OPTIMIZE

MERCURY BUSINESS AVAILABILITY CENTER™

IT Universe Manager Administration

MERCURY™

BUSINESS TECHNOLOGY OPTIMIZATION

Mercury Business Availability Center

IT Universe Manager Administration

Version 6.1

Document Release Date: March 5, 2006

MERCURY™

Mercury Business Availability Center, Version 6.1
IT Universe Manager Administration

This manual, and the accompanying software and other documentation, is protected by U.S. and international copyright laws, and may be used only in accordance with the accompanying license agreement. Features of the software, and of other products and services of Mercury Interactive Corporation, may be covered by one or more of the following patents: United States: 5,511,185; 5,657,438; 5,701,139; 5,870,559; 5,958,008; 5,974,572; 6,137,782; 6,138,157; 6,144,962; 6,205,122; 6,237,006; 6,341,310; 6,360,332, 6,449,739; 6,470,383; 6,477,483; 6,549,944; 6,560,564; 6,564,342; 6,587,969; 6,631,408; 6,631,411; 6,633,912; 6,694,288; 6,738,813; 6,738,933; 6,754,701; 6,792,460 and 6,810,494. Australia: 763468 and 762554. Other patents pending. All rights reserved.

Mercury, Mercury Interactive, the Mercury logo, the Mercury Interactive logo, LoadRunner, WinRunner, SiteScope and TestDirector are trademarks of Mercury Interactive Corporation and may be registered in certain jurisdictions. The absence of a trademark from this list does not constitute a waiver of Mercury's intellectual property rights concerning that trademark.

All other company, brand and product names may be trademarks or registered trademarks of their respective holders. Mercury disclaims any responsibility for specifying which marks are owned by which companies or which organizations.

Mercury provides links to external third-party Web sites to help you find supplemental information. Site content and availability may change without notice. Mercury makes no representations or warranties whatsoever as to site content or availability.

Mercury Interactive Corporation
379 North Whisman Road
Mountain View, CA 94043
Tel: (650) 603-5200
Toll Free: (800) TEST-911
Customer Support: (877) TEST-HLP
Fax: (650) 603-5300

© 2005-2006 Mercury Interactive Corporation, All rights reserved

If you have any comments or suggestions regarding this document, please send them by e-mail to documentation@mercury.com.

AMLIB_ITUAD6.1/01

Table of Contents

Welcome to IT Universe Manager Administration v
How This Guide Is Organized v
Who Should Read This Guide vi
Getting More Information vii

Chapter 1: Introduction to IT Universe Manager Administration 1
Using IT Universe Manager to Model Your IT World 1
Populating the IT Universe Model 2

Chapter 2: Getting Started with IT Universe Manager 5
Working with the IT Universe Manager Tab 5
Working with Pattern and Instance Views in the IT Universe Manager Tab 7
Service View Tree Structure 12
Configuration Item Types 13
Overview for Building a Business View 18
Creating a New View 20

Chapter 3: Working with CIs in IT Universe Manager 23
About Working with CIs in IT Universe Manager 24
Related Configuration Items Tab 24
How CIs Are Manually Added to a View 27
Editing with the Instance View Editor 28
Creating a New CI 40
Attaching Existing CIs 49
Deleting CIs 56
Working with Relationships 57

Chapter 4: Configuration Item Properties 65
Editing CI Properties 65
General Properties for CIs 66
CIT-Specific Properties 72

Chapter 5: Working in Topology Map Mode 75
About the Topology Map 76
Displaying a View in Topology Map Mode 77
Viewing Layers with a Large Number of CIs 77
Understanding Topology Map Concepts 78
Using the Toolbar Options Specific to IT Universe Manager 81
Using the Topology Map Toolbar 82
Printing the Topology Map 85
Understanding Layer Setup 86
Defining a Layer's Layout 86
Modifying Layer Layout Options 88
Understanding Layout Options 105

Chapter 6: Working with Dynamic Node Factory 109
What Is Dynamic Node Factory? 110
How Dynamic Node Factory Works 111
When to Use Dynamic Node Factory 112
Defining/Editing Properties for a Dynamic Node Factory CI 113
DNode Definition Key Property 120

Chapter 7: Example Scenario for an Instance View 133
About the Example Scenario 134
Stage 1: Create the View 134
Stage 2: Build the View 136
Stage 3: Customize KPI Settings 160
Stage 4: Define Viewing Permissions 168
Stage 5: See KPI Status in Dashboard 173

Index 175

Welcome to IT Universe Manager Administration

This guide describes how to work with the IT Universe Manager tab in CMDB Administration to build instance views and to manage the content of the IT universe model in the CMDB.

How This Guide Is Organized

The guide contains the following chapters:

Chapter 1 Introduction to IT Universe Manager Administration

Introduces the IT universe model and the concept of building views.

Chapter 2 Getting Started with IT Universe Manager

Describes the IT Universe Manager tab and the process of building views, and how to create new instance views.

Chapter 3 Working with CIs in IT Universe Manager

Describes how to build and modify views by adding CIs, and defining new CIs and relationships.

Chapter 4 Configuration Item Properties

Describes how to define and edit properties for a configuration item.

Chapter 5 Working in Topology Map Mode

Describes how to work with CIs in the IT universe model and to build instance views when working in Topology Map mode.

Chapter 6 Working with Dynamic Node Factory

Explains the concept of Dynamic Node Factory, and describes how to define and customize Dynamic Node Factory CIs.

Chapter 7 Example Scenario for an Instance View

Provides a step-by-step scenario for defining an instance view for Dashboard.

Who Should Read This Guide

This guide is intended for the following users of Mercury Business Availability Center:

- Mercury Business Availability Center administrators
- Mercury Business Availability Center platform administrators
- Mercury Business Availability Center application administrators
- Mercury Business Availability Center data collector administrators

Readers of this guide should be knowledgeable about enterprise system administration, have familiarity with ITIL concepts, and be knowledgeable about Mercury Business Availability Center in general and Mercury Application Mapping technology specifically.

Getting More Information

For information on using and updating the Mercury Business Availability Center Documentation Library, reference information on additional documentation resources, typographical conventions used in the Documentation Library, and quick reference information on deploying, administering, and using Mercury Business Availability Center, refer to *Getting Started with Mercury Business Availability Center*.

1

Introduction to IT Universe Manager Administration

This chapter introduces IT Universe Manager Administration and describes the origination of the configuration items in the IT universe model.

This chapter describes:	On page:
Using IT Universe Manager to Model Your IT World	1
Populating the IT Universe Model	2

Using IT Universe Manager to Model Your IT World

Mercury Business Availability Center enables you to build your own logical configuration in hierarchical tree format, according to how you view your organization. You can map key business processes at the top level of the tree to the relevant underlying IT infrastructure, drilling down to the leaves of the tree represent measurements collected by Business Process Monitor, SiteScope, or other monitoring tools.

The trees are contained in service views, and are built from the configuration items (CIs) in the Mercury Universal CMDB. Each CI in a tree can have one or more associated KPIs, which display different aspects of operational status for the CI when the view is deployed in the presentation layer.

Each CI and each relationship exists only once in the CMDB, as part of the IT universe model that mirrors your organization's IT world. You can add instances of the same CI to multiple service views, with a different relationship defined for the CI in each view.

The service views may be of type pattern or instance. (For an explanation of these view types, see "Instance Views and Pattern Views" in *View Manager Administration*.) You can build your own pattern views in the View Manager tab, by defining TQLs that display the hierarchy you want; however, when you require logical configurations that are tailored to the needs of your organization, you will generally build instance views in the IT Universe Manager tab.

The IT Universe Manager tab enables you to define and manage the CIs and relationships in your service views. When working in a service view in the IT Universe Manager tab, you can create new CIs, modify existing CIs, and delete CIs. All these actions have a direct affect on the whole IT universe model, not just the current view. For example, if you delete a CI in your view, that CI is deleted from the IT universe model—it will no longer appear in any of your views.

Populating the IT Universe Model

The IT universe model in the CMDB contains the CIs and relationships produced by the various CI-generating tools that operate within Mercury Business Availability Center. Most CIs and relationships originate from one of the following sources:

- the source adapters, which automatically create CIs and relationships for the monitor objects in Monitor Administration. For more information, see *Source Manager Administration*.
- the discovery process, which automatically creates CIs and relationships based on the discovered IT resources and topology in your company's network infrastructure. For more information, see *Discovery Manager Administration*.
- the Dynamic Node Factory tool, which automatically creates CIs and relationships based on data from incoming samples. For more information, see Chapter 6, "Working with Dynamic Node Factory."
- the IT Universe editor, which you use to manually create CIs and relationships that represent logical and physical elements of the business. For more information, see Chapter 3, "Working with CIs in IT Universe Manager."

- Monitor Administration, which enables you to manually create relationships between the CIs for monitor objects and CIs in the CMDB. For more information, see "Configuration Items and Monitor Objects" in *Working with Monitor Administration* and "Monitor Deployment Wizard" in *Working with SiteScope*.

  In the IT Universe Manager tab, you can build service views where you map together the CIs and relationships from these different worlds contained in the CMDB. This functionality enables you to build views that meet your business requirements and objectives, and that monitor what is important to you.

2

Getting Started with IT Universe Manager

This chapter introduces the IT Universe Manager tab and functionality, and provides explanations for some of the concepts used when building views. It also describes how to create a new instance view when working in the IT Universe Manager tab, and provides an overview for building the view.

This chapter describes:	On page:
Working with the IT Universe Manager Tab	5
Working with Pattern and Instance Views in the IT Universe Manager Tab	7
Service View Tree Structure	12
Configuration Item Types	13
Overview for Building a Business View	18
Creating a New View	20

Working with the IT Universe Manager Tab

You access the IT Universe Manager tab from **Admin** > **CMDB** > **IT Universe Manager**.

The IT Universe Manager tab is divided into two panes:

- The left pane contains View Explorer, which enables you to select and display the required service view, and to search for configuration items (CIs) in views. For general information on working with View Explorer, see "Using View Explorer" in *Working with the CMDB*.

View Explorer lists the existing views (including empty views) that are defined as applicable for IT Universe Manager and for which you have viewing permissions (for details, see "Defining View Properties to Create Views" and "Setting Permissions for Views" in *View Manager Administration*).

- The right pane displays information for the view currently open in View Explorer. The display format for this information depends on the selected viewing mode.

  The IT Universe Manager tab works in two viewing modes: **default** and **topology map**. When you first access the tab, it uses the default viewing mode. You can change to topology map viewing mode by selecting that option from the **Viewing Mode** list (top-right corner). The same functionality is available from both modes, the difference is in how the CI information is presented in the GUI:

- In **Default** viewing mode, the IT Universe Manager tab opens at the welcome page, which provides basic information on getting started, and provides links for creating new instance views (described in "Creating a New View" on page 20), and adding CIs without creating new relationships (described in "Working with CIs in IT Universe Manager" on page 23). The information and specific links presented depend on the view type (pattern or instance). For details, see "Working with Pattern and Instance Views in the IT Universe Manager Tab" on page 7.

  The welcome page is displayed whenever you are working at the highest level of the view (you can return to the welcome page at any time by clicking the view name in View Explorer in the left pane).

  In Default viewing mode, when you select a CI in View Explorer, the right pane displays information for the CI in two tabs:

  - The **Properties** tab provides information on the properties for the selected CI, and enables you to edit the properties. For details, see "Configuration Item Properties" on page 65.
  - The **Related Configuration Items** tab lists the relationships for the selected CI, and provides functionality to add additional related CIs, or to remove relationships. For details, see "Working with CIs in IT Universe Manager" on page 23.

➤ In **Topology Map** viewing mode, the right pane displays the CIs in the current view organized into single-level or multi-level topological maps. You can create new instance views, add CIs, add relationships, edit CI properties, and delete CIs and relationships, just as in default viewing mode. For details, see "Working in Topology Map Mode" on page 75.

Note: You must have **Change** permissions for a view in order to edit it. If you do not have the necessary permissions, contact your Mercury Business Availability Center Administrator.

Working with Pattern and Instance Views in the IT Universe Manager Tab

There are two types of service views: pattern and instance. Pattern views are defined in View Manager, while instance views are generally created and built in the IT Universe Manager tab. The functionality for working with the two view types in the IT Universe Manager tab differs somewhat, as described in the following sections.

For information on the different view types and how they function, see *View Manager Administration*.

Working in Instance Views

Mercury Business Availability Center provides one default, empty instance view, the First Service View. Any further instance views you create will also be empty until you manually add CIs to the view.

You can add any sort of CI to an instance view, either by adding existing CIs or by creating new CIs within the view. When you create a new CI in the view, the CI (and any associated relationship) is also created in the Mercury Universal CMDB, and will automatically be added to any other view where the new CI is relevant. For example, if the parent of the created CI appears in another instance view and is set to show its child CIs, then the new child CI will automatically be added to the parent in that view; or if the new CI fits the requirements of the TQL query in a pattern view, then it will automatically appear in the pattern view.

The welcome page for an instance view (displayed when working at the highest level of the view) provides functionality for:

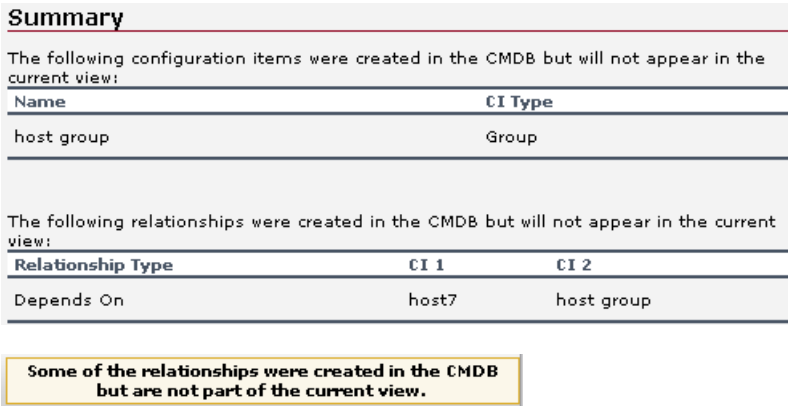
- creating new non-related CIs, as described in "Creating a New CI" on page 40.
- adding existing CIs to the view, and other edit functionality, as described in "Editing with the Instance View Editor" on page 28.
- creating new instance views, as described in "Creating a New View" on page 20.

Working in Pattern Views

CIs created by Discovery Manager or Source Manager will automatically be added to each pattern view where the CIs (and their relationships) fit the TQL definition for the view. When you open a pattern view in View Explorer, the TQL query for the view queries the CMDB for CIs that match the defined pattern. If there are no matches (for example, in the case of a view for a discovery process where the process has not yet been run) then the view will be empty.

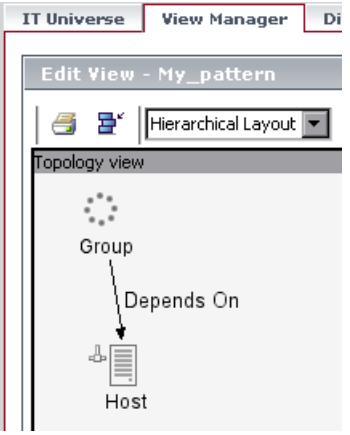
When you create a new CI while working in any view in the IT Universe Manager tab, the CI (and any associated relationship) is created in the CMDB. The CI is also added to your current view if you are working in an instance view. However, if you are working in a pattern view, the new CI (and relationship) is added to the view only if it meets the TQL specifications for the view. If the created CI and relationship do not meet the TQL requirements, then the CI and relationship will be created in the CMDB and not added to the view. Similarly, if you try to attach an existing CI to a CI in the view, and the relationship does not meet the TQL requirements, then the attached CI will not appear in the view, but the relationship is still created in the CMDB.

The New CI Wizard, or a message box in the top-right corner of the tab, will inform you that the CI and/or relationship are being created without adding them to the view.

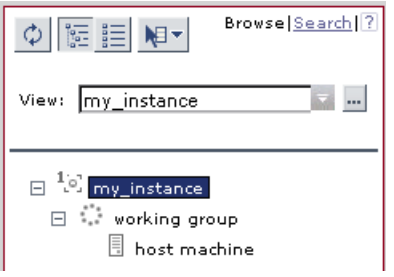


If a created CI and its relationship conform to the TQL queries defined for other pattern views, it will automatically appear in those views when the views are opened in View Explorer.

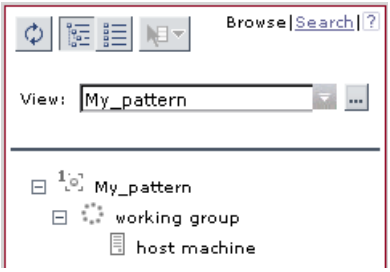
For example, a TQL defined for a pattern view in View Manager includes the nodes for **Group** and **Host** with relationship **Depends On**.



In an instance view in the IT Universe Manager tab, a Group CI, working group, is created with a **Depends On** relationship to a child CI of type host, host machine.



The new CIs and relationship fit the TQL definition in the pattern view, so they are automatically added to the pattern view.



The welcome page for a pattern view (displayed when working at the highest level of the view) enables you to:

- open the New CI Wizard to create a new non-related CI, as described in "Creating a New CI" on page 40.
- create new instance views, as described in "Creating a New View" on page 20. (You cannot create new pattern views from the IT Universe Manager tab.)

