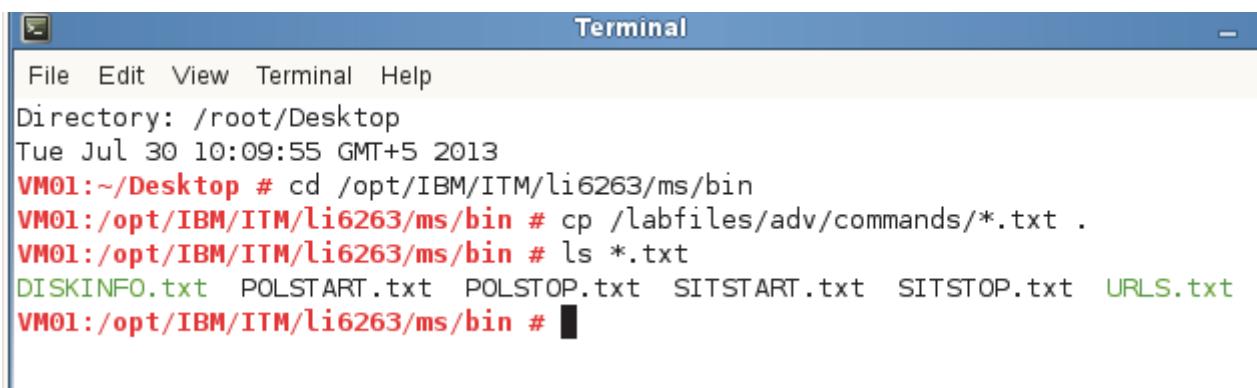


5. From a terminal window, copy the files from your exercise lab files directory to the directory where the **kshsoap** command is located. Issue the following commands (notice that the second command has a trailing period so the files are copied into the current directory):

```

cd /opt/IBM/ITM/li6263/ms/bin
cp /labfiles/adv/commands/*.txt .
ls *.txt

```



In this exercise, you issue SOAP commands that generate some events to the Situation Event Console. You can also view activity in the Message Log view.

6. From the portal interface, click the **Enterprise** Navigator and leave that workspace active during this exercise.
7. The **URLS.txt** file lists the location of the SOAP servers. View the file by issuing the following command:

```
more URLs.txt
```

```
VM01:/opt/IBM/ITM/li6263/ms/bin # more URLs.txt
http://vm01:1920///cms/soap
VM01:/opt/IBM/ITM/li6263/ms/bin #
```

8. View the **SITSTART.txt** file with the following command:

```
more SITSTART.txt
```

```
VM01:/opt/IBM/ITM/li6263/ms/bin # more SITSTART.txt
<CT_Activate>
<hub>SOAP</hub>
<name>Linux_Missing_Process</name>
<type>situation</type>
<userid>sysadmin</userid>
<password>object00</password>
</CT_Activate>
is VM01:/opt/IBM/ITM/li6263/ms/bin #
```

The file contains the following XML information for a CT\_Activate request, as listed in the following table.

<hub>	SOAP server alias
<name>	Name of a situation or policy
<type>	Situation or a policy
<userid>	Portal server defined user ID
<password>	Password for the portal server user ID

## 9 Command-line interfaces exercises

### Exercise 1. Using the SOAP command interface to work with a situation and a policy

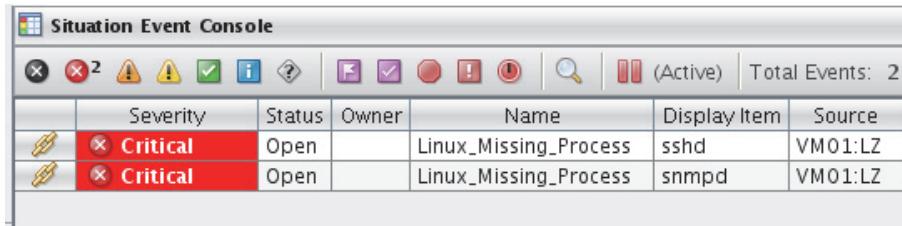
- Issue the following command to make the SOAP request to start the Linux\_Missing\_Process situation:

```
./kshsoap SITSTART.txt URLs.txt
```

View the output displayed in the terminal window. You see the SOAP XML statements and diagnostic messages. Near the bottom of the display, you see XML statements indicating the status of the request with a return code. A successful return code is zero (**0**).

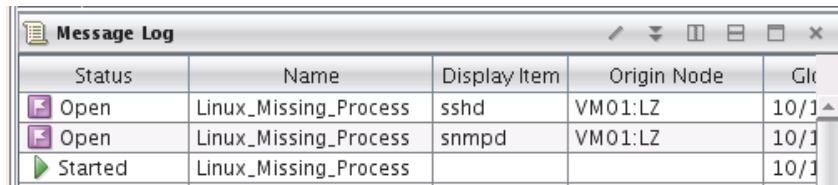
- From the portal interface, view the Situation Event Console.

You see the Linux\_Missing\_Process critical events.



Situation Event Console						
	Severity	Status	Owner	Name	Display Item	Source
	<b>Critical</b>	Open		Linux_Missing_Process	sshd	VM01:LZ
	<b>Critical</b>	Open		Linux_Missing_Process	snmpd	VM01:LZ

- From the Message Log view, you see the Linux\_Missing\_Process messages showing that it started.