  For details on creating and working with pattern views and TQLs, see "Working with Pattern Views" in *View Manager Administration*.

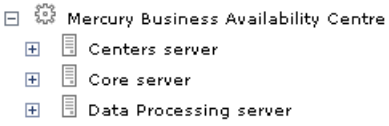
Note: Changes to pattern views take longer to update to the view than changes to instance views.

Service View Tree Structure

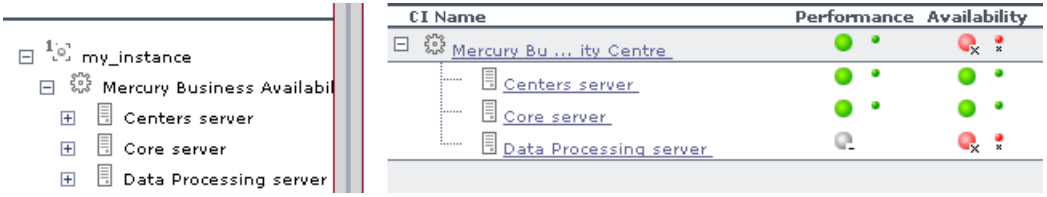
Service views are organized into an hierarchical tree format; in View Explorer, this tree structure is shown inverted. The highest node in the tree shows the view name, displayed beside the view icon. The view name is not a CI, but rather acts as the containing folder for the CIs in the view.

At the next level of the View Explorer tree are the nodes for the root CIs. A root CI is the starting point for a view subtree. A view may contain any number of subtrees. Under the root CIs come the branch CIs, expanding down to the leaf CIs as the last nodes at the ends of each branch.

The CIs at each level in a subtree are considered to be attached to the CI on the branch directly above. This attachment generally depicts a dependant relationship, so that the higher branch (closer to the root CI) represents a component of your business that depends on the CIs on the next branch down (closer to the leaf CIs) in the business process. For example, an application depends on the servers running the application.



The attachment between a higher node and lower node is referred to as a parent-child relationship. The parent CI may have many attached child CIs, and the child CIs pass operational status to the parent. The status at each level is viewed in Dashboard.



When defining an instance view, you build your own topology for the subtrees, according to the business structure or process you want represented. For more information, see “Overview for Building a Business View” on page 18.

When working with pattern views, the tree topology is determined by the pattern defined for the view. For more information on defining pattern views, see "Working with Pattern Views" in *View Manager Administration*.

Note: If required, you can manually reorder CIs within a branch in the tree. For details, see "Managing Views" in *View Manager Administration*.

Configuration Item Types

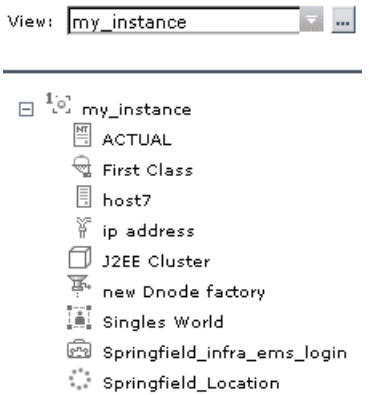
Every CI must belong to a class of CIs, called a CI type (CIT). There are four main categories of CITs:

- **Business** – CITs that correspond to the logical elements of your business, such as processes and organizational groups.
- **System** – CITs that correspond to the physical elements (hardware and software) installed in your business environment.
- **Dynamic Node Factory** – CITs that dynamically generate CIs in response to incoming data from your data sources.
- **Monitor** – CITs that handle incoming metrics collected from your business universe:
  - Monitor-type CIs under this category are added by the source adapters, as described in *Source Manager Administration*.
  - UDX Measurement Filter CIs are added when you create a measurement filter, as described in "Working with Measurement Filters" in *Platform Administration*.
  - Diagnostics Probe CIs are added when you integrate with Mercury Diagnostics, as described in *Mercury Diagnostics Installation and User's Guide*.

Monitor CIs cannot be manually created in the IT Universe Manager tab, but you can attach them to other CIs and edit some of their properties. For more information on the role of monitor CIs, see “Role of Monitor CIs in a View” on page 17.

Each CIT has a model that provides the template for creating the CI and defines the properties that are specific to that class of CIs. The CI type models are defined in the CI Type Manager, where you can edit the models, and define new CITs, if required. In the Edit Configuration Item Type window for a CIT model, you can view a description for each CIT, and view its properties. For details, see “Managing CITs” in *CI Type Manager Administration*.

The different CIT categories are represented by different icons in View Explorer, New CI Wizard, and other windows, for example:

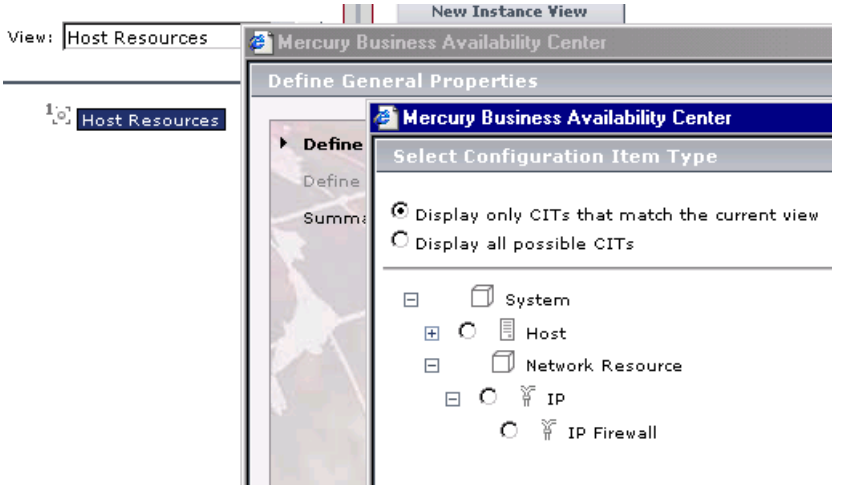


When defining a new CI using the New CI Wizard, on the first page you must select the CIT for the CI. The wizard provides a list of the CITs that are relevant at the location where you are adding the new CI. (For pattern views, you have the option to override this list and select from all possible CITs.)

The list of relevant CITs depends on:

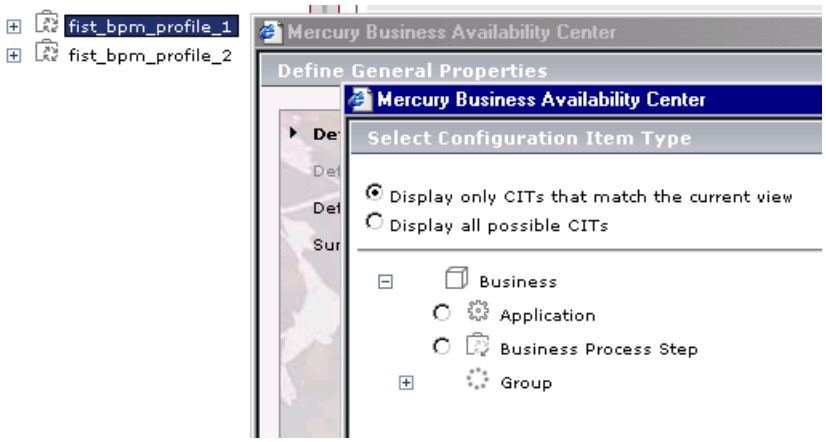
- **the view definition** – The TQLs for pattern views define which CITs and relationships can be added to the view.

  For example, when creating a new CI in a Host Resources (pattern) view, only the System CITs for the **IP** CIT group and for the **Host** CIT group, are relevant for the view.

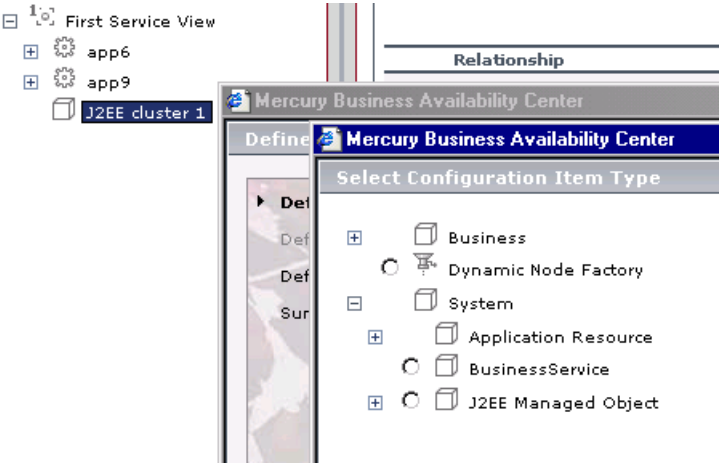


For more information on CIs in pattern views, see “Working in Pattern Views” on page 8.

- **the definition for the selected CI** – The CI Type Manager defines which other CITs can be logically related to each CIT. For example:
  - If the selected CI is of type **Business Process Group**, the CIT list includes relevant CITs under the **Business** category, but excludes the **System** category. (System CITs would not be logically added to a business process group.)



  - If the selected CI is of type **J2EE Cluster**, the CIT list includes (under the System category) only the **J2EE Managed Object** CIT group, and certain relevant **Application Resource** CITs.



Most System CITs and the Dynamic Node Factory CIT have properties that are specific to the type (some with default values). When creating a new CI for one of these CITs, you define the values for these properties on the Define CIT-Specific Properties page of the New CI Wizard. Some of the properties are mandatory.

For CIs created by the source adapters or by the discovery process, the required CIT-specific properties are defined automatically. Some of these properties can be edited in the **Other Properties** area of the IT Universe Manager Properties tab.

Role of Monitor CIs in a View

In the Mercury Business Availability Center Dashboard and Service Level Management applications, the CIs display colored indicators (KPIs) to show operational status information for the components of your IT universe. To give status color to a KPI, it needs to have a dependency on one or more of the leaf CIs that receive actual monitoring information, for example, a SiteScope monitor or Business Process Monitor transaction. (For information on KPIs, see "Configuring KPIs" in *Application Administration*.)

The dependency can be direct (in which case the parent CI has a **monitored by** relationship with the monitor CI child), or through the branches of a subtree. For example, a logical CI for location could receive KPI status through either one of the following hierarchies:

- **Location CI -> monitor leaf CI**
- **Location CI -> group CI -> group CI -> monitor leaf CI**

Mercury Business Availability Center automatically creates a dependency connection to a monitor CI for all CIs added using the source templates in the Source Manager, or added using the Dynamic Node Factory template. However, CIs added by the discovery process, or that you create manually, are not automatically associated with monitor CIs; you must manually define the appropriate connections.

You can manually create a monitor dependency for a CI in one of the following ways:

- By attaching a monitor CI (or a subtree containing a monitor CI) to the parent CI, in the IT Universe Manager tab. For details, see "Attaching Existing CIs" on page 49.
- By making an association between a monitor object in the Monitor Administration tree and an existing CI. The monitor CI created for the monitor object will automatically be attached to the selected CI. For details, see "Configuration Items and Monitor Objects" in *Working with Monitor Administration*.
- By using Monitor What Matters to create a new SiteScope monitor for a specific CI. The created monitor will generate a new monitor CI in the CMDB, with an automatic attachment to the selected CI. For details, see "Monitor Deployment Wizard" in *Working with SiteScope*.

Overview for Building a Business View

You use the IT Universe Manager tab to build and manage the CI hierarchy of the service views. These views are used in Dashboard to monitor the business environment. The service views should provide the business users using Dashboard with logical representations of your company's business structure, processes, and goals.

To meet these requirements, you will generally build multiple instance views representing different aspects of your business; for example, you may build business views that revolve around:

- company structure, so that data is organized by the geographical location of branches, or by data centers, or so forth.
- business processes, so that data is organized by line of business, or by applications, or so forth.
- business goals, so that data is organized by sales, or by key customers, or so forth.

Of course, the views you create will depend on the particular demands of your organization, and the actual hierarchical structure of the views will reflect those demands.

Whatever aspect of the business a view is intended to represent, the most usual hierarchical format is as follows:

- **Top level** – the top-most levels of the view, consisting of the root CI and the branches below it, will normally be built from logical **Business** CI types such as **Location**, **Line of Business**, **Customer**, **Application**, and so forth.
- **Intermediate level** – the intermediate branches of the view will normally consist of: **System** CI types, representing actual hardware and software; and/or monitoring group CI types, representing groups and profiles containing monitors.
- **Bottom level** – the bottom-most level of the view, consisting of the leaf CIs at the end of the subtrees, is built from the **Monitor** CI types such as **BPM Transaction**, **SiteScope Measurement**, and so forth. The leaf CIs in a view should all be of type **Monitor**; these CIs receive the actual data from the data sources, used to determine KPI status all the way up the view subtrees, and so provide business context to the view.

In some cases, you may choose to use the **Dynamic Node Factory** CI type to dynamically build a hierarchy for you. For details, see Chapter 6, "Working with Dynamic Node Factory."

Summarized Steps for Creating and Building Your Instance View

The following gives a summary of the steps for creating a typical business view.

A step-by-step process chart for creating views is provided in "How Do I Reflect My IT Environment in Mercury Business Availability Center?" in *Getting Started with Mercury Business Availability Center*.

A full example scenario for creating a business view is provided in Chapter 7, "Example Scenario for an Instance View."

Steps to create and build an instance view:

1. Create a new instance view. For details see "Creating a New View" on page 20.
2. Create new (non-related) CIs to form the roots of your view subtrees. For details, see "Creating a Non-Related CI" on page 40.
3. For the intermediate branches of the subtrees, create new related CIs (for details, see "Creating a Related CI" on page 41), or attach existing CIs (for details, see "Attaching Existing CIs" on page 49). Also see "Editing with the Instance View Editor" on page 28.
4. For the leaf CIs of the subtree, attach monitoring CIs. For details, see "Attaching Existing CIs" on page 49.
5. Define user permissions for the view. For details, see "Setting Permissions for Views" in *View Manager Administration*.

Creating a New View

From the IT Universe Manager tab you can create a new instance view. There are two stages to the process of creating a new view: First you define the properties to create the view framework, as described in the procedure below; then you build the view by adding CIs, as described in Chapter 3, "Working with CIs in IT Universe Manager."

You can also create new instance views and new pattern views in View Manager. For details, see "Defining View Properties to Create Views" in *View Manager Administration*.

You can select to hide the view in Dashboard and Service Level Management applications (if, for example, you do not want the view to be seen by the application users until you have finished building it).

Note: After creating the view, if you want to then modify the view properties, you must do so from View Manager. For details, see "Working with View Manager" in *View Manager Administration*.

To create an instance view from the IT Universe Manager tab:

1 In the **IT Universe Manager** tab, click the **New Instance View** button 

at the top of the tab. (When working in an instance view, you also have the option of selecting the view name and clicking the **Create a new empty instance view** link in the right pane.)

Note: The view will be created in the same folder as the view currently displayed in View Explorer. If required, you can later move the view to a different folder in the View Manager tab.

---

2 Define the properties for the view. For details, see "Defining View Properties to Create Views" in *View Manager Administration*.

Note: If you deselect the **IT Universe** check box, then after you close the Create View - View Properties window, the view will not appear in the **View** list in the IT Universe Manager tab.

---

3 Click **OK**. View Explorer opens the new view in the IT Universe Manager tab, ready to add CIs.

3

Working with CIs in IT Universe Manager

This chapter describes how to use the IT Universe Manager tab to manage configuration items (CIs) and relationships in your service views, and to build views.

This chapter describes:	On page:
About Working with CIs in IT Universe Manager	24
Related Configuration Items Tab	24
How CIs Are Manually Added to a View	27
Editing with the Instance View Editor	28
Creating a New CI	40
Attaching Existing CIs	49
Deleting CIs	56
Working with Relationships	57

About Working with CIs in IT Universe Manager

The topics in this chapter describes the functionality available from the **Related Configuration Items** tab when working in default viewing mode. However, the same CI functionality is available when working in topology map viewing mode (as described in "Working in Topology Map Mode" on page 75).

Related Configuration Items Tab

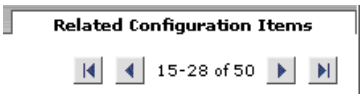
The Related Configuration Items tab is displayed in the IT Universe Manager tab by clicking any CI in View Explorer and selecting **Related Configuration Items** in the right-hand pane.



The relationship table in the tab lists the relationships for the CI currently selected in View Explorer. The table only includes the relationships that exist in the current view for the selected CI; relationships for the CI that exist in the Mercury Universal CMDB but are not part of the CI definition in the view are not shown.

(You can view a list of the relationships that exist for a CI in all views, by right-clicking the CI in View Explorer and selecting **Show Related CIs**. For more information, see "Show Related Configuration Items" in *Working with the CMDB*.)

If the relationship list is very long, the entries are divided up between pages; in this case, the page scroll arrows in the top-right corner of the tab are enabled.



The relationship table contains the following columns:

- **Relationship** – Describes the type of relationship between the CI selected in View Explorer and the related CI. For more information, see "Working with Relationships" on page 57.
- **Direction** - Describes the direction of the relationship between the two CIs. The selected CI in View Explorer is to the left of the arrow, the related CI is to the right. For example, when the selected CI in View Explorer is **Parent group**, the relationship to **Child group** is shown in the table as follows:

Relationship	Direction	CI Name
Depends On	→	Child group

The arrow shows that you read the relationship from left to right, meaning that Parent group depends on Child group.