Status	Name	Display Item	Origin Node	Gl
	Linux_Missing_Process	sshd	VM01:LZ	10/1
	Linux_Missing_Process	snmpd	VM01:LZ	10/1
	Started	Linux_Missing_Process		10/1

12. Issue the following command to start the policy Starting\_Linux\_Process:

```
./kshsoap POLSTART.txt URLs.txt
```

```

Terminal
File Edit View Terminal Help
dle-soap:attributes"><TABLE name="04SRV.TOBJACCL">
<OBJECT>04SRV.TOBJACCL</OBJECT>
<DATA>
<ROW>
<RC>0</RC></ROW>
</DATA>

</TABLE>
</SOAP-CHK:Success></SOAP-ENV:Body></SOAP-ENV:Envelope>
<<=====

Broadcast Message from root@VM01
(somewhere) at 12:42 ...

The process snmpd on VM01:LZ was not running and could be automated

.
.

Broadcast Message from root@VM01
(somewhere) at 12:42 ...

The process sshd on VM01:LZ was not running and could be automated

VM01:/opt/IBM/ITM/li6263/ms/bin #
```

13. Watch the Situation Event Console and the terminal window to see the Linux\_Missing\_Process events close and the ssh and snmp daemons start.

14. Issue the following command to stop the policy Starting\_Linux\_Process:

```
./kshsoap POLSTOP.txt URLs.txt
```

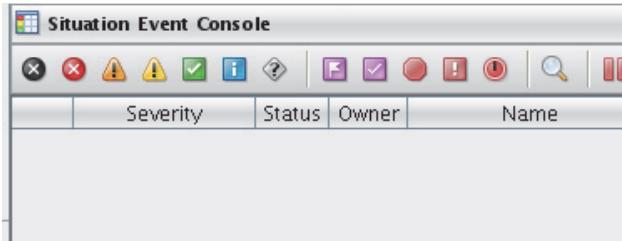
15. Close the sshd and snmpd daemons. Watch the Situation Event Console to see the Linux\_Missing\_Process events open for the sshd and snmpd daemons.

Situation Event Console					
	Severity	Status	Owner	Name	Display Item
✗	Critical	Open		Linux_Missing_Process	sshd
✗	Critical	Open		Linux_Missing_Process	snmpd

16. Issue the following command to stop the situation Linux\_Missing\_Process:

```
./kshsoap SITSTOP.txt URLs.txt
```

17. Watch the Situation Event Console to see the Linux\_Missing\_Process events close.



## Exercise 2. Using the SOAP command interface to extract attribute information

The SOAP CT\_Get request is used to request and retrieve current monitoring information that is associated with attributes for the Linux\_Disk attribute group.

- From the portal interface Physical Navigator, click **Enterprise > Linux Systems > VM01 > Linux OS > Disk Usage**. In the Disk Usage view, the Disk Name column displays the various disks associated with the mount points available on VM01.

Mount Point	Disk Name	Size (MB)	Disk Used (MB)	Disk Free (MB)	Total Inodes	Inodes Used	Inodes Free
/	/dev/sda2	38673	13905	23196	2495296	258876	2236420
/proc	proc	0	0	0	0	0	0
/sys	sysfs	0	0	0	0	0	0
/sys/kernel...	debugfs	0	0	0	0	0	0
/dev	devtmpfs	998	1	997	0	0	0
/dev/shm	tmpfs	998	1	998	216127	3	216124
/dev/pts	devpts	0	0	0	0	0	0
/sys/fs/fus...	fusectl	0	0	0	0	0	0
/proc/fs/v...	none	0	0	0	0	0	0
/proc/sys/f...	none	0	0	0	0	0	0
/root/.gvfs	gvfs-fuse-dae...	0	0	0	0	0	0

- View the **DISKINFO.txt** file.

This file contains the information that is shown in the following examples.

```
VM01:/opt/IBM/ITM/li6263/ms/bin # more DISKINFO.txt
<CT_Get>
<hub>SOAP</hub>
<userid>sysadmin</userid>
<password>object00</password>
<object>Linux_Disk</object>
<source>VM01:LZ</source>
</CT_Get>
VM01:/opt/IBM/ITM/li6263/ms/bin #
```

<CT_Get>	Name of the method
<hub>SOAP</hub>	Alias name for the monitoring server
<userid>sysadmin</userid>	User ID to log in with
<password>object00</password>	Password to the user ID
<object>Linux_Disk</object>	Name of the attribute group
<source>VM01:LZ</source>	Source managed system
</CT_Get>	Termination of the method statements

3. Issue the following command to make the SOAP request:

```
./kshsoap DISKINFO.txt URLs.txt > DISKINFO.out
```

This action places the response into a file that can be examined with gedit.

4. View the output displayed in **DISKINFO.out** by using gedit.

```
File Edit View Terminal Help
bash: DISKINFO.out: command not found
VM01:/opt/IBM/ITM/li6263/ms/bin # ./kshsoap DISKINFO.txt URLs.txt > DISKINFO.out
VM01:/opt/IBM/ITM/li6263/ms/bin # gedit DISKINFO.out
```

5. In gedit, search for the string **/dev/sda2**. Scroll down 8 lines to see which **system name** that the query was issued against. Compare the values that are displayed in the Tivoli Enterprise Portal with the values that are displayed from the SOAP call. They should be similar values.

The screenshot shows a gedit window titled "DISKINFO.out". A search dialog box is overlaid on the main text area. The search term "/dev/sda2" is entered in the "Search for:" field. Two red arrows point to the search term in the search bar and to the opening tag <Disk\_Name>/dev/sda2</Disk\_Name> in the XML output below.

```

<Disk_Name>/dev/sda2</Disk_Name>
<Size dt="number">38673</Size>
<FS_Type>ext3</FS_Type>
<Inodes_Free dt="number">2209100</Inodes_Free>
<Total_Inodes dt="number">2495296</Total_Inodes>
<Inodes_Used dt="number">286196</Inodes_Used>
<Mount_Point></Mount_Point>
<Mount_Point_U></Mount_Point_U>
<System_Name>VM01:LZ</System_Name>
<Inodes_Available_Percent dt="number">88</Inodes_Available_Percent>
<Inodes_Used_Percent dt="number">12</Inodes_Used_Percent>
<Space_Available_Percent dt="number">26</Space_Available_Percent>
<Space_Used_Percent dt="number">74</Space_Used_Percent>
<Space_Available dt="number">9891</Space_Available>
<Space_Used dt="number">27210</Space_Used>
<Timestamp>1130708124904000</Timestamp>

```

The screenshot shows the Tivoli Monitoring interface. The Navigator pane on the left displays a hierarchical tree of system components under an 'Enterprise' root. Under 'Linux Systems', nodes include 'VM01' (expanded to show 'ABK Transactions', 'Warehouse Proxy', 'Tivoli Log File Agent - VM01A:VM01:LO', 'Summarization and Pruning Agent', and 'Linux OS'), 'VM02', and 'VM04'. Under 'Windows Systems', there is one node. The 'Physical' view is selected in the Navigator toolbar. The bottom pane, titled 'Disk Usage', contains a table with the following data:

Mount Point	Disk Name	Size (MB)	Disk Used (MB)	Disk Free (MB)	Total Inodes
/	/dev/sda2	38673	27211	9891	2495296
/proc	proc	0	0	0	0

## Exercise 3. Exporting and importing infrastructure data

Many commands are available to back up and restore infrastructure data that is used for monitoring. The results of these commands can also be used to migrate information from one Tivoli Monitoring enterprise to another. These commands can be used to migrate information to other components within the same enterprise. With these commands, you can make backup copies of the following items:

- Bundles
- Situations
- Policies
- Workspaces
- Queries
- Calendar entries
- Navigators that can include all workspaces, queries, and situations associated with the Navigator

Some access privileges are required to use the **exportNavigator** and **importNavigator** commands. Verify that sysadmin has the following permissions:

- Modify permission enabled for the Custom Navigator Views object
- Modify permission enabled for the Query object
- Workspace Administration Mode and Workspace Author Mode permissions enabled for the Workspace Administration object

Do not change these permissions now. You enter changes with **tacmd** commands.

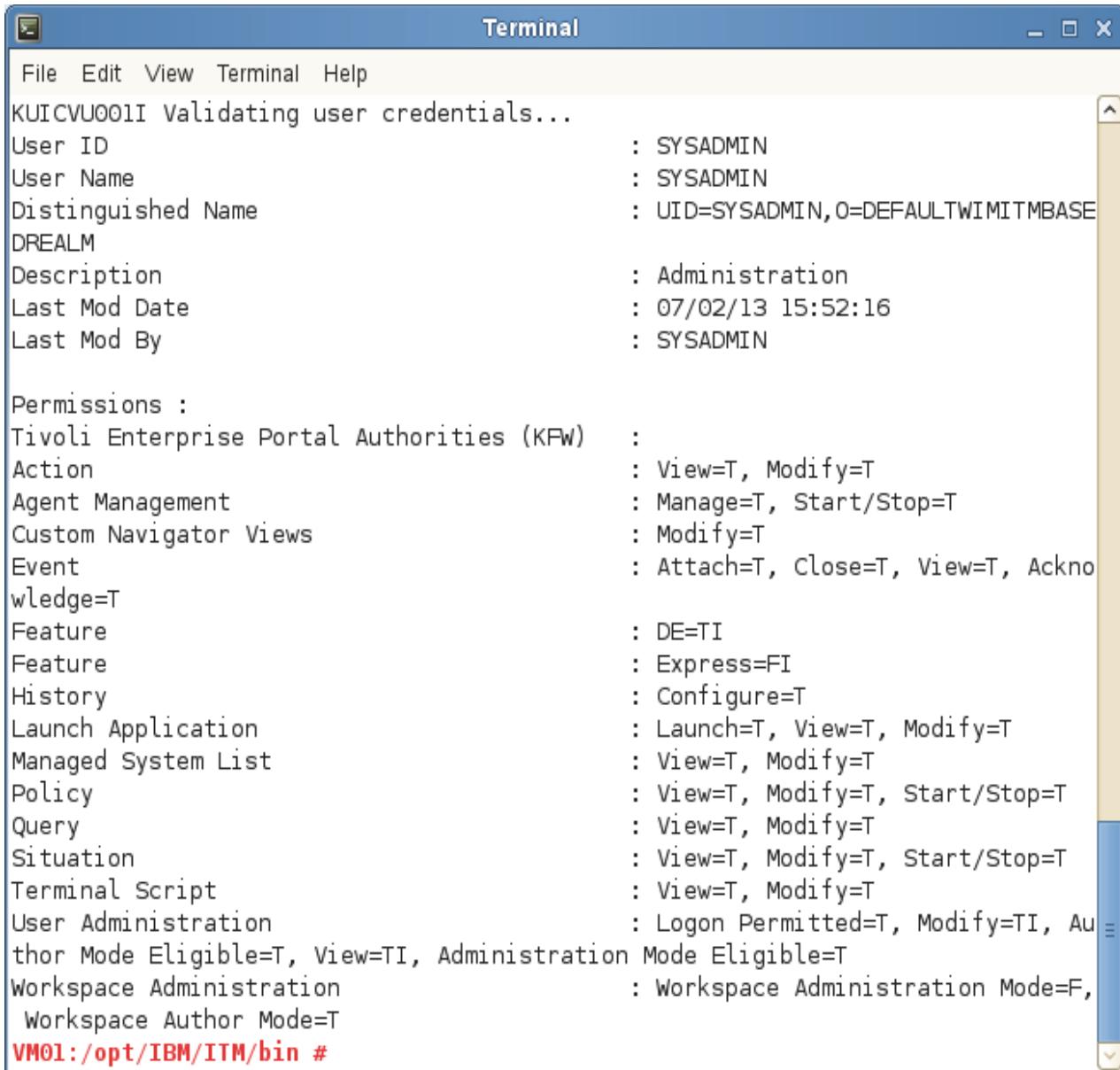
You can view the permission settings for sysadmin with this command. Specify **SYSADMIN** in the command, using uppercase instead of lowercase letters.