When the selected CI in View Explorer is **Child group**, the relationship to **Parent group** is shown in the table as follows:

Relationship	Direction	CI Name
Depends On	←	Parent group

The arrow shows that you read the relationship from right to left, meaning, again, that Parent group depends on Child group.

Tip: Place the cursor over the direction arrow to see a tooltip with a descriptive summary of the relationship.

- **CI Name** – Displays the name of the CI related to the selected CI.
- **CI Type** – Displays the configuration item type (CIT) for the related CI.
-  **Edit Relationship** button – Opens the Relationship Properties window, where you can edit certain properties for the relationship. For details, see "Editing Relationship Properties" on page 59.

Tip: You can sort the information displayed in the table by clicking on a column header. You can switch the column between ascending and descending order by clicking the header again.

The buttons under the relationship table provide the following functionality:

- **Delete Relationship** – Enables you to delete relationships from the view and CMDB. For details, see "Deleting a Relationship" on page 62.
- **New Related CI** – Enables you to create a new CI that is related to the selected CI in View Explorer. For details, see "Creating a New CI" on page 40.
- **Attach Related CIs/Attach Monitor CIs** – Enables you to attach existing CIs to the selected CI in View Explorer. For details, see "Attaching Existing CIs" on page 49.

How CIs Are Manually Added to a View

There are a number of ways in the IT Universe Manager tab for managing your view content. One of the main difference between the various methods lies in whether or not new entities are created and added to the CMDB.

The functionality provided by each method is as follows:

- **Create new CIs without creating relationships** – You can create new CIs that are added at the root level of the view, so that no new relationships are created. For details, see "Creating a Non-Related CI" on page 40.
- **Create new CIs and create relationships for them** – You can create new CIs and attach them to existing CIs in the view, so creating a new CI and a new relationship in the CMDB. For details, see "Creating a Related CI" on page 41.
- **Add existing CIs and create relationships for them** – You can copy CIs and CI subtrees from another view to the current view, and attach them to an existing CI in the view, so creating new relationships for existing CIs. For details, see "Attaching Existing CIs" on page 49.
- **Add existing CIs and create new CIs** – The Instance View Editor provides a single edit tool from which you can perform all the above operations, and acts as a workspace for trying out changes to your views before the changes are updated to the CMDB. For details, see "Editing with the Instance View Editor" on page 28.
- **Add existing CIs without creating anything new** – For instance views, View Manager provides "quick edit" functionality to copy CIs and CI subtrees from another view to the instance view. This method does not create any new CIs or relationships. For details, see "Working with Instance Views" in *View Manager Administration*.

Editing with the Instance View Editor

The Instance View Editor enables you to add and remove CIs in your instance views. The changes made to a view using the Instance View Editor are not applied in the system until you choose to save them; this enables you to build your view more quickly than with other IT Universe Manager editing tools, and puts less weight on the system during the view building process.

This section includes the following topics:

- "Accessing the Instance View Editor" on page 28
- "Understanding the Instance View Editor" on page 30
- "Adding Existing CIs to the View Using the Instance View Editor" on page 33
- "Additional Functionality in the Instance View Editor" on page 36

Accessing the Instance View Editor

The Instance View Editor is used only for instance views. If you do not have modifying permissions for the view, then the **Edit View** button to open the editor is disabled.

For information on understanding Instance View Editor functionality, see "Understanding the Instance View Editor" on page 30.

Note: While you are working in the Instance View Editor, other users can still make changes to the view, which may affect your work as follows:

- Changes made by other users are displayed in the view after clicking **Save**, or after closing the Instance View Editor.
- Changes made by other users may override your changes, or vice versa.
- Changes made by other users (for example, deleting CIs) may cause the **Save** operation to fail.

To access the Instance View Editor:

1 In the **IT Universe Manager** tab, display in View Explorer the instance view you want to edit.

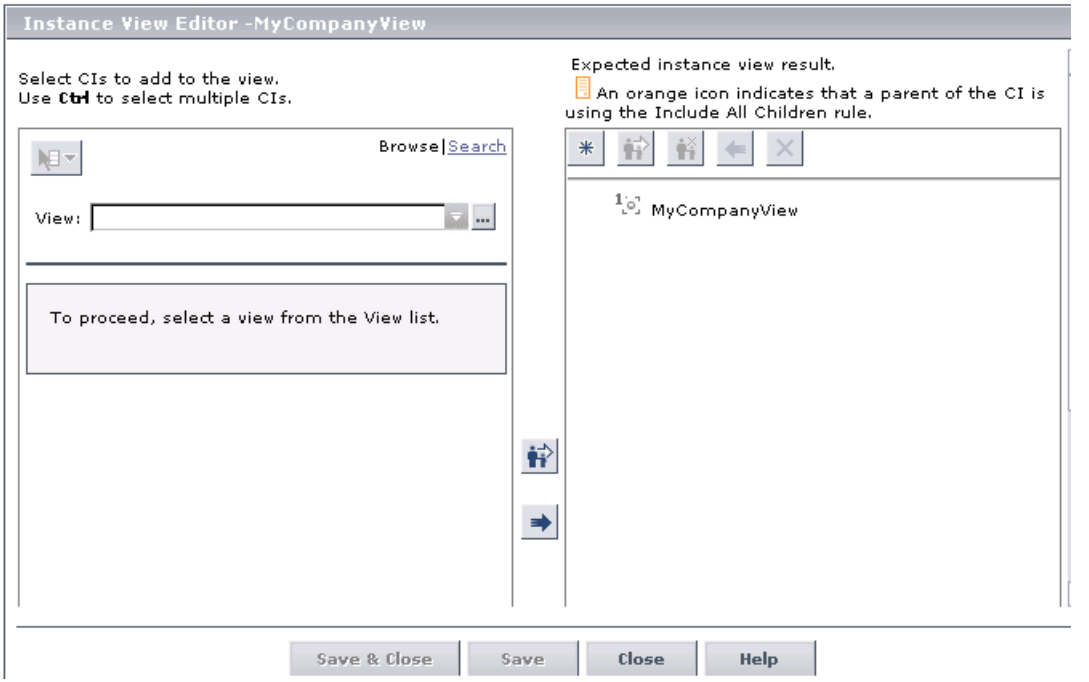
2 In the right pane, click **Edit View**.



The Instance View Editor opens, displaying the name of view to be edited in the title bar.

Understanding the Instance View Editor

The Instance View Editor contains two panes. In the left pane, you can traverse the views that exist in Mercury Business Availability Center, and view information for CIs, such as CI properties. In the right pane, you build and modify the contents of your view.



This section includes the following topics:

- "Left Pane Functionality" on page 30
- "Right Pane Functionality" on page 31
- "Child Rule Functionality" on page 32

Left Pane Functionality

You use the left pane to traverse the views in the system and select CIs, when you want to add CIs from other views to the view being edited. For more information, see "Adding Existing CIs to the View Using the Instance View Editor" on page 33.

The left pane uses View Explorer functionality for searching and viewing additional information for CIs. For more information, see "Using View Explorer" in *Working with the CMDB*.

Right Pane Functionality

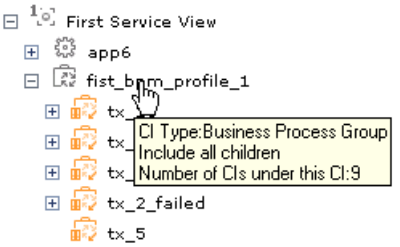
The right pane provides a snapshot of the view being edited, and functions as a workspace for making changes to the view. No changes are updated to the CMDB until you click **Save**. If you make a mistake, you can click **Close** without saving and your changes will not be implemented.

Note: If you navigate away from IT Universe Manager tab without clicking **Save**, your changes are not updated to the CMDB.

Various edit functionality is provided for the CIs in the right pane, using the right-click menu options or the toolbar buttons. For more information, see "Additional Functionality in the Instance View Editor" on page 36.

Each CI in the right pane has a tooltip showing:

- the CI type, for example, Business Process Group.
- the child rule in effect for the CI—displays **Include all children** if the CI is set to include all child CIs, or **Exclude all children** if the CI is dynamically included in the view because it is under a parent set to include all child CIs, or if the CI is dynamically included in the view because it is under a parent set to include all child CIs. For more information, see "Child Rule Functionality" on page 32.
- the number of CIs included in the subtree under the CI in the current view.



Child Rule Functionality

The child rule assigned to a CI determines what child CIs are displayed for the CI in the view:

- A CI defined as **include all child CIs** recursively displays in the view all child CIs in its subtree. The subtree is dynamically updated with every added child CI, whether it is added within the current view or in a different view.

  In the Instance View Editor, all child CIs under a CI with include all child CIs status, are displayed with orange icons. These CIs are all marked in their tooltips as **Exclude all children** (in fact, their child CIs are included automatically in the displayed subtree) and there is no option to change their child rule status.

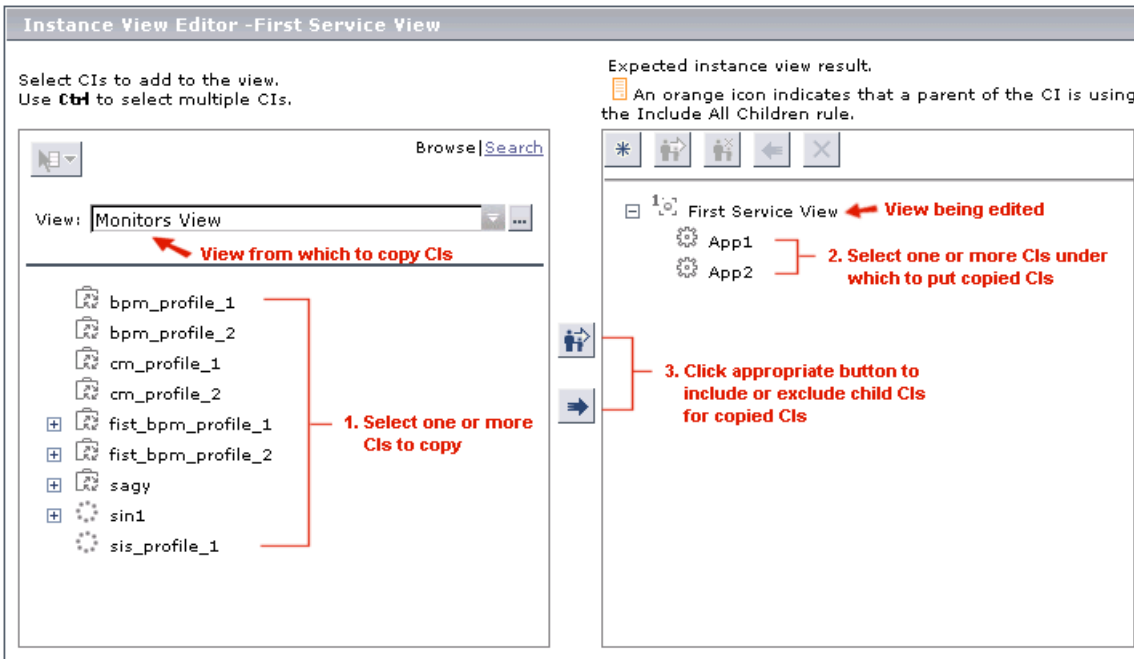
- A CI defined as **exclude all child CIs** does not automatically display the child CIs in its subtree. It may display some child CIs, for example, child CIs that were created or added to the parent CI within the current view, or CIs that have additional instances elsewhere in the view.

  If the status of a CI is changed to exclude all child CIs, then all or most of its child CIs are hidden in the view (unless they have instances elsewhere in the view). They can be restored by changing the status back to include all child CIs. However, a new created CI in the view that has not yet been saved is not restored after being hidden, and is never created in the CMDB.

  For more information on the child rules in IT Universe Manager, see "Include Child CIs" on page 70.

Adding Existing CIs to the View Using the Instance View Editor

The Instance View Editor enables you to add any CIs in the CMDB to your view, by copying them from other views in the system.



You perform the following operations to add existing CIs to your view:

1. Select one CI or multiple CIs in the left pane to add to your view.
2. Select one CI or multiple CIs in the right pane. The CIs currently selected in the left pane will be added under each selected CI in the right pane.
3. Add the currently selected CIs to the view with either:



Include All Child CIs status (by clicking the recursive arrow button)

Or:



Exclude All Child CIs status (by clicking the single arrow button

For an explanation of the significance of each status, see “Child Rule Functionality” on page 32.

When a CI is added to your view under another CI, a relationship is automatically created between the two CIs, using the default relationship for the two CI types. This will generally be a **depends on** relationship (for example, between two logical CIs) or a **monitored by** relationship (between a logical CI and a monitor CI). If there is no valid relationship between the two CIs (for example, between two monitor CIs), then a message is displayed saying that no valid link was found between the two items.

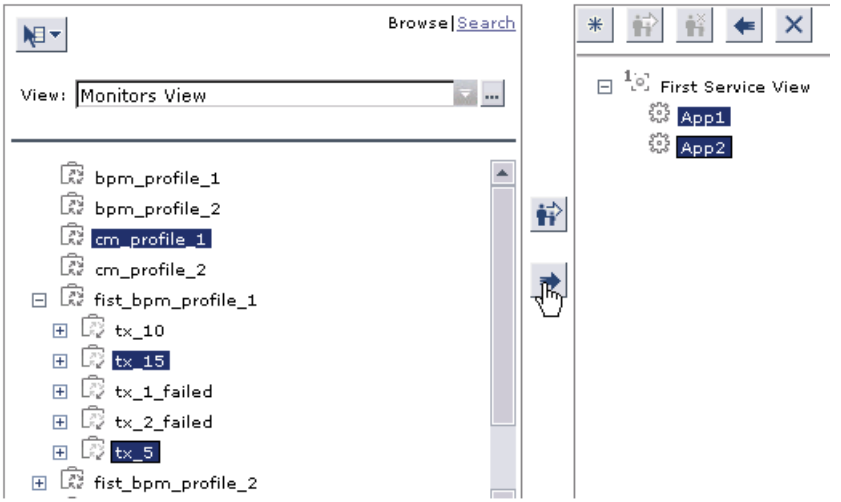
To add existing CIs to the edited view:

1. In the left pane of the Instance View Editor, select the required view from the **View** list (this is the view containing the CIs you want to add).
2. In the left pane, expand the tree and select the CI that you want to add. Use the CTRL key to select multiple CIs. You can select multiple CIs at different levels of the tree.
3. In the right pane, select the CI that you want to add the CIs under. You can select multiple CIs using the CTRL key—the added CIs will be repeated under each selected CI.

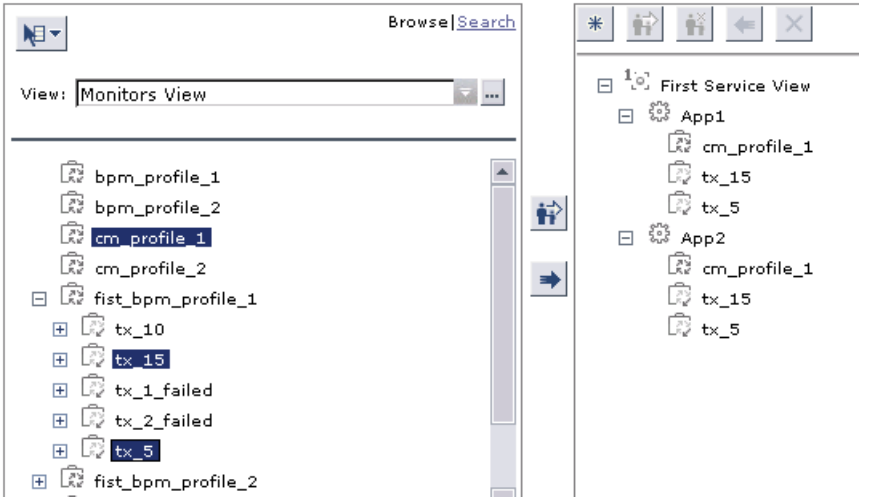
Note: If you do not select a CI in the right pane, then the CIs are automatically added at the root level in the view.

---

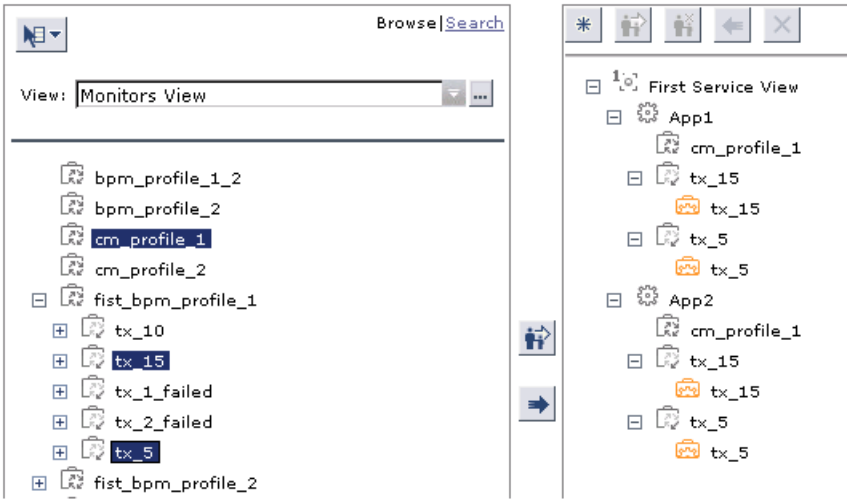
4 Click the appropriate button to add the CIs to the view, according to whether you want each added CI to include or exclude all child CIs. (For more information on the include/exclude statuses, see "Editing with the Instance View Editor" on page 28.)