1. Using these commands, log in to the monitoring server and the portal server.

```
cd /opt/IBM/ITM  
. ./tacmd login -s VM01 -u SYSADMIN  
. ./tacmd tepslogin -s -VM01 -u SYSADMIN
```

2. Display the permissions of the user SYSADMIN.

```
./tacmd viewUser -i SYSADMIN -p
```



The screenshot shows a terminal window titled "Terminal". The output displays user information and a detailed list of permissions for the user "SYSADMIN".

```

File Edit View Terminal Help
KUICVU001I Validating user credentials...
User ID : SYSADMIN
User Name : SYSADMIN
Distinguished Name : UID=SYSADMIN,O=DEFAULTTWIMITMBASE
DREALM
Description : Administration
Last Mod Date : 07/02/13 15:52:16
Last Mod By : SYSADMIN

Permissions :
Tivoli Enterprise Portal Authorities (KEFW) :
Action : View=T, Modify=T
Agent Management : Manage=T, Start/Stop=T
Custom Navigator Views : Modify=T
Event : Attach=T, Close=T, View=T, Ackno
wledge=T
Feature : DE=TI
Feature : Express=FI
History : Configure=T
Launch Application : Launch=T, View=T, Modify=T
Managed System List : View=T, Modify=T
Policy : View=T, Modify=T, Start/Stop=T
Query : View=T, Modify=T
Situation : View=T, Modify=T, Start/Stop=T
Terminal Script : View=T, Modify=T
User Administration : Logon Permitted=T, Modify=TI, Au
uthor Mode Eligible=T, View=TI, Administration Mode Eligible=T
Workspace Administration : Workspace Administration Mode=F,
    Workspace Author Mode=T
VM01:/opt/IBM/ITM/bin #

```

3. Add Workspace Administration Mode by using the following command. If you have a portal interface open for sysadmin, you do not see the results of adding Workspace Administration Mode. The \*Admin Mode\* in the portal title is visible only after logging out and logging back in.

```
./tacmd edituser -i SYSADMIN -u sysadmin -w object00 -s VM01 -p  
"Permissions=Workspace Administration.Workspace Administration Mode=T"
```

```
File Edit View Terminal Help
VM01:/opt/IBM/ITM/bin # ./tacmd edituser -i SYSADMIN -u sysadmin -w object00 -s VM01 -p "Permissions=Workspace Administration.Workspace Administration Mode=T"
KUICEU001I Validating user credentials...
KUICEU001W The following dependent permissions will also be modified..
User Administration.Administration Mode Eligible
User Administration.Author Mode Eligible
Workspace Administration.Workspace Author Mode
Are you sure you want to edit the user SYSADMIN (Y/N) ?y
KUICEU002I The user SYSADMIN has been successfully edited on the TEPS located at
http://VM01:15200.
VM01:/opt/IBM/ITM/bin #
```