The added CIs are displayed in the right pane under the selected CIs:



If you selected to add the CIs with include all child CIs status, then the child CIs (with orange icons) are also displayed in the right pane, under the parent CIs:



5 Click **Save** or **Save & Close** to update the changes to the CMDB.

Additional Functionality in the Instance View Editor

The following functionality is available from the right pane of the Instance View Editor:

- Create a new CI in your view and in the CMDB, as described in "Creating CIs" on page 37.
- Change the include/exclude all child CIs rule for a CI, as described in "Changing the Include/Exclude Child CIs Status" on page 37.
- Disconnect a CI from the parent CI, so removing the relationship between the two CIs in the IT universe, as described in "Disconnect CI from Parent" on page 38.
- Delete a CI from the CMDB, as described in "Delete CIs from the CMDB" on page 39.

Creating CIs

You can create new CIs from the Instance View Editor, at the root level of your view without a relationship, or related to one of the CIs in your view.

To create a new CI in the Instance View Editor:

1 In the right pane of the Instance View Editor, decide under which CI you want to add the new CI, then do one of the following:

- Right-click the CI and select **Create New CI** from the displayed menu.
- Select the CI and click the **Create New CI** button.



2 The New CI Wizard opens. Define the information for the new CI. For details, see "Using the New CI Wizard" on page 42.

3 After creating the CI using the New CI Wizard, the CI is added to the right pane, at the appropriate point in the view. If the new CI was created related to another CI, it is shown in the view as a child of that CI, even if the parent CI is set to exclude all child CIs status. If the parent is set to include all child CIs status, then the new CI is added with an orange icon. (For more information, see "Editing with the Instance View Editor" on page 28.)

4 Click **Save** or **Save & Close** to update the changes to the CMDB. The new CI is not physically created in the CMDB and added to the view until you save your changes in the Instance View Editor.

Changing the Include/Exclude Child CIs Status

You can toggle the status of some CIs in your view between include all child CIs and exclude all child CIs.

- If you change the status of a CI from **exclude all child CIs** to **include all child CIs**, then all child CIs that exist in the CMDB under the parent CI are added to the view (only added after saving the view). All child CIs have the status of exclude all child CIs, and their status cannot be changed.
- If you change the status of a CI from **include all child CIs** to **exclude all child CIs**, then all child CIs in the subtree under the parent CI are hidden. If the subtree includes a CI created during the current Instance View Editor session and not yet saved, then that CI is removed completely and will never be created in the CMDB.

For more information on the Include/Exclude Child CIs statuses, see "Child Rule Functionality" on page 32.

To change the include/exclude child CIs status for a CI:

1 In the right pane of the Instance View Editor, use one of the following methods to change the status:

- Right-click the required CI and select **Include All Child CIs** or **Exclude All Child CIs**, as relevant, from the displayed menu.



- Select the CI and click the select **Include All Child CIs** button, or **Exclude All Child CIs** button, as relevant.

Note: One or both of these menu options and buttons are disabled when the status is not currently applicable for the CI.

---

2 Click **Save** or **Save & Close** to update the changes to the CMDB. You may have to close the Instance View Editor to see the changes to the view.

Disconnect CI from Parent

You can remove the relationship between a CI and its parent CI (or remove a root CI from the view) by using the **Disconnect from Parent CI** option. Using this option removes the relationship in the CMDB, so the CI will no longer appear as a child of the parent CI in any view.

Any child CIs in the subtree under the disconnected CI are also removed from the view, (without affecting their relationship with the disconnected CI). If there are other instances of the disconnected CI (or any of its children) anywhere else in the view, these other instances are affected, they remain in the view.

If a CI created during the current Instance View Editor session and not yet saved, is removed by the **Disconnect from Parent CI** option, then that CI is removed completely and will never be created in the CMDB.

To disconnect a CI from its parent CI:

1 In the right pane of the Instance View Editor, use one of the following methods to disconnect CIs from parent CIs:

- Right-click a single CI and select **Disconnect from Parent CI** from the displayed menu.
- Select a single CI, or multiple CIs (using the CTRL key), and click the  **Disconnect from Parent CI** button.

The relevant CIs are removed from the subtree.

2 Click **Save** or **Save & Close** to update the changes to the CMDB.

Delete CIs from the CMDB

From the Instance View Editor, you can delete single or multiple CIs from the CMDB, so that the CIs are deleted from all views. When you delete a CI, the subtree under the CI is removed from the view.

Note: The subtree under the deleted CI is not removed from views other than the one being edited—in other views, the subtree moves to the root level of the view.

To delete a CI from the CMDB:

1 In the right pane of the Instance View Editor, use one of the following methods to delete a CI:

- Right-click a single CI and select **Delete CI from the world** from the displayed menu.
- Select a single CI, or multiple CIs (using the CTRL key), and click the  **Delete CI from the world** button.

The relevant CIs are removed from the subtree.

2 Click **Save** or **Save & Close** to update the changes to the CMDB.

Creating a New CI

In the IT Universe Manager tab, you can manually create new CIs. All new CIs are added to the CMDB; they are also appear in the currently open service view, if:

- it is an instance view.
- it is a pattern view, and the CI, together with the relationship defined for the CI, are appropriate for the pattern view definition.

This section contains the following topics:

- "Creating a Non-Related CI" on page 40 discusses the creation of a new CI without any current relationship to another CI.
- "Creating a Related CI" on page 41 discusses the creation of a new CI together with its relationship to an existing CI.
- "Using the New CI Wizard" on page 42 describes the procedure for creating a new CI (related or non-related).

Creating a Non-Related CI

When you build a new subtree within your view, you generally start by creating a root CI representing a high-level logical element in your business. You can add a root CI when working at the top level of the view (when you first access the view, or when the view name is selected in View Explorer), by clicking the **Create New CIs** link in the welcome page.

The **Create New CIs** link opens the New CI Wizard, where you define the properties for the new CI, as described in "Using the New CI Wizard" on page 42. The wizard creates the CI in the CMDB and adds the created CI as a root CI in your view (if you are working in a pattern view, the CI is added to the view only if it meets the pattern specifications).

A CI created in this way is added as an independent entity without any current relationship to any other CI existing in the CMDB. After creating the first CI, you can then create or add related CIs below the first CI.

For example, if you are building an instance view to show your lines of business, you may add root CIs representing different LOBs within your organizational structure.



Each Line of Business CI in the example is an independent entity, with no relations, ready to become the root CI for a subtree. You then attach CIs to build the subtree.

To build the branches of the subtree, you can attach new CIs that you create, or you can attach CIs that already exist in the CMDB. Each CI you attach is immediately related to the first CI:

- For details on creating new CIs under the CIs in the view, see "Creating a Related CI" on page 41.
- For details on attaching existing CIs, see "Attaching Existing CIs" on page 49.

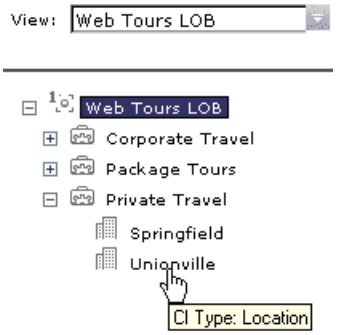
Creating a Related CI

You can create a new CI to be added to your view as a child of an existing CI, using the New CI Wizard. Both the new CI and the relationship for it are created in the CMDB.

You first select the CI in your view to which you want to attach the new CI, then open the New CI Wizard. In the New CI Wizard you define the properties and relationship for the CI, as described in "Using the New CI Wizard" on page 42. The wizard creates the CI in the CMDB, and adds the new CI as a related CI in your view (if you are working in a pattern view, the CI is added to the view only if it meets the pattern specifications).

After creating the new child CI, it will also appear for other instances of the parent CI, wherever the subtree is set to use the include all child CIs rule. For information on the relationships between CIs, see "Working with Relationships" on page 57.

For example, if you are building an instance view to show your lines of business, then under your Line of Business CIs at the root of your subtrees you may want to add additional logical CIs representing different locations dealing with that business line within your organizational structure. Each location is added as a child of the Line of Business CI, so that the Line of Business CI depends on the Location CI.

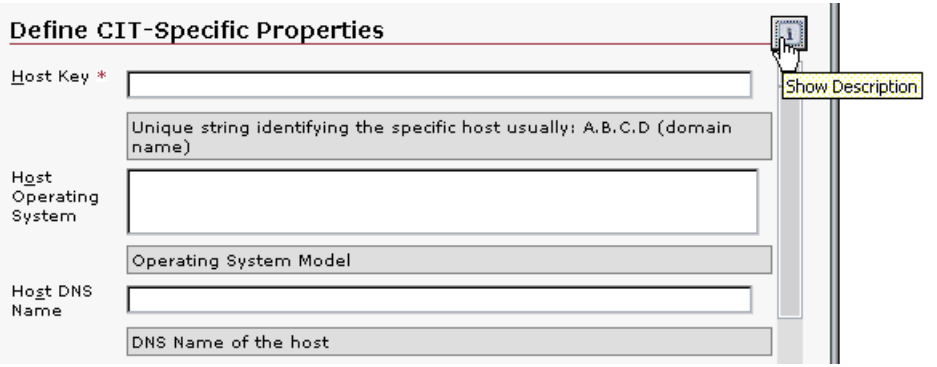


You can continue to build the branches of each LOB subtree by creating more related CIs to attach under the locations, and/or by attaching CIs or whole CI subtrees that already exist in the CMDB. For details on attaching existing CIs and CI subtrees, see "Attaching Existing CIs" on page 49.

Using the New CI Wizard

The New CI Wizard is used to define both non-related CIs (described in "Creating a Non-Related CI" on page 40) and related CIs (described in "Creating a Related CI" on page 41). The wizard content is dynamically updated during the definition process, according to your selections.

Some of the wizard pages contain a **Show Description** button in the top-right corner of the page. Click this button to display additional information for the CI properties, for example, a description of the property, or an alternative name used for the property. Click the button again to hide the descriptions.

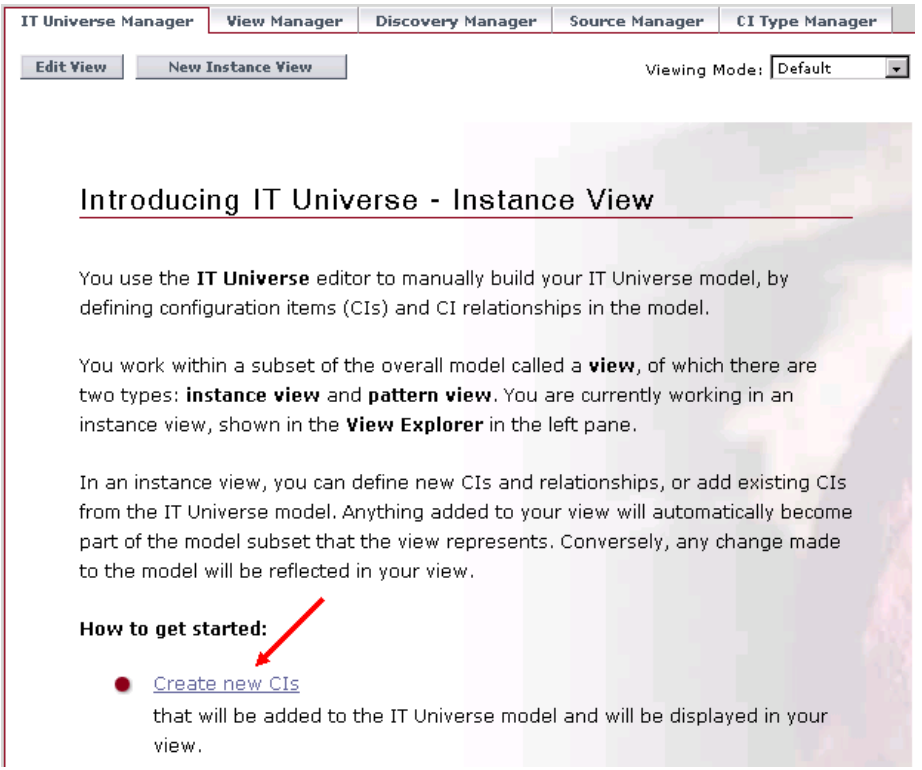


To create a new CI:

1. Open the **Admin** > **CMDB** > **IT Universe Manager** tab.
2. In View Explorer (left pane), access the view in which you want to create the CI.

3 Click the appropriate button/link as follows:

➤ If you want to create a CI that is not related to an existing CI, select the view name in the view tree and click **Create New CIs**.



➤ If you want to create a CI that will be the parent or child of an existing CI, select the existing CI in the view tree and click **New Related CI**.



Tip: You can also select the required option from the right-click menu for a CI.

The **New CI Wizard** opens at the Define General Properties page, where you select the CIT category for the new CI, and define the properties associated with every CI, such as name, geographical location, and appropriate context menu.

4 Select the CI type for the new CI. The **CI Type** property is context sensitive, permitting you to select only the CITs that are appropriate for the CI or view under which you are creating the new CI. You can select the CIT:

- from the **CI Type** dropdown list, which contains an abbreviated list of the most commonly used CITs, as well as CITs that you have previously used (as appropriate for the context).
- from the Select Configuration Item Type window, opened by clicking the button beside the **CI Type** box. The Select Configuration Item Type window contains a list of all available CITs that are appropriate for the context. You can toggle the display between **CI Types Tree** (which organizes the CITs by category) and **CI Types Table** (which displays an alphabetical list of all available CITs).

  For pattern views, you have the option of toggling between the **Display only CITs that match the current view** list, and the **Display all possible CITs** list, where you can view all available CITs (not only the CITs that match the context). Select the appropriate radio button for the list you require. When you define a CI that does not match the context, it is created in the CMDB, but not added to the current view.

For more information on the available CITs, see "Configuration Item Types" on page 13.

5 Define the other general properties for the CI. Mandatory properties are marked with an asterisk. For information on the properties, see "General Properties for CIs" on page 66.

Note the following:

- ➤ Some CIs, for example **Group**, have a default context menu defined automatically. You can change the context menu, or define new ones, by clicking **Edit Menu List**. For details, see "Editing the Menu List" on page 68.
- ➤ CIs added to instance views generally include the **Include Related CIs** option:
  - If the **Include Related CIs** check box is selected, the CI is set to include all child CIs, and all CIs in the subtree for the CI are included in the view.
  - If the check box is not selected, the CI is set to exclude all child CIs, and the child CIs are not automatically displayed in the view. However, the view might include children for the CI (for example, for relationships that are created within the current view).

The **Include Related CIs** option may be non-editable. For more information on this option, see "Include Child CIs" on page 70.

Proceed as follows:

- ➤ If you are creating a non-related CI that does not require CIT-specific properties, then the **Finish** button is enabled. Click it and skip to step 8.
- ➤ If you are creating a CI that requires CIT-specific properties, the **Next** button is enabled. Click the button and continue with the next step.
- ➤ If you are creating a related CI that does not require CIT-specific properties, then the **Next** and **Finish** buttons are enabled. Continue according to the option you require:
  - Click **Next** to continue to the Define CI Relationship page, and skip to step 7.
  - Click **Finish** if you do not want to define relationship properties. In this case, IT Universe Manager applies the default relationship properties for the CI you are creating. Skip to step 8.

6 Some CITs require the definition of properties specific to that CIT category, in which case the wizard displays the Define CIT-Specific Properties page (for information, see "CIT-Specific Properties" on page 72).

Mandatory properties are marked with an asterisk.

Enter the required information in the property fields and click **Next** to define relationship properties, or click **Finish** and skip to step 8.

7 If you are defining a related CI, then the wizard displays the Define Configuration Item Relationship page. This page enables you to define the relationship between the new CI and the selected CI in the view, and the properties for the relationship. For more information on relationships, see "Working with Relationships" on page 57.

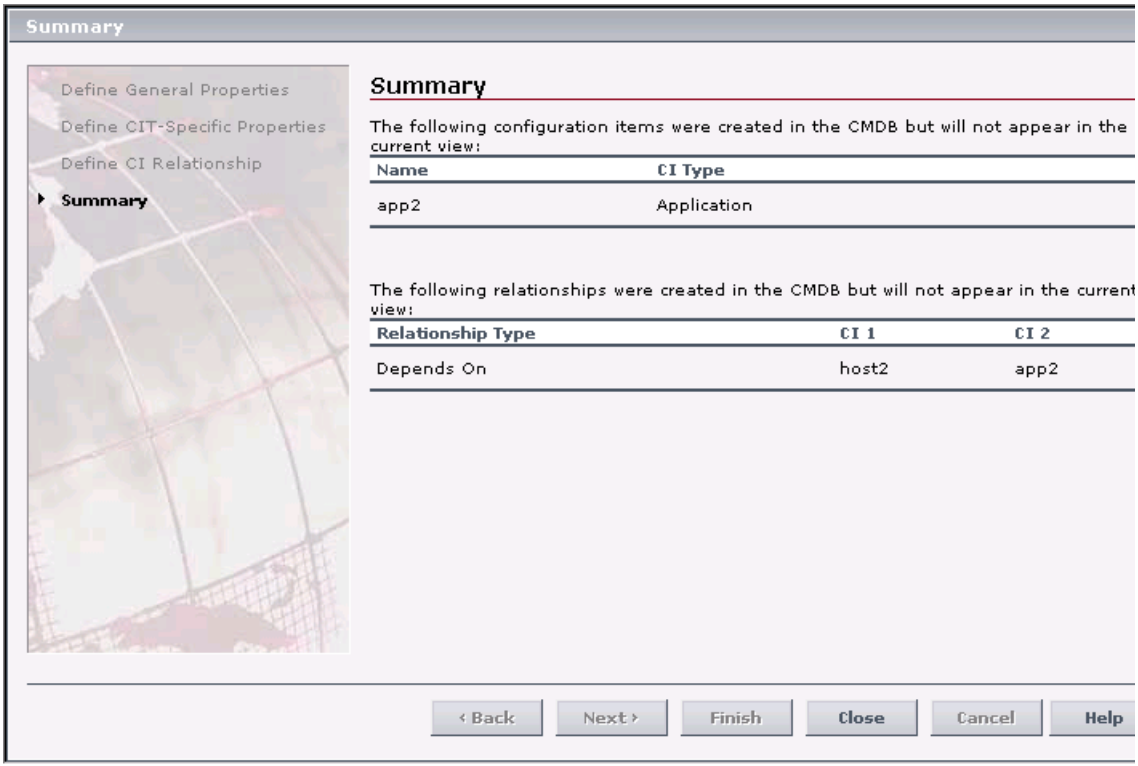
The **Relationship Definition** area of the page contains the following properties:

- **CI 1** – Displays the name of the CI on the first side of the relationship. This is the CI that you selected in the view.
- **CI 2** – Displays the name of the CI on the second side of the relationship. This is then new CI that you are creating.
- **Relationship Type** – Enables you to define the type of relationship that will be created between the two CIs. For most relationships, only the **Depends On** option is available. Select an option from the list.

The **Relationship Properties** area of the page contains properties that are applied to the relationship, such as the **Must** and **Weight** attributes. For information on the properties, see "Relationship Property Descriptions" on page 60.

Enter the required information and click **Finish**.

8 The new CI and relationship are created in the CMDB and the created CI is added at the specified point in the view tree, if the created CI is appropriate for the current view. If you are working in a pattern view, and the CI or relationship are not appropriate for the TQL definition for the view, then the Summary page is displayed, stating that the CI was not added to the view.





Tip: If the CI does not appear in the view, try refreshing the view using the **Refresh** button in View Explorer.

Attaching Existing CIs

In the IT Universe Manager tab, you can manually attach existing CIs in the CMDB to another existing CI that appears in the current view. When you attach a CI to another CI, you create a new relationship between them in the CMDB and in the view (where applicable). For information on the relationships between CIs, see "Working with Relationships" on page 57.

The IT Universe Manager tab provides two options for creating relationships between existing CIs:

- **Attach Related CIs** – This option enables you to create a relationship between the selected CI and one or more CIs in any view.
- **Attach Monitor CIs** – This option provides a shortcut for you to create relationships between the selected CI and one or more CIs in the Monitors View. This is useful when adding the monitor CIs at the bottom of your hierarchies.

You attach existing CIs (for both the above options) using the Attach Related CIs wizard. The wizard has two main stages:

- Selecting the CIs that you want to attach.
- Defining the relationship between the original CI and the CIs to attach. (The information required at this stage is the same as in the Relationship Properties window that opens when you draw a relationship line between two CIs in a view, when working in topology map viewing mode.) For information on relationships, see "Working with Relationships" on page 57.

When selecting the CIs to attach, you can select multiple CIs from a view to attach to the original CI. However, all CIs that you want to attach in one operation must have the same relationship types. For example, you cannot attach a CI that uses the relationship **Depends On** in the same operation as a CI that uses the relationship **Monitored By**.

When attaching CIs to a parent CI in an instance view, you can choose to add the CIs set to include all child CIs, by using the **Add CIs and Their Children** button. For more information, see "Include Child CIs" on page 70.

When working in pattern views, the **Add CIs and Their Children** button is hidden. Both the added CI and its child CIs appear in the view only if the CI(s) and the relationship(s) conform to the requirements of the TQL for the view. For more information, see "Working in Pattern Views" on page 8.

Note: When copying a CI from a discovery process pattern view to an instance view, if you choose to include the child CIs, be aware that you may be adding hundreds of hidden child CIs that do not appear in the pattern view.

To attach existing CIs:

1. In the **IT Universe Manager** tab, access the required view in View Explorer.
2. In the view tree, select the CI for which you want to create a relationship.
3. In the right pane (**Related Configuration Items** tab), click **Attach Related CIs** to attach any CI, or **Attach Monitor CIs** if you want to attach a monitoring CI.

Tip: You can also select the required option from the right-click menu for the CI in View Explorer.

The Attach Related CIs wizard opens at the Select Related CIs page, which contains two panes: The left pane enables you to select the view from which you want to copy CIs; the right pane displays the CIs that you add to the view.

The left pane of the page uses View Explorer functionality. For information on the search capabilities of View Explorer, see "Using View Explorer" in *Working with the CMDB*.

4. If required, select a different view from the **View** list in the left pane of the Select Related CIs page.

5 In the view tree, select the check boxes for the CI(s) that you want to attach.



Tip: To make your selections, you can also use the buttons at the bottom of the page for **Select All**, **Clear All**, and **Invert Selection**.

---

6 Click one of the arrow buttons between the panes to move the selected CIs to the list in the right pane:



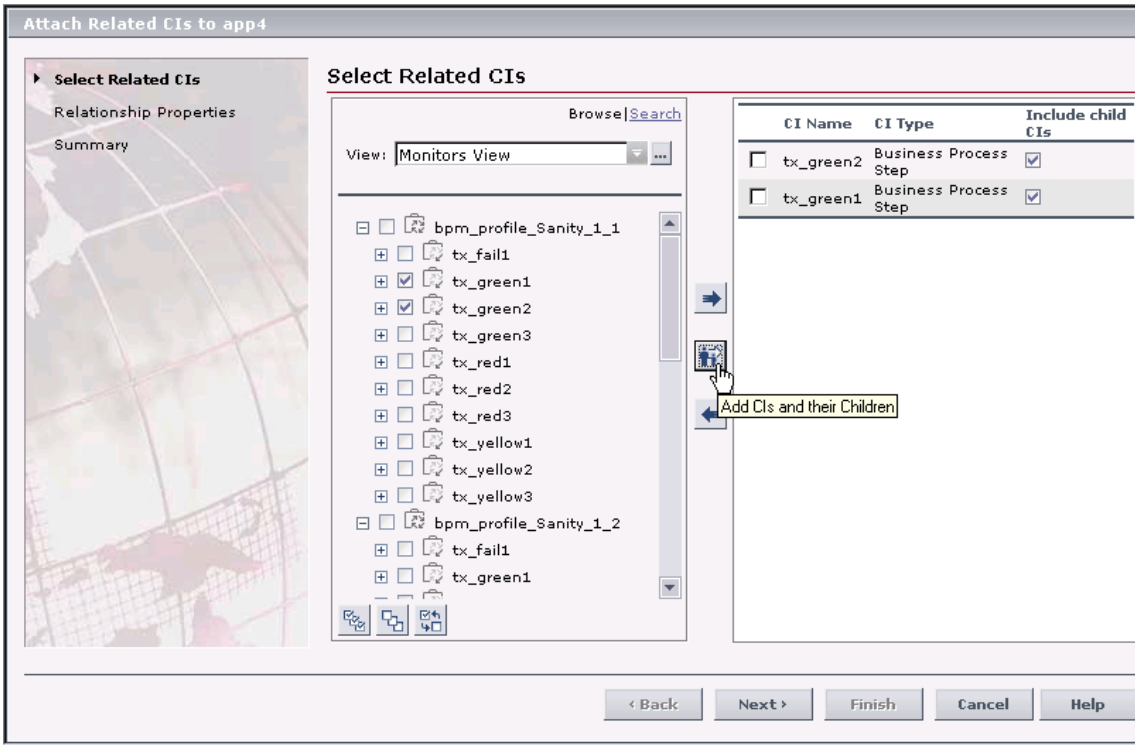
- Click the **Add CIs** (plain arrow) button if you want the added CIs set to exclude all child CIs. For CIs using this rule, the child CIs are not automatically displayed in the view. However, the view might include children for the CI, for example, for relationships that are created within the current view.



- Click the **Add CIs and Their Children** (recursive arrow) button if you want the added CIs set to include all child CIs. This means that all child CIs are recursively displayed in the view and the CI subtree is dynamically updated when child CIs are added or deleted, even if the change was made in a different view.

If you are adding CIs under a CI that is already set to include all child CIs, then the child CIs are displayed, regardless of which child rule the CIs in the subtree are using.

The CIs are added to the list in the right pane, with the **Include Child CIs** column selected as relevant.

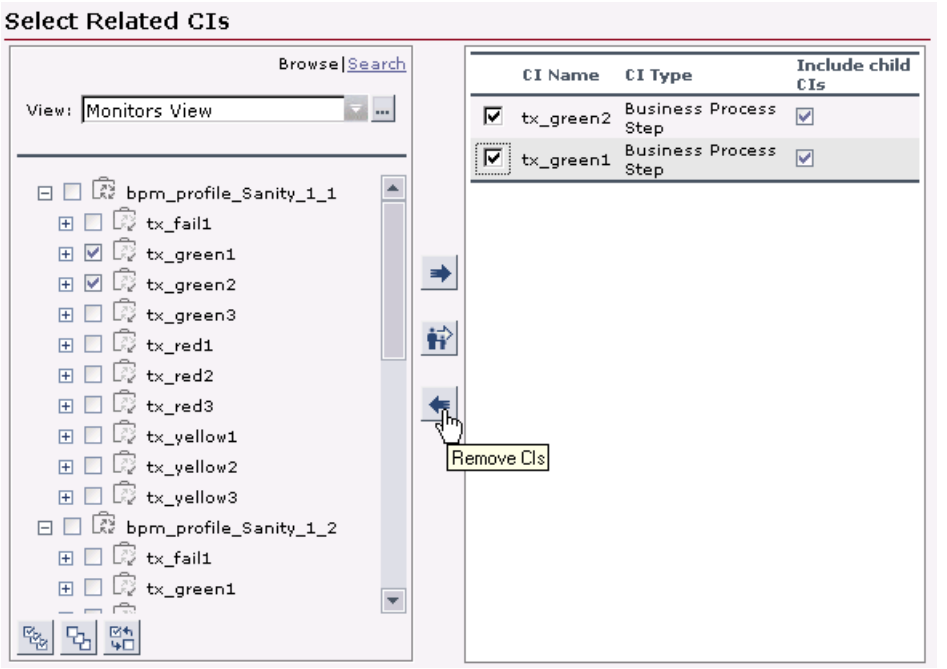


If you want to add both CIs set to include child CIs, and CIs set to exclude child CIs, do this in two separate operations, each time copying them to the view using the appropriate button.

7 After adding the CIs, they are listed in the right pane. If you wish to remove one or more CIs from this list, select the check boxes for those CIs in the



right pane and click the **Remove CIs** arrow button (this does not delete the CIs from the CMDB).



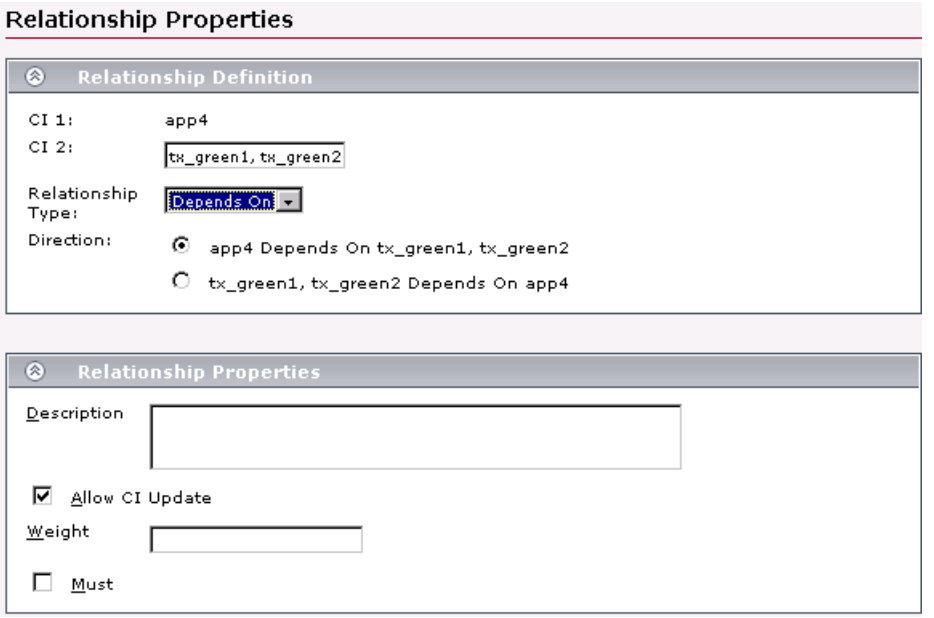
The selected CIs are removed from the list.

8 If you want to copy additional CIs from another view, select the view in the left pane and repeat the instructions in steps 5 and 6. All copied CIs should require the same relationship with the original CI.

9 When you are satisfied with the list in the right pane, click **Next**. If the selected CIs have the same relationship type, the wizard moves to the Relationship Properties page. Continue with step 10.

If there is a compatibility problem with the relationship types for the different CIs that you selected, then an error message is displayed saying that no valid relationships were found. In this case, select the CI(s) that you think are causing the problem from the list in the right pane, and remove them using the **Remove CIs** arrow button. Click **Next**.

10 In the Relationship Properties page, check the details in the **Relationship Definition** area and modify as required:



- **CI 1** – Displays the name of the CI on the first side of the relationship. This is the CI that you selected in the view.
- **CI 2** – Displays the name(s) of the CI(s) on the second side of the relationship, that will be attached to the original CI.

  If you want to modify the list of CIs to attach, use the **Back** button to return to the previous page, and edit your selection.
- **Relationship Type** – Enables you to define the type of relationship created between the original CI and the CIs being attached. Select an option from the list. For most relationships, only the **Depends On** option or the **Monitored By** option is available, depending on the type of CIs for which you are creating a relationship.
- **Direction** – Enables you to define the direction of the relationship, by selecting the appropriate radio button. For example, if [**CI 1**] **Depends On** [**CI 2**] (generally the default option), then the new CIs will be created as

child CIs of the originally selected CI. If [**CI 2**] **Depends On** [**CI 1**], then the new CI will be created as a parent to the originally selected CI.

For some relationship types (for example, **Container link**, or between a group CI and a monitor CI), only one direction is enabled.

11 In the **Relationship Properties** area, define the properties that you want to apply for these relationships, such as the **Must** and **Weight** attributes. For information on these properties, see "Relationship Property Descriptions" on page 60.

Note: When defining multiple relationships, if you need to define different descriptions and property values for each relationship, then skip the property definitions here and instead edit each relationship individually after it's created. For details, see "Editing Relationship Properties" on page 59.

---

12 Click **Finish**. The Summary page is displayed, summarizing the new relationships that are being created. If you are working in a pattern view, and the CIs or relationships are not appropriate, then the page tells you that the CIs are created in the CMDB but not added to the view.

13 Click **OK**. Where appropriate, the CIs are added to the view according to the defined relationships.

Deleting CIs

When you delete a CI, you are also deleting all relationships between that CI and its parent CIs. The CI is deleted from the CMDB and from all views where it appears.

To delete CIs from the CMDB:

1. In the **IT Universe Manager** tab, right-click the CI in View Explorer that you want to delete.
2. Select **Delete CI** from the displayed menu.
3. Click **OK** in the confirmation dialog box. The CI is removed from the view and from the CMDB.

Notes and Limitations for Deleting CIs

- Deleting a CI does not delete its child CIs. If the child CIs do not appear in any other view, then they will remain in the CMDB without appearing in a view. The child CIs may appear in the future in pattern views where they meet the pattern requirements.

  One exception to this rule is when the relationship between a child and parent CI is a **Container link**; then, when the parent CI is deleted, the contained CI is also removed from the CMDB and the view. For example, when a host CI is the container for a child CPU CI, then when the host CI is deleted, the CPU is also removed from the CMDB.

- If you delete a CI without first deleting the relationship between the CI and any child CIs that were added within the view, then the child CIs are not removed from the view, they move up to the root level.

- When you delete a CI that was created by the source adapters or the discovery process, the CI will be restored the next time that the CMDB is updated by that source (if the CI is still valid), and will reappear in all relevant pattern views.

Working with Relationships

This section contains the following topics.

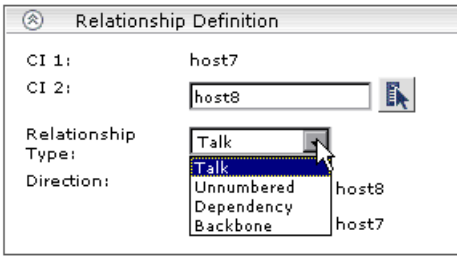
- About Relationships – for details, see page 57
- Editing Relationship Properties – for details, see page 59
- Relationship Property Descriptions – for details, see page 60
- Notes and Limitations for Creating Relationships – for details, see page 61
- Deleting a Relationship – for details, see page 62

About Relationships

Many of the relationships between CIs in the CMDB are created automatically within Mercury Business Availability Center, for example, by the discovery process, the source adapters, and the Dynamic Node Factory. In addition, you can manually create relationships using the IT Universe Manager tab and in Monitor Administration.