4. From a terminal window, verify that the AnyCorp custom Navigator exists on your system by using the following command:

```
./tacmd listnavigators
```

```
File Edit View Terminal Help
VM01:/opt/IBM/ITM/bin # ./tacmd listnavigators
KUICLN001I Validating user credentials...
KUICLN007I The following custom navigator views are available for user "SYSADM
N" on the Tivoli Enterprise Portal Server at http://VM01:15200:

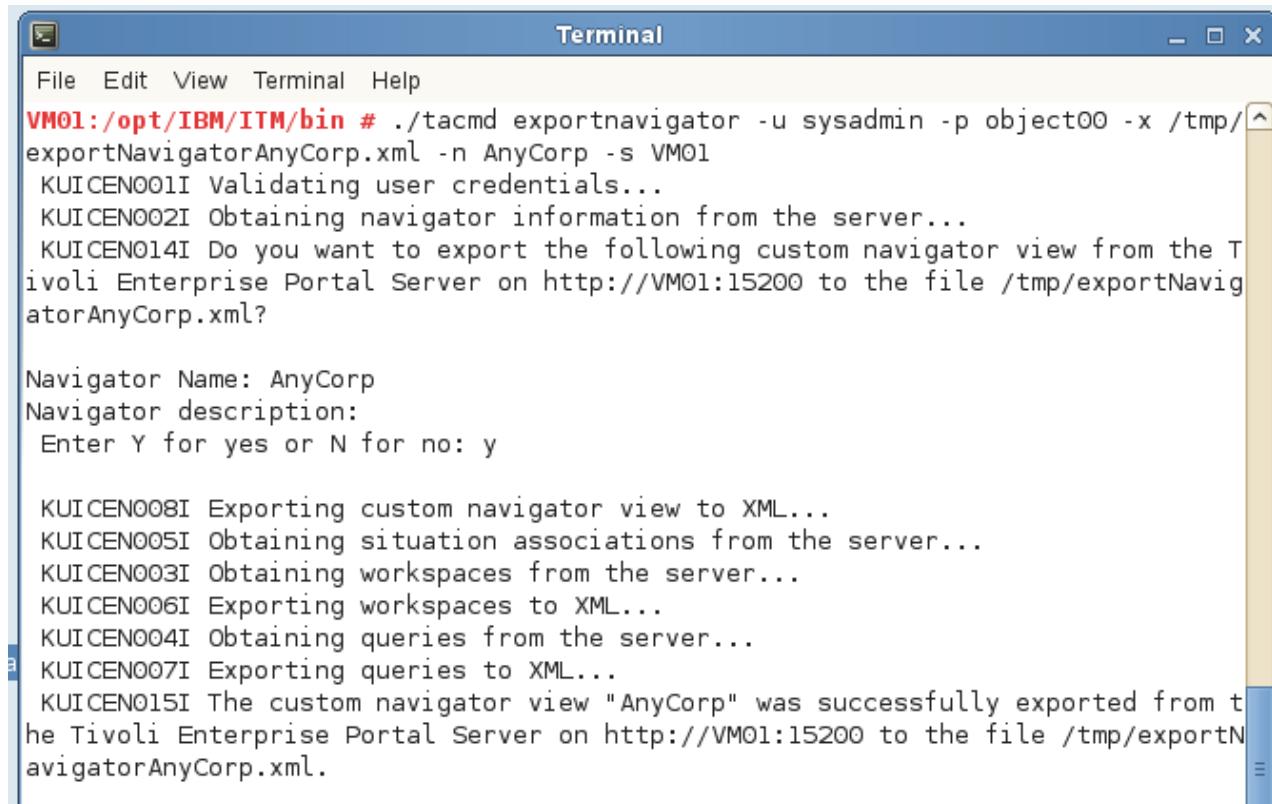
Navigator Name: Logical
Navigator description: Logical View

Navigator Name: AnyCorp
Navigator description:

Navigator Name: Operations
Navigator description:
VM01:/opt/IBM/ITM/bin #
```

5. Export the custom Navigator AnyCorp with all of its associated Navigator views, situations, workspaces, and queries with the following command:

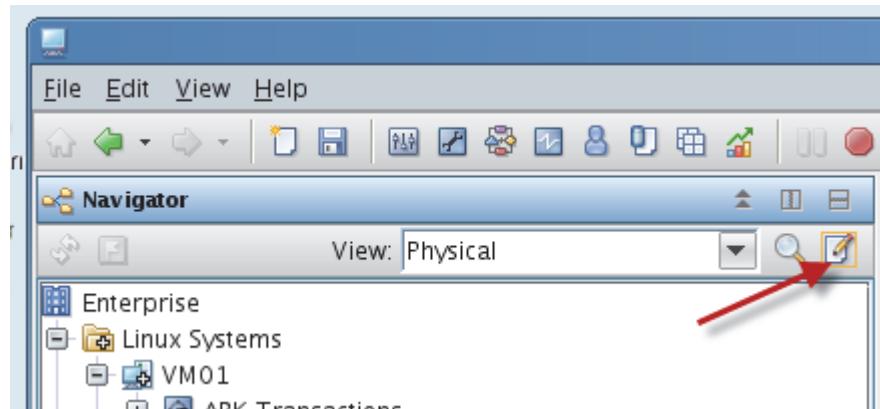
```
./tacmd exportnavigator -u sysadmin -p object00 -x  
/tmp/exportNavigatorAnyCorp.xml -n AnyCorp -s VM01
```



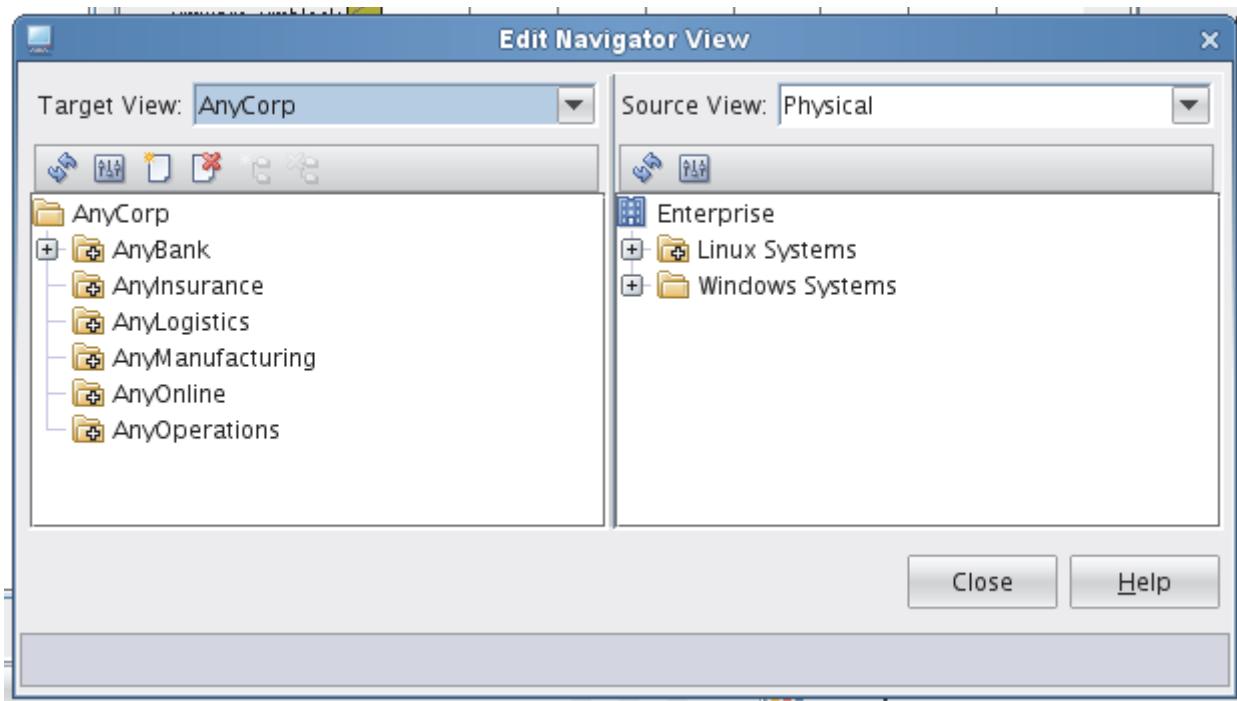
A terminal window titled "Terminal" is shown. The command `./tacmd exportnavigator -u sysadmin -p object00 -x /tmp/exportNavigatorAnyCorp.xml -n AnyCorp -s VM01` is entered. The output shows the process of validating user credentials, obtaining navigator information from the server, and prompting the user to confirm the export of the custom navigator view. The user enters "y". The process then exports the custom navigator view to XML, obtains situation associations, workspace associations, and finally exports the queries. A success message indicates the file was successfully exported.

```
VM01:/opt/IBM/ITM/bin # ./tacmd exportnavigator -u sysadmin -p object00 -x /tmp/  
exportNavigatorAnyCorp.xml -n AnyCorp -s VM01  
KUICEN001I Validating user credentials...  
KUICEN002I Obtaining navigator information from the server...  
KUICEN014I Do you want to export the following custom navigator view from the T  
ivoli Enterprise Portal Server on http://VM01:15200 to the file /tmp/exportNavi  
gatorAnyCorp.xml?  
  
Navigator Name: AnyCorp  
Navigator description:  
Enter Y for yes or N for no: y  
  
KUICEN008I Exporting custom navigator view to XML...  
KUICEN005I Obtaining situation associations from the server...  
KUICEN003I Obtaining workspaces from the server...  
KUICEN006I Exporting workspaces to XML...  
KUICEN004I Obtaining queries from the server...  
KUICEN007I Exporting queries to XML...  
KUICEN015I The custom navigator view "AnyCorp" was successfully exported from t  
he Tivoli Enterprise Portal Server on http://VM01:15200 to the file /tmp/exportN  
avigatorAnyCorp.xml.
```