When working in topology map mode, you also have the option to define a relationship between two CIs in the view by drawing a line between the two. This action directly opens the Relationship Properties window, which is the same as the Relationship Properties page of the Attach Related CIs wizard (described in "Attaching Existing CIs" on page 49).

Attaching one CI to another CI in the IT Universe Manager tab or Monitor Administration creates a hierarchical relationship (parent-child) between them, so that one CI is influenced by the other. Most relationships defined in this way can only be of relationship type **depends on** (if the relationship is between two non-monitoring CIs), or of relationship type **monitored by** (if the child CI in the relationship is a monitor-type CI); however, other relationship types are used between specific types of CIs. For example, if a relationship is defined between two Host CIs, then there are four possible relationship types.



The possible relationship types between two CITs are defined in the CI Type Manager (for details, see *CI Type Manager Administration*).

When you create or attach a child CI to a CI in your view, you are also creating that relationship in the CMDB. The child will therefore appear for all other instances of the parent CI in every view, where the subtree is set to include all child CIs (and for pattern views, where the child and relationship matches the TQL). For more information, see “Include Child CIs” on page 70.

You can create as many relationships as you want between a CI in the CMDB and other CIs. Each new relationship is its own entity in the CMDB, and has its own properties that can be edited in the IT Universe Manager tab (as described in “Editing Relationship Properties” on page 59). You can also create more than one relationship between the same two CIs, when each relationship is based on a different relationship type.



The following lists the relationships types available in the CIT Manager (and used in TQLs) that the Business Logic Engine recognizes when performing status calculations:

- member
- applicationLink
- monitored_by_for_sap
- contained
- contains
- container_f
- depends_on
- monitored_by
- eum_links
- system_monitor_links
- monitor_links

Editing Relationship Properties

You can edit some relationship properties (but not the relationship definition).

To edit relationship properties:

1. In the **IT Universe Manager** tab, access the required view in View Explorer and select the parent or child CI in the relationship.
2. Open the **Related Configuration Items** tab in the right pane, and find the required relationship in the table.
3.  Click the **Edit relationship** button for the relationship you want to edit. The Relationship Properties window is displayed.

4. Modify the values in the **Edit Relationship Properties** area, as required. For information on these properties, see "Relationship Property Descriptions" on page 60.
5. Click **OK** to save your changes. After modifying relationship properties, the **Update Time** property will be updated with the time that you made the changes.

Relationship Property Descriptions

The following properties are displayed for relationships. Some properties can be defined while creating a relationship, others are defined automatically.

- **CI Type** – the type of relationship, for example, **Monitored By** or **Depends On**.
- **Create Time** – the date and time when the relationship was created.
- **Update Time** – the date and time when the relationship was last updated.
- **Description** – a description for the relationship.
- **Origin** – the user name of the administrator who manually created the relationship, or an ID for the source that automatically created the relationship.
- **Allow CI Update** – when the check box is selected, the discovery process will automatically update the relationship.
- **Weight** – when a weight value is entered in the box, the child CI is given more weight in percentage calculations, when percentage rule is used to calculate status for the parent CI. For example, if a child CI is given a weight of **3**, then it will have three times more impact on the parent status calculation than its unweighted siblings.

  The weight defined for a relationship only applies for that relationship—it will not be used in any other relationships the child CI may have. For more details, see "Percentage Rule" in *Application Administration*.

- **Must** – when the check box is selected, the child CI is dominant when the percentage rule is used to calculate status for the parent CI. If the CI has low status, the parent must take the child CI status, if necessary overriding the percentage rule calculation. This is useful to give emphasis to an important CI. For example, you may want to define **Must** for a database CI, so that status will be **critical** all the way up the subtree if the database falls, regardless of the status of the other CIs in the subtree.

  The must status defined for a relationship only applies for that relationship—it will not be used in any other relationships the child CI may have. For details, see "Percentage Rule" in *Application Administration*.

- **Network Address** – For relationships created by the discovery process, defines the discovered IP net address or the net address range.

Notes and Limitations for Creating Relationships

- If, when attaching multiple CIs to a CI, you define that the attached CIs will all be parents to the original CI (for example, so that the relationship direction reads that CI X, CI Y, CI Z Depends On CI 1), then a separate subtree will be created in the view for each parent, with the original CI appearing as a child in each subtree.

- You cannot create exactly the same relationship twice within a view. For example, if parent CI Italy depends on child CI Rome, Rome will appear under the CI Italy in the view. If you then again attach Rome to Italy with a **depends on** relationship, Rome will still only appear once under that parent CI. This is true even if you attach each instance of the child with different settings, for example, one instance set to include child CIs, and one instance set to not include them.

- An instance of a CI at the root level, with no relationships, is automatically removed when another instance of the CI is added to the view, this time as a child CI. If you later remove the relationship for the second instance, the first instance of the CI will reappear as a root CI in the view. (This does not apply when working with the Instance View Editor.)

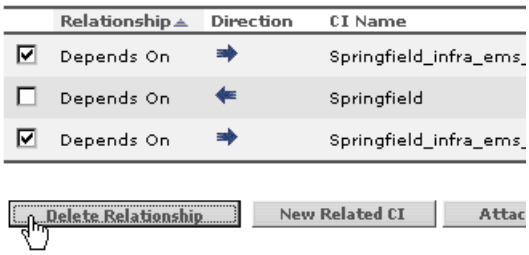
- When you add an existing CI to an instance view, the CI may also appear at other places in the view where a relationship exists for that CI, but the CI was hidden.

Deleting a Relationship

You can permanently remove the relationship created between two CIs. This will not remove either of the related CIs from the CMDB, but will remove the relationship between the two CIs from the CMDB and all views.

To delete relationships from the CMDB:

1. In the **IT Universe Manager** tab, select the CI in View Explorer for which you want to delete a relationship.
2. In the **Related Configuration Items** tab in the right pane, select one or more check boxes for the relationships that you want to delete.
3. Click **Delete Relationship**.



4. Click **OK** in the confirmation dialog box. The relationship(s) are removed from the list in the **Related Configuration Items** tab and from the CMDB.

Notes and Limitations for Deleting a Relationship

- The relationship table shows only the relationships for the CI that are part of the current view; however, the same relationships may also be part of other views, so the relationships will also be deleted from those views.
- Deleting a relationship does not remove from the view CIs that were added within that view. If, after deleting a relationship, the CI is no longer related to any other CI in the view, then the CI is shown at the top level of the view. To remove the CI completely from the view (without deleting it from the CMDB), use the **Disconnect from Parent CI** option in the Instance View Editor (as described in “Disconnect CI from Parent” on page 38).

- When you delete a relationship that was created by the source adapters or the discovery process, the relationship will be restored the next time that the CMDB is updated by that source (if the relationship is still valid), and will reappear in all relevant pattern views.
- It may take a few minutes to delete subtrees, or multiple CIs and relationships, from a view.

4

Configuration Item Properties

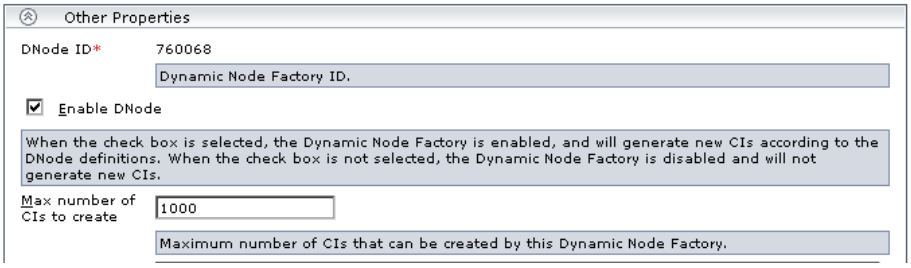
Configuration items (CIs) have general properties, common to all CIs, and properties specific to the configuration item type (CIT). Some of these properties can be modified in the IT Universe Manager Properties tab.

This chapter describes:	On page:
Editing CI Properties	65
General Properties for CIs	66
CIT-Specific Properties	72

Editing CI Properties

You can edit different properties for CIs in the Properties tab, depending on the type of CI.

Click the **Help** button at the bottom of the Properties tab to see more information about the properties fields or to see the internal name of the property.



To edit CI properties:

1. In the **Admin** > **CMDB** > **IT Universe Manager** tab, select the required view and CI in View Explorer.
2. In the right pane, click the **Properties** tab, to display the properties for the selected CI.
3. Edit the appropriate fields. For information on the properties, see "General Properties for CIs" on page 66 and "CIT-Specific Properties" on page 72.
4. Click **Apply** to save the changes. After modifying the CI properties, the **Update Time** and **Updated By** properties are immediately updated with the time that you made the changes and your user name.

General Properties for CIs

The CI general properties are common to all CIs. Some of the general properties are defined automatically during CI creation; others can only be defined manually in the General Properties area/page.

You can view properties for a CI from any application that uses View Explorer, by right-clicking the CI in View Explorer and selecting **Properties** from the displayed menu. Editing the properties can only be done in the IT Universe Manager Properties tab, as described in "Editing CI Properties" on page 65.

This section includes the following topics:

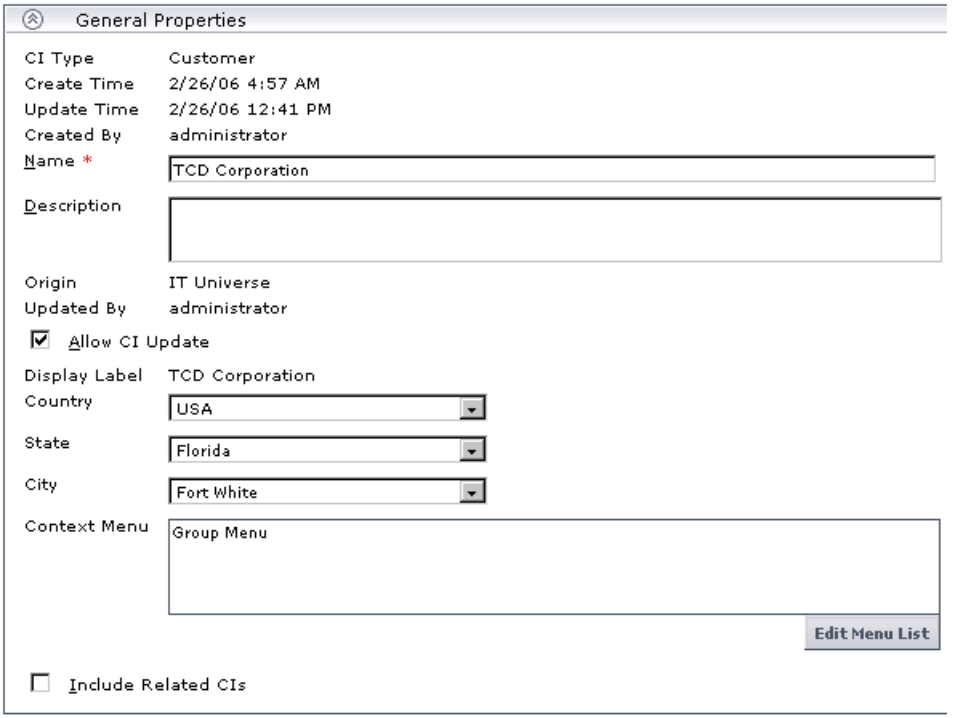
➤ General Property Descriptions – for details, see page 67

➤ Editing the Menu List – for details, see page 68

➤ Include Child CIs – for details, see page 70

General Property Descriptions

The General Properties area includes the following information:



- **CI Type** – The CIT for the CI. For information about CITs in Mercury Business Availability Center, see *CI Type Manager Administration*.
- **Create Time** – The date and time when the CI was created.
- **Update Time** – The date and time when the CI properties were last updated.
- **Created By** – Where applicable, user name of the administrator who manually created the CI.
- **Name** – The name of the CI; it may be different from the **Display Label**.
- **Description** – A description of the CI.
- **Origin** – An ID for the source that automatically created the CI, or the application where the CI was created.

- **Updated by** – The user name of the administrator who updated the CI properties.
- **Allow CI Update** – When this option is selected, the discovery process can automatically update the CI properties with information that it discovers. If you change the value of a property supplied by the discovery process, the value will be overridden by the discovered value. When this option is cleared, the CI properties are not overridden by the discovery process.
- **Display Label –** The name of the CI as it appears in View Explorer.
- **Country** – The name of the country where this CI is located. Select an option from the list. You must define a geographical location—country, state (if applicable), and city—if you want the CI to appear on the Geographical Map in Dashboard. For more information, see "Configuring the Geographical Map" in *Application Administration*.
- **State** – The name of the state - when applicable for the selected country. The list is automatically updated when you select a country. Select an option from the list.
- **City** – The name of the city. The list is automatically updated when you select a country (or state, when applicable). Select an option from the list.
- **Context Menu** – The context menu that is attached to the specific CI and displayed in Dashboard. Some of the CIs are automatically assigned a context menu; for example a Group CI is automatically assigned the Group Menu. To modify or add other context menus to the list, click **Edit Menu List**. For details, see "Editing the Menu List" on page 68.
- **Include Related Elements** – (Used in instance views only) When this option is selected, the child CIs for the selected CI are included in the current view. For details, see "Include Child CIs" on page 70.

Editing the Menu List

You can define one or more context menus for a CI or modify the menu list. The menus are displayed for the CI in Dashboard.

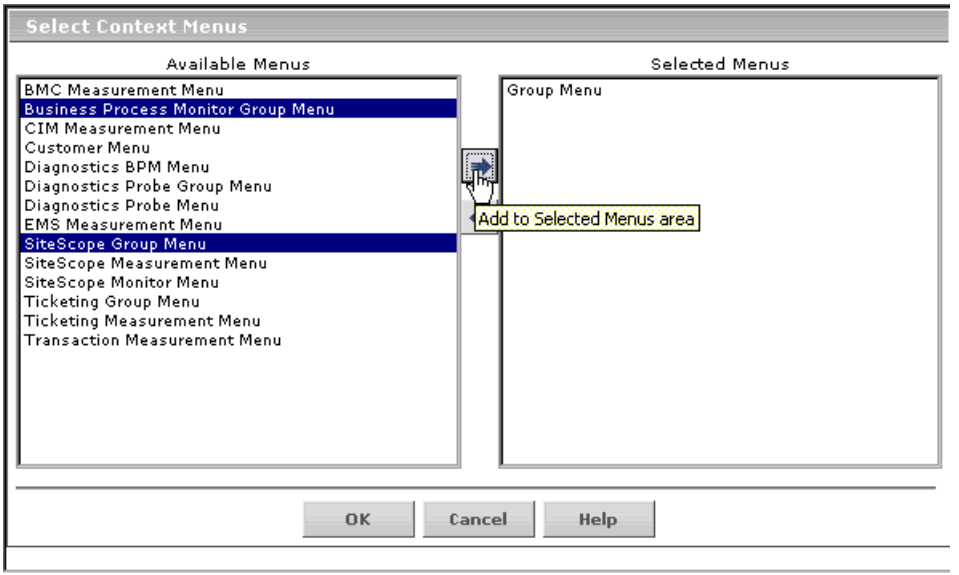
When defining menus, you should make sure that the selected menus are logical for the CI.

For information on context menu content, see "Context Menu Items Repository" in *Application Administration*.

To edit the context menu list for a CI:

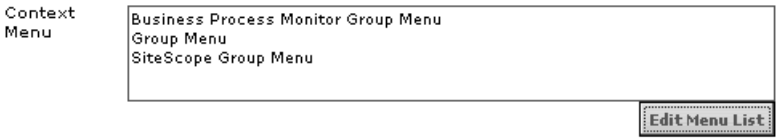
1 Open the IT Universe Manager **Properties** tab for the CI and click **Edit Menu List** (to the right of the **Context Menu** box).

2 In the displayed Select Context Menus window, select the required menu(s) in the **Available Menus** area (you can select more than one using CTRL and SHIFT) and click the right arrow to move them to the **Selected Menus** area.



3 To remove menus from the menu list for the CI, select them in the **Selected Menus** box and click the left arrow.

4 Click **OK**. The selected menus are displayed in the **Context Menus** box on the Properties tab.



Note: You must click **Apply** in the Properties tab for the menu changes to be saved.

Include Child CIs

In instance views, CIs can be displayed so that they include all their child CIs (the full subtree), or they can exclude most of the child CIs, depending on the child rule used for the CI. (In a pattern view, all child CIs are always displayed for a CI, as long as they match the TQL.)

The child rule assigned to a CI determines what child CIs are displayed in the view, as follows:

- A CI set to **include all child CIs** recursively displays in the view all child CIs in its subtree. The subtree is dynamically updated with every added child CI, whether it is added within the current view or in a different view.

  Once a parent CI is set to include all child CIs, the child rule used for the CIs on the next branches in the subtree is irrelevant—all child CIs are displayed all the way down the subtree.