6. From the portal client Navigator, click the icon for **Edit Navigator View**.

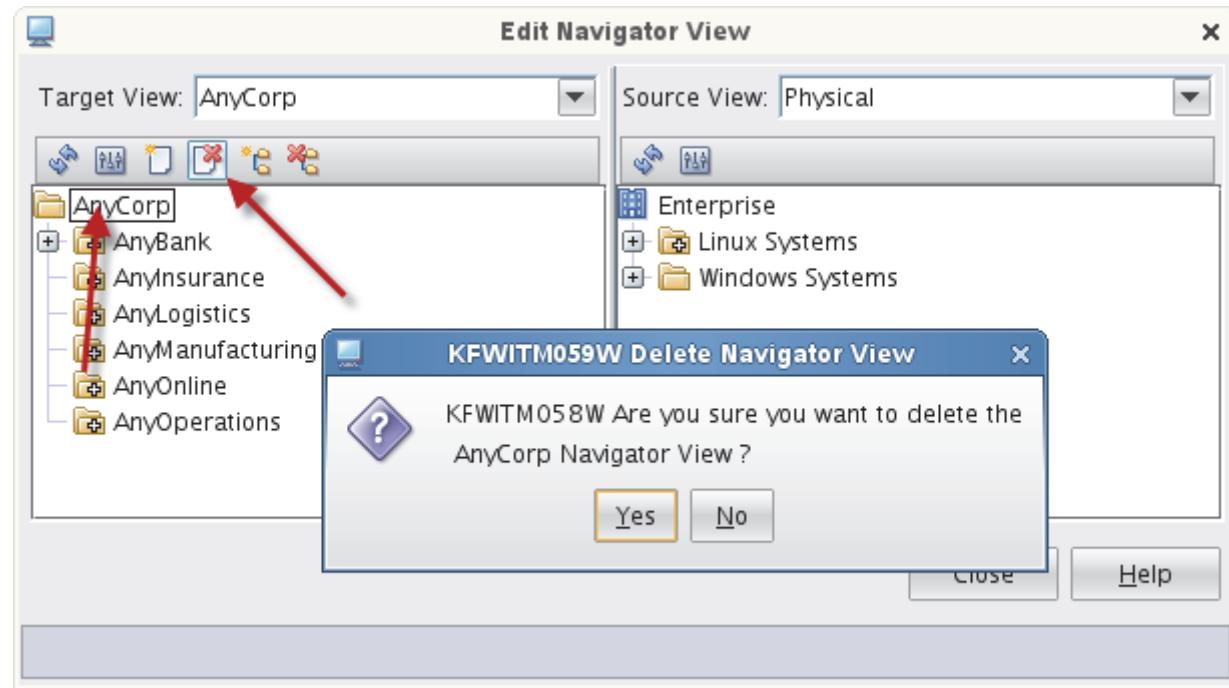


7. Select the Target View **AnyCorp**.



8. Click the **AnyCorp** Navigator in the area where the Navigator is displayed.

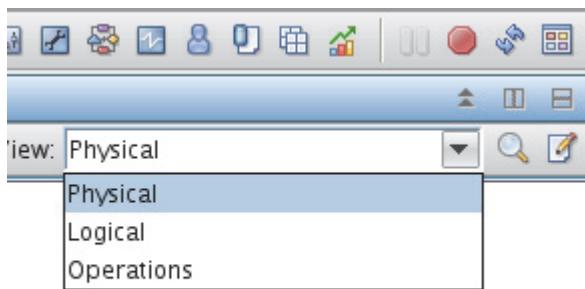
9. Click the **Delete Navigator View** icon.



10. Confirm that you want to delete the AnyCorp Navigator view.

11. Close the Edit Navigator View window.

12. Exit the portal client and log back in.
13. From the Navigator, notice that AnyCorp is no longer a choice.



You can also verify this change by using a **listNavigators** command.

```
./tacmd listNavigators
```

A screenshot of a terminal window titled 'Terminal'. The window has a menu bar with 'File', 'Edit', 'View', 'Terminal', 'Help'. The main area of the terminal shows the command `./tacmd listNavigators` being run. The output indicates that user credentials are being validated, and then lists two custom navigator views available for the user 'SYSADMN' on the Tivoli Enterprise Portal Server at `http://VM01:15200`:

```
VM01:/opt/IBM/ITM/bin # ./tacmd listnavigators
KUICLN001I Validating user credentials...
KUICLN007I The following custom navigator views are available for user "SYSADMN" on the Tivoli Enterprise Portal Server at http://VM01:15200:

Navigator Name: Logical
Navigator description: Logical View

Navigator Name: Operations
Navigator description:
VM01:/opt/IBM/ITM/bin #
```

14. From the terminal window, restore the AnyCorp Navigator and its associated components with the following command:

```
./tacmd importNavigator -x /tmp/exportNavigatorAnyCorp.xml -u sysadmin -p
object00 -s VM01
```

```
VM01:/opt/IBM/ITM/bin # ./tacmd importNavigator -x /tmp/exportNavigatorAnyCorp.xml -u sysadmin -p object00 -s vm01
KUICIN001I Validating user credentials...
KUICIN003I Reading XML and converting to custom navigator view...
KUICIN011I Do you want to import the custom navigator view "AnyCorp" to the Tivoli Enterprise Portal Server on http://vm01:15200?
Enter Y for yes or N for no: y

KUICIN004I Writing custom navigator view to the server...
KUICIN005I Writing queries to the server...
KUICIN021I The queries for the custom navigator view were successfully imported

KUICIN006I Writing workspaces to the server...
KUICIN022I The workspaces for the custom navigator view were successfully imported.
KUICIN020I The custom navigator view "AnyCorp" was successfully imported from the file /tmp/exportNavigatorAnyCorp.xml to the Tivoli Enterprise Portal Server on http://vm01:15200. The navigator view import operation did not assign the imported navigator view to any users. If the navigator view already existed, the existing user assignments were preserved. The tacmd editUser command or the Administer Users dialog can be used to assign the navigator view to a user.

VM01:/opt/IBM/ITM/bin #
```

The Navigator view import operation did not assign the imported Navigator view to any users.

You can use the **tacmd editUser** command or the Administer Users window to assign the Navigator view to a user.

15. Assign the Navigators AnyCorp, Logical, Physical, and Operations to the user sysadmin. In the command example, after the **-i** option, the entry **SYSADMIN** is case-sensitive.

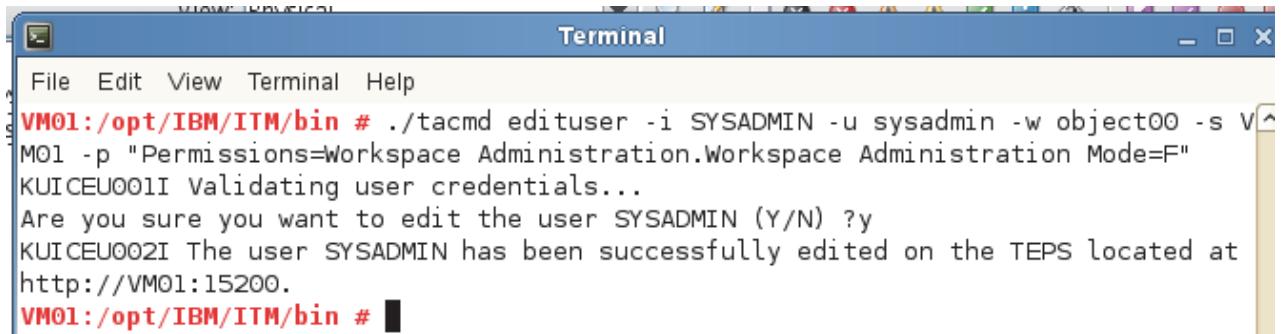
```
./tacmd edituser -i SYSADMIN -u sysadmin -w object00 -s VM01 -p
"NavigatorViews=AnyCorp,Logical,Physical,Operations"
```

```
VM01:/opt/IBM/ITM/bin # ./tacmd edituser -i SYSADMIN -u sysadmin -w object00 -s VM01 -p "NavigatorViews=AnyCorp,Logical,Physical,Operations"
KUICEU001I Validating user credentials...
Are you sure you want to edit the user SYSADMIN (Y/N) ?y
KUICEU002I The user SYSADMIN has been successfully edited on the TEPS located at http://VM01:15200.

VM01:/opt/IBM/ITM/bin #
```

16. Change the sysadmin ID to not be in admin mode with the following command:

```
./tacmd edituser -i SYSADMIN -u sysadmin -w object00 -s VM01 -p  
"Permissions=Workspace Administration.Workspace Administration Mode=F"
```



The screenshot shows a terminal window titled "Terminal". The window has a menu bar with "File", "Edit", "View", "Terminal", and "Help". The main area of the terminal displays the following command and its output:

```
VM01:/opt/IBM/ITM/bin # ./tacmd edituser -i SYSADMIN -u sysadmin -w object00 -s VM01 -p "Permissions=Workspace Administration.Workspace Administration Mode=F"  
KUICEU001I Validating user credentials...  
Are you sure you want to edit the user SYSADMIN (Y/N) ?y  
KUICEU002I The user SYSADMIN has been successfully edited on the TEPS located at  
http://VM01:15200.  
VM01:/opt/IBM/ITM/bin #
```

17. Close the portal client and log back in to see the changes made to the Navigator choices. Select the **AnyCorp** Navigator to ensure that the AnyCorp objects are restored.



# More about Cloud & Smarter Infrastructure

You can find the latest information about IBM Cloud & Smarter Infrastructure education offerings online at the following location:

[www.ibm.com/software/tivoli/education/](http://www.ibm.com/software/tivoli/education/)

Also, if you have any questions about education offerings, send an email to the appropriate alias for your region:

- Americas: [tivamedu@us.ibm.com](mailto:tivamedu@us.ibm.com)
- Asia Pacific: [tivtrainingap@au1.ibm.com](mailto:tivtrainingap@au1.ibm.com)
- EMEA: [tived@uk.ibm.com](mailto:tived@uk.ibm.com)

## Cloud & Smarter Infrastructure user groups

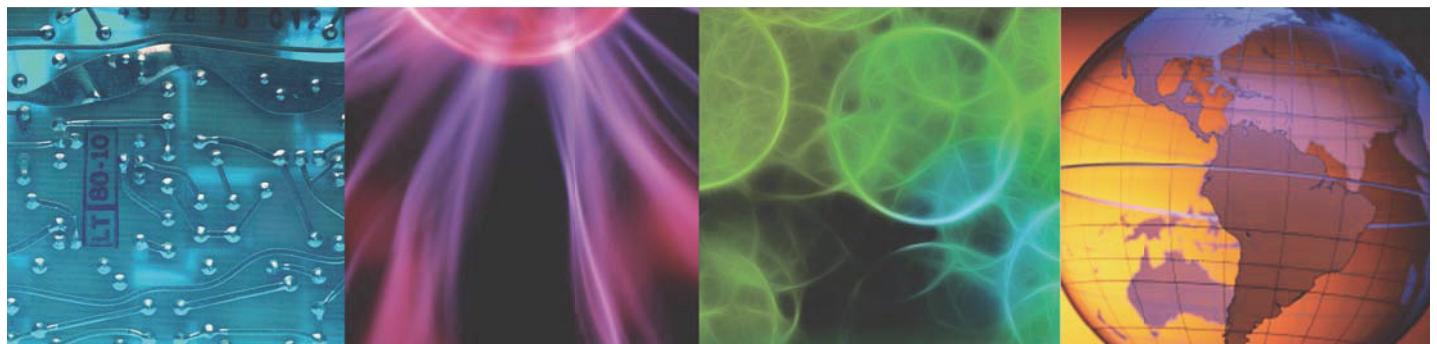
You can get even more out of Cloud & Smarter Infrastructure software by participating in one of the 91 independently run Cloud & Smarter Infrastructure user groups around the world. Learn about online and in-person user group opportunities near you at [www.tivoli-ug.org](http://www.tivoli-ug.org).

## Certification

All IBM certifications are based on job roles. They focus on a job a person must do with a product, not just the product's features and functions. Online certification paths are available to guide you through the process for achieving certification in many IBM Cloud & Smarter Infrastructure areas. See [ibm.com/tivoli/education](http://ibm.com/tivoli/education) for more information about certification.

**Special offer for having taken this course:** *Now through 31 December 2013:* For completing this course, you are entitled to a 15% discount on your next examination at any Thomson Prometric testing center worldwide. Use this special promotion code when registering online or by telephone to receive the discount: **15CSWR**. (This offer might be withdrawn. Check with the testing center.)

**IBM**<sup>®</sup>



[ibm.com/training](http://ibm.com/training)

Authorized  
**IBM** | Training





Printed in Ireland