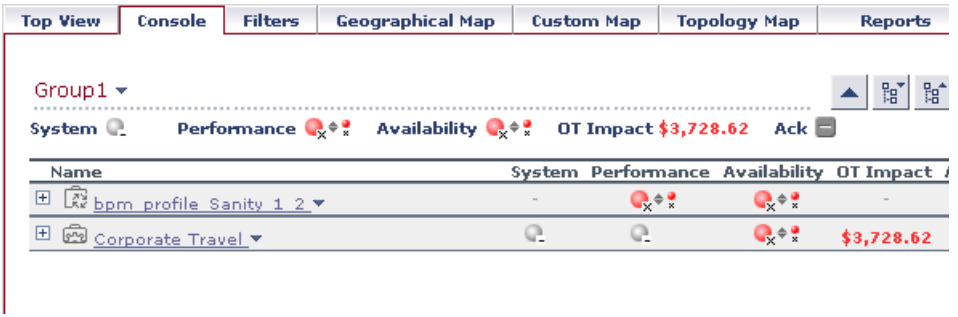
- A CI set to **exclude all child CIs** does not automatically display the child CIs in its subtree. It may display some child CIs, for example, child CIs that were created or added to the parent CI within the current view, or CIs that have additional instances elsewhere in the view.

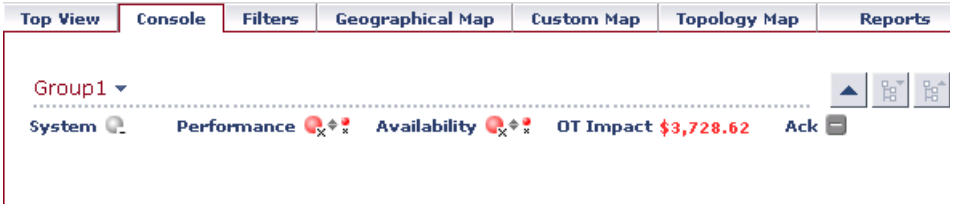
There are various ways that a CI can be set to include all child CIs:

- By selecting **Include Related Elements** in the CI properties when creating or editing the CI. (This option may be non-editable, depending on the definitions for the parent CI.)
- By using the recursive arrow when adding the CI to a view.
- By selecting the **Include All Child CIs** option in the Instance View Editor. For details, see "Child Rule Functionality" on page 32.

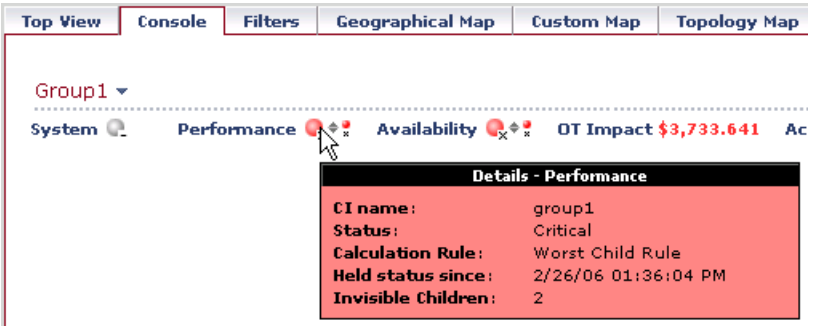
Whether or not all child CIs are included in the view does not influence the status calculations made for the parent CI and the status displayed in Dashboard. The Business Logic Engine calculates status from all child CIs attached to that parent CI, across all views. The calculation is based on all the relationships for the CI in the CMDB, without connection to the child CIs displayed in any particular view.

For example, the following pictures show status in Dashboard for the same CI, Group1, in two views: In the first view, the CI is set to include all child CIs, and in the second, to exclude all child CIs. The same status result is shown for the CI in both views, regardless of the child rule in use.





When a CI is set to exclude all child CIs, then the KPI tooltip in Dashboard shows the number of invisible children (the number of child CIs directly under the CI), to indicate that the KPI status is based on CIs that are not in the view.



CIT-Specific Properties

The CIT-specific properties are created as part of the CIT definition in the CI Type Manager (for details, see *CI Type Manager Administration*). The values for the CIT-specific properties are taken from the data samples for the source adapter CIs, or discovered during the discovery process for the discovery CIs. The values can also be entered manually, when you manually create CIs to represent entities in your infrastructure.

The CIT-specific properties are displayed in the **Other Properties** area in the Properties tab. Some of the properties for the automatically created CIs can be edited; however, if the CI is set to **Allow CI Update** (in the general properties), then the discovery process may override your changes when it updates the CI.



Use the **Help** button at the bottom of the Properties tab to see more information on the CIT-specific properties in the **Other Properties** area; use the **Show Description** button in the top-right corner of the New CI Wizard to see more information on the CIT-specific properties contained there.

Note the following regarding CIT-specific properties:

- The CIT-specific properties for the Dynamic Node Factory CI are described in Chapter 6, “Working with Dynamic Node Factory.”

- For CIs generated by the source adapters, the only editable CIT-specific property is the **Selector Expression** for the monitor-type CIs. To edit the selector, click **Edit Selector Expression**. For details on editing selectors, see "Selectors for KPIs" in *Application Administration*.

  After modifying the selector, the current CI will be updated according to the new selector definition, and a new CI will be generated by the source adapters from the incoming samples based on the original selector.

- Some of the CIT-specific properties for discovery process CIs are set by default to be monitored for changes. When one of these properties has been modified, information on the change can be viewed in a change report, and in some cases, and icon is displayed in Dashboard. For more information, see "Change Report" in *Using Dashboard*.

5

Working in Topology Map Mode

This chapter introduces the topology map. The topology map enables you to view single level or multilevel maps, which display the results of instance views and pattern views defined in the View Manager.

This chapter describes:	On page:
About the Topology Map	76
Displaying a View in Topology Map Mode	77
Viewing Layers with a Large Number of CIs	77
Understanding Topology Map Concepts	78
Using the Toolbar Options Specific to IT Universe Manager	81
Using the Topology Map Toolbar	82
Understanding Layer Setup	86
Defining a Layer's Layout	86
Modifying Layer Layout Options	88
Understanding Layout Options	105

About the Topology Map

The topology map enables you to view the results of the views you created in the View Manager. Mercury Business Availability Center allows you to create two types of views: instance views and pattern views. For more information on creating views, see "Defining View Properties to Create Views" on page 17 in *View Manager Administration*.

Instance views are created by defining the individual CIs and relationships that make up the view directly in the view. All the CIs in an instance view are seen on one level.

Pattern views are created from a TQL (Topology Query Language) query that defines the structure of the view. Pattern views allow you to define organization rules, enabling you to display topology maps in multiple layers. For more information on organization rules, see "Adding Organization (Folding) Rules to Relationships" on page 82 in *View Manager Administration*.

The topology map displays the results of the currently selected view in the View Explorer, and consists of CIs that are defined in the view and the relationships between them. You can change the way the maps are displayed by selecting one of the view layouts (**Hierarchical, Symmetric, or Orthogonal**). For details, see "Understanding Layout Options" on page 105. You can also modify the values of the default layer layouts, as described in "Defining a Layer's Layout" on page 86.

In addition, you can right-click a CI in the topology map to perform the following functions.

- Edit CI properties
- Create a related CI
- Attach a related CI
- Delete a CI

For information on how to perform these functions, see "Using the Toolbar Options Specific to IT Universe Manager" on page 81.

Displaying a View in Topology Map Mode

There are two modes in which you can display view information.

- **Default**
- **Topology mode**

When you first access open Mercury Business Availability Center, the view is automatically displayed in **Default** mode. This section describes how to display the view in **Topology Map** mode.

To display the view in Topology Map mode:

1. Select **Admin** > **CMDB** and then select the **IT Universe Manager** tab.
2. In the View Explorer, select the required view from the **View** list.
3. From the **Viewing Mode** list in the top-right corner of the **IT Universe Manager** tab, select **Topology Map**.

Viewing Layers with a Large Number of CIs

A layer cannot display more than 900 CIs in a topology map view. If the layer contains more than 900 CIs, it cannot be opened. If you try to open a layer that contains more than this amount, a message is displayed. In this case, it is recommended to use organization rules to spread the CIs among several layers. For using organization rules, see "Adding Organization (Folding) Rules to Relationships" in *View Manager Administration*.

Understanding Topology Map Concepts

This section describes the following topology map features:

- "Understanding CI Icon Composition" on page 78
- "Selecting Multiple CIs" on page 79
- "Simultaneous Selection" on page 79
- "Working with Context Menu Options" on page 79
- "Tooltips" on page 79
- "Navigating Between Different Map Levels" on page 80

Understanding CI Icon Composition

The icon that represents each CI in the topology map is defined by the administrator in the CI Type Manager. For information on how to assign an icon to a CIT, see "Assigning an Icon to a CIT" in *CI Type Manager Administration*.

CI icons are made up of the following components:



- An icon.



- The group to which it belongs. For example, Network or SAP System, appears in the upper left-hand corner.



If an icon appears with a shadow, it means that there is an additional CI layer beneath it, created by a parent or child organization rule defined in the View Manager. For information on how to set up organization rules, see "Adding Organization (Folding) Rules to Relationships" in *View Manager Administration*. For information on how to navigate between layers, see "Navigating Between Different Map Levels" on page 80.

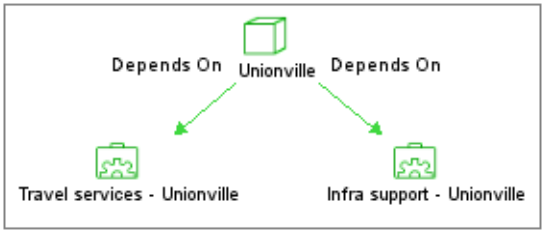
Selecting Multiple CIs

In the topology map, you can select multiple CIs.

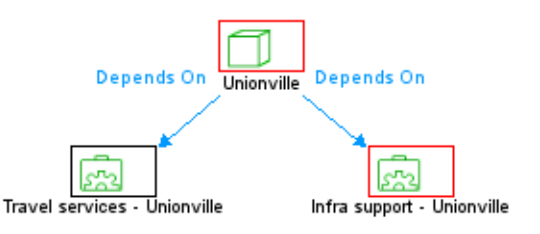
To select multiple CIs:

Hold down the CTRL button and click the CIs or drag the mouse diagonally along the CIs as shown in the following figure:

Before



After



If you select multiple CIs, only one CI has a black selection rectangle around it, while all the other selected CIs have red selection triangles around them (see the After picture above). In this situation, when you right-click anywhere on the map to open the context menu, the option you choose is performed only on the CI surrounded by the black rectangle.

Simultaneous Selection

When you select a CI, it simultaneously selects the CI in both the View Explorer and the topology map. For more information on the View Explorer, see "Using View Explorer" in *Working with the CMDB*.

Working with Context Menu Options

Right-clicking a CI in the topology map opens a context menu. For details about the context menu options, see "Using the Toolbar Options Specific to IT Universe Manager" on page 81.

Tooltips

When you point to a CI in the topology map, a tooltip is displayed that contains the CI type of the selected CI as defined in the CI Type Manager.

CI Labels

Each CI on the topology map has a label that appears underneath it. The label definition can be customized to include different attribute values. For information on how to customize the label definition, see "Defining an Attribute for a CIT Label" in *CI Type Manager Administration*.

Navigating Between Different Map Levels



An icon with a shadow indicates that an additional CI layer lies beneath it, created by a parent or child organization rule defined in the View Manager.

For example, if the IP CI is defined as a child of the host CI in the View Manager, you can double-click any host icon in the topology map to display a map of IPs connected with that host.

To drill down to the lower level(s):

Double-click an icon with a shadow.

To move up one level in the topology map:

1. Select a CI that has an additional CI layer above it.
2.  Click the **Up layer** button to move up to the next level.

   The **Up layer** button is enabled only if there is an additional CI layer above it, created by a parent or child organization rule defined in the View Manager. For information on how to set up organization rules, see "Adding Organization Rules to Relationships" in View Manager Administration.

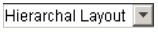
Using the Toolbar Options Specific to IT Universe Manager

The following table provides a brief description of each toolbar option that is specific to IT Universe Manager.

Option Name	Icon	Use this option to...
Up one level		Move up one level in the topology map. This option is enabled only if there is an additional CI layer above it, created by a parent or child organization rule defined in the View Manager. For information on how to set up organization rules, see "Adding Organization Rules to Relationships" in View Manager Administration.
Draw relationship		Create a relationship between existing CIs. For details, see "Attaching Existing CIs" in *IT Universe Administration*.
New CI		Create a new CI. For details, see "Creating a New CI" in *IT Universe Administration*.
Edit properties		Edit CI properties. For details, see "Editing CI Properties" in *IT Universe Administration*.
New related CI		Create a related CI. For details, see "Creating a New CI" in *IT Universe Administration*.
Attach related CIs		Attach a related CI. For details, see "Attaching Existing CIs" in *IT Universe Administration*.
Delete		Delete the selected CI.

Using the Topology Map Toolbar

The following table describes the common toolbar buttons that appear in CMDB Administration and Dashboard.

Option Name	Icon	Use this option to...
Print		Print the contents of the topology map.
Layer Setup		Define the view's layer layout, by assigning positions for the nodes and relationships of the view. For details, see "Modifying Layer Layout Options" on page 88.
Layout		Select the layout display from the list. For details, see "Understanding Layout Options" on page 105.
Reorganize layer		Reorganize the CIs/CITs/nodes according to the selected layer. This is used when CIs/CITs/nodes have been moved around.
Save manual layout		Save changes to the manual layout in the topology map. For details, see "Manual Layout" on page 107. Save manual layout is relevant only to the topology map in Dashboard (see "Dashboard Topology Map" in *Using Dashboard*) and the topology map in IT Universe Manager (see "Working in Topology Map Mode" in *IT Universe Administration*).
Up layer		Go up one layer. This option is relevant only to the topology map in Dashboard (see "Dashboard Topology Map" in *Using Dashboard*) and the topology map in IT Universe Manager (see "Working in Topology Map Mode" in *IT Universe Administration*).
Refresh		Go to the server and load the layer again.

Option Name	Icon	Use this option to...
Hide All Relationship Labels		Toggle between showing or hiding the relationship labels.
Hide All Nodes Labels		Toggle between showing or hiding the CI/CIT/node labels.
Set label size		Enter the maximum number of characters you want the labels to contain.
Zoom		Zoom in on a specific section of the topology map by clicking and drawing a selection rectangle around the part you want to magnify. The area you have selected is displayed at the highest percentage that fits in the topology map.
Interactive Zoom		➤ Decrease the view magnification by clicking and pushing the pointer upwards. ➤ Magnify the view by clicking and pulling the pointer downwards.
Fit to window		Fit the complete view in the window.
Overview map		Opens a separate window with only the topology map.
Pan		Drag the view.

Option Name	Icon	Use this option to...
Navigation		Take the pointer to the next connected CI/CIT/node. ➤ Left-click to go from CI/CIT/node to CI/CIT/node in a clockwise direction. ➤ Right-click to point to the next CI/CIT/node in a clockwise direction. **Note:** You can use this option only on CIs/CITs/nodes that are connected by relationships.
Place in center		Place the selected CI/CIT/node or CIs/CITs/nodes in the center of the map.
Select mode		Select a CI/CIT/node or multiple CIs/CITs/nodes.

Printing the Topology Map

You can print the contents of the topology map pane. The result is similar to a screen capture. Therefore, it is recommended to arrange the contents of the Topology map according to your requirements before printing.

To print the contents of the topology map:

1. Select and display the contents you want to print in the topology map.
2. To arrange the contents of the topology map for print, you can use the **Layer Setup** and **Layout** options.

   For details about the **Layer Setup** option, see "Understanding Layer Setup" on page 86.

   For details about the **Layout** option, see "Understanding Layout Options" on page 105.



3. Select the **Print Map** button to open the Print dialog box.
4. In the **Copies** area, enter the number of copies you want to print.
5. Click **Properties** to open the Document Properties dialog box.
6. Define document properties as required.
7. Click **OK** in the Document Properties dialog box to save your changes.
8. Click **OK**.

Understanding Layer Setup

The **Layer Setup** option enables you to define the layout of a view's layers in a way that helps you achieve the clearest presentation possible. By assigning logical positions for the nodes and relationships that appear in the layer, you can fine-tune your view's visualization and thus facilitate the understanding and monitoring of your managed data.

This option applies only to nodes and relationships that are displayed in the topology map. The definitions that are set in the dialog boxes apply to a certain layer in a certain view and they determine the effects and behavior of the available layout settings (**General**, **Disconnected**, **Hierarchical**, **Orthogonal**, **Symmetric**, and **Routing**).

The **Layer Setup** option provides you with default values for each parameter, which you can leave or change as required. The parameter values are in pixels.

Defining a Layer's Layout

You can customize the layer layouts by modifying the default values.

To define a layer layout:

1 Click **Layer Setup** to open the Layer Properties dialog box with the following tabs:

- **General** – Defines general layout types for all layout types. For details, see "General Layout" on page 88.
- **Disconnected** – Defines general layout parameters for disconnected nodes and connected components. For details, see "Disconnected Layout" on page 89.
- **Hierarchical** – Shows the precedence relationships of the nodes in the selected layer. For details, see "Hierarchical Layout" on page 91.
- **Orthogonal** – Routes relationships only horizontally and vertically. For details, see "Orthogonal Layout" on page 99.
- **Symmetric** – Displays a clear representation of complex networks. For details, see "Symmetric Layout" on page 100.

- **Routing** – Leave nodes where they are in the graphic representation and reroutes the edges orthogonally. For details, see "Routing Layout" on page 102.

2 Enter new values for the layout type you want to change. See the following sections for details on each layout type.

- Click **Defaults** to restore the original default values.
- Click **Apply** to apply the changes you have made. The changes you made are saved but the dialog box remains open.
- Click **OK** to save the changes you have made to the database and close the dialog box.

Modifying Layer Layout Options

This section describes how to define the layout of view's layers.

Note: The layout definitions you set are per layer only and cannot be saved once you move to a different layer or a different view.

It includes the following topics:

- General Layout – for details, see page 88
- Disconnected Layout – for details, see page 89
- Hierarchical Layout – for details, see page 91
- Orthogonal Layout – for details, see page 99
- Symmetric Layout – for details, see page 100
- Routing Layout – for details, see page 102

General Layout

The General Layout settings are settings that are available with all layout styles. If your layout contains more than one graph, each graph is laid out separately, and then the intergraph edges (the relationships between the graphs) are routed.

In this tab you define General Layout parameters.

- **Margin Spacing** – Sets the borders around each node in the layer.
  - **Constant Spacing** – Sets a constant space around the displayed layer, vertically and horizontally. This space remains the same even if the size of the layer changes.
    - **Left** – The size of the space to the left of the display. Default is 20.
    - **Top** – The size of the space above the display. Default is 20.
    - **Right** – The size of the space to the right of the display. Default is 20.
    - **Bottom** – The size of the space under the display. Default is 20.

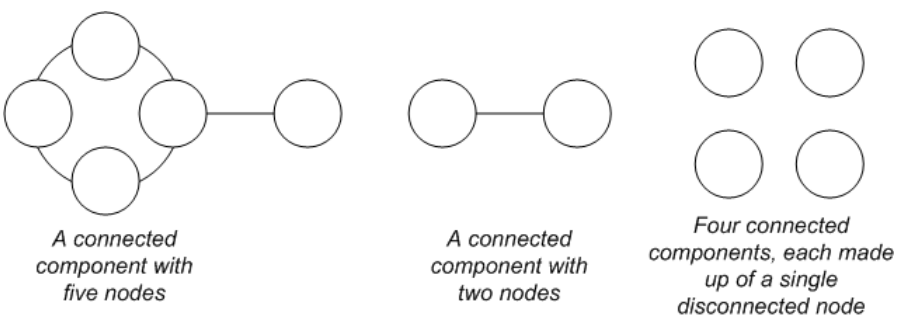
- **Proportional Spacing** – Sets a space around the displayed layer; the space is proportional to the display's size. This space changes when the size of the display changes.
  - **Left** – The size of the space to the left of the display. Default is 0.
  - **Top** – The size of the space above the display. Default is 0.
  - **Right** – The size of the space to the right of the display. Default is 0.
  - **Bottom** – The size of the space under the display. Default is 0.
- **Nested View Spacing** – For future use.
- **Intergraph Edge Spacing** – If intergraph edges are routed orthogonally, you can specify horizontal and vertical spacing for the edges.
  - **Horizontal Spacing** – Sets the horizontal distance between edges.
  - **Vertical Spacing** – Sets the vertical distance between edges.

Disconnected Layout

In this tab, you define general layout parameters for disconnected nodes and connected components:

- A disconnected node is a node that is not connected to any other node.
- A disconnected component is a component that is not connected to any other component. It contains a set of nodes that can be connected to each other.

The following example illustrates six connected components:



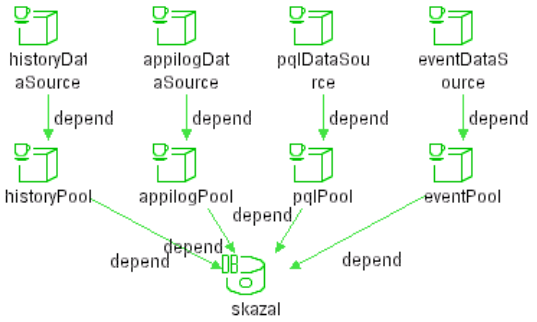
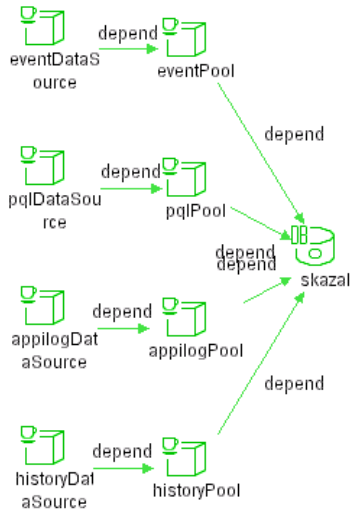
- **Components** – The spacing between the components in a disconnected graph consists of both a constant value and a proportional value based on the sizes of the components.
  - **Detect Components** – Select this option to view the disconnected components. You can specify that all components be laid out together or individually, regardless of the other components. If the grouping for components is selected, each component is laid out, and then the resulting components are packed together.
  - **Constant Spacing** – Sets constant spacing (horizontal and vertical) around each disconnected component. Default is 20.
  - **Proportional Spacing** – Sets spacing (horizontal and vertical) that is proportional to the size of the component around each disconnected component. Default is 0.05.
- **Disconnected Nodes** – The spacing between the nodes in a disconnected graph consists of both a constant value and a proportional value based on the sizes of the nodes.
  - **Detect Disconnected Nodes** – Select this option to view the disconnected nodes. You can specify that disconnected nodes be grouped into one component or treated individually while being laid out.
  - **Constant Spacing** – Sets constant spacing (horizontal and vertical) around each disconnected node. Default is 20.
  - **Proportional Spacing** – Sets spacing (horizontal and vertical) that is proportional to the size of the component, around each disconnected node. Default is 0.05.
- **Aspect Ratio** – The layout's proportions. This only affects the placement of components with respect to each other, not the layout of each individual component. It has no effect if the graph is connected.
  - **Automatic** – The layout is performed automatically.
  - **Custom** – You can specify the aspect ratio. The default is 1.0.

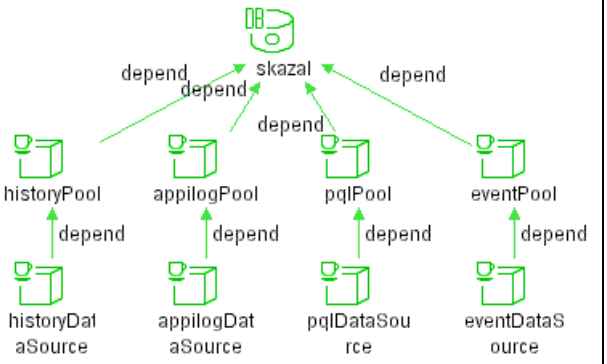
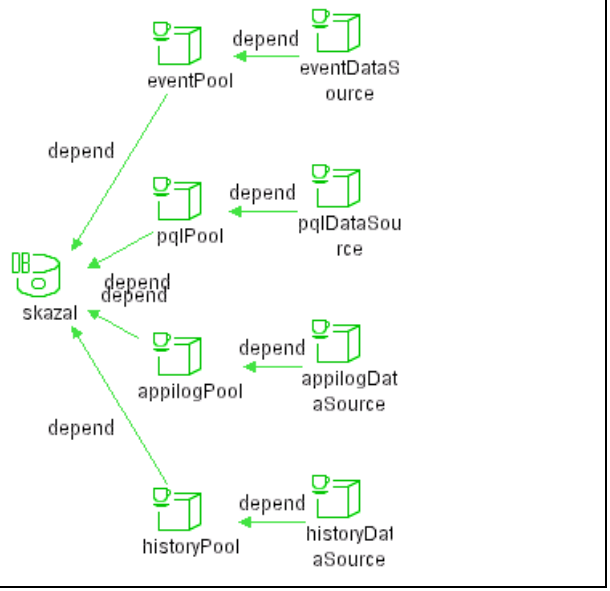
Hierarchical Layout

The hierarchical layout shows the precedence relationships that can represent organizational or information management system dependencies, as well as process models, software call graphs, and work flows. The hierarchical layout emphasizes dependencies by placing the nodes at different levels.

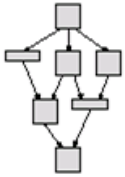
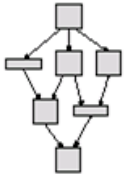
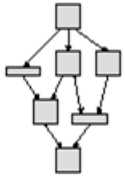
There are several layout options you can apply to a hierarchical layout, as described below.

➤ **Orientation –** The orientation of the hierarchy.

Top to Bottom – The children are above the parent (in the example, Alignment = Center).	
Left to Right – The children are to the left of the parent (in the example, Alignment = Center).	

Bottom to Top – The children are below the parent (in the example, Alignment = Center).	
Right to Left – The children are to the right of the parent (in the example, Alignment = Center).	

➤ **Level Alignment –** This option allows you to control the way nodes are aligned on a vertical level.

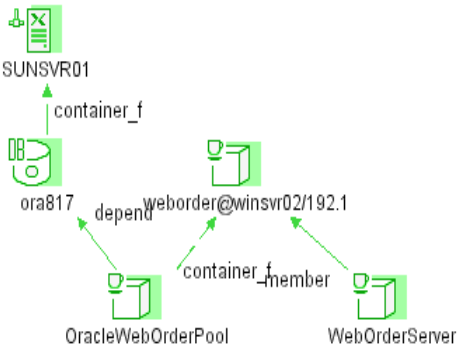
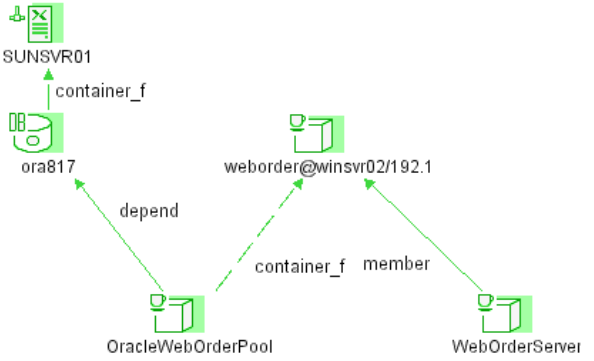
Top – The top of the nodes at the same level of the hierarchy are aligned.	**Center** – The center of the nodes at the same level of the hierarchy are aligned.	**Bottom** – The bottom of the nodes at the same level of the hierarchy are aligned.
		

➤ **Layout Quality** – You can adjust the quality of the layout produced to fit your application's needs. The quality reflects the number of steps or the method used to produce the layout .

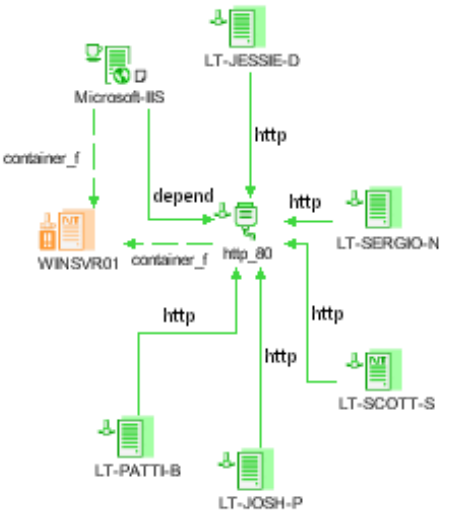
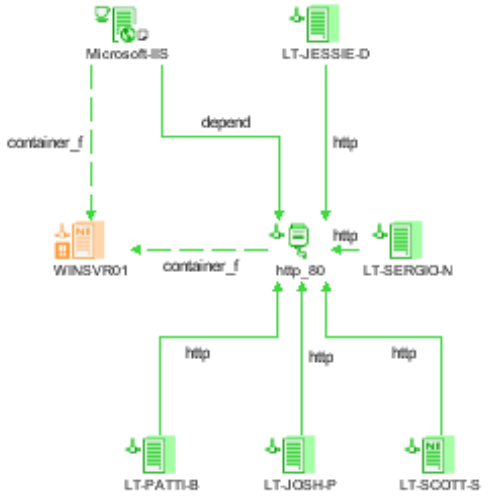
- ➤ **Draft**
- ➤ **Default**
- ➤ **Proof**

➤ **Horizontal Spacing –** Sets the minimum horizontal distance between two neighboring nodes on each level.

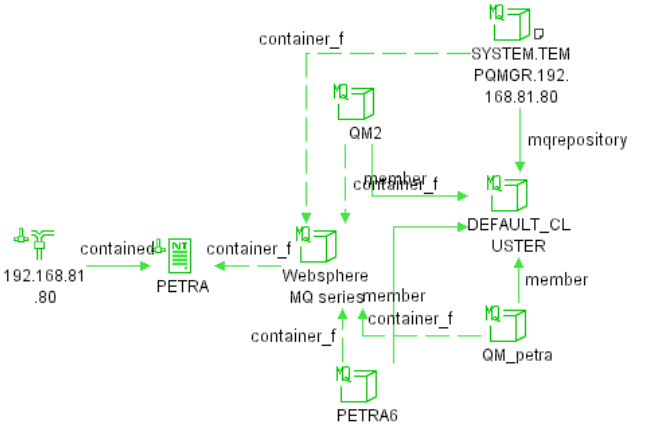
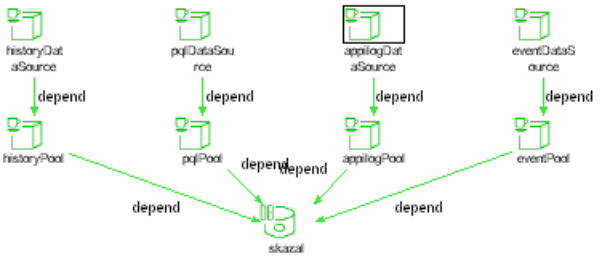
➤ **Spacing Between Nodes** – Sets a space around each node horizontally in the layer. The default value is 20.

Spacing Between Nodes = 20	
Spacing Between Nodes = 75	

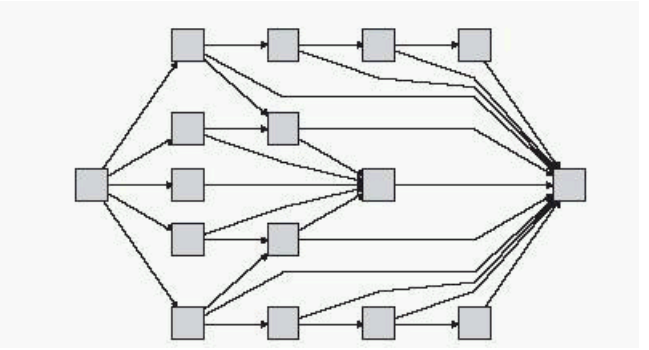
➤ **Spacing Between Edges** – This setting is available only when you select **Routing** > **Orthogonal**. It sets a horizontal space between elements in the graph.

Spacing Between Edges = 12	
Spacing Between Edges = 90	

- **Vertical Spacing** – Sets the minimum vertical distance between two neighboring nodes on different levels.
  - **Spacing Between Nodes** – Sets a vertical space around each node in the layer. The default value is 20.
  - **Spacing Between Edges** – This setting is available only when you select **Routing** > **Orthogonal**. It sets a vertical space between edges in the graph.
- **Routing** – Sets the type of relationship between CIs. This option is useful when layers have nodes with a very large number of connecting relationships.

Orthogonal – To have orthogonal relationships between CIs.	
Polyline – To have straight lines between CIs.	

- **Orthogonal Routing** – Routes edges horizontally and vertically.
  - **Fix Node Sizes** – For future use.
- **Polyline Routing** – Routes edges as one or more straight line segments with arbitrary angles. Polyline routed edges are prevented from overlapping by adding extra segments.
  - **Spacing Between Bends** – This setting is available only when you select **Routing** > **Polyline**. Polyline routing routes edges as one or more straight line segments with arbitrary angles. Path nodes are added automatically to keep the edges from overlapping.



- **Variable Level Spacing** – Adjusts the spacing between pairs of neighboring levels according to the density of edges between the levels.

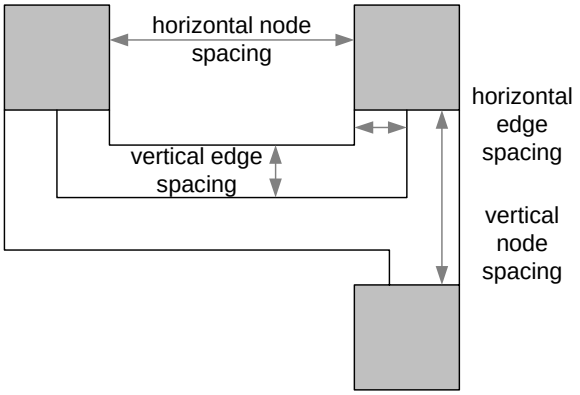
  If the drawing's edges are orthogonally routed, this ensures the required vertical spacing between horizontal edges (in a top-to-bottom or bottom-to-top layout) routed between levels.

  If the routing is polyline, variable level spacing makes it easier to distinguish between edges in very dense drawings.
- **Undirect Layout** – Edge direction is not used to build the levels of the hierarchical drawing.

Orthogonal Layout

The orthogonal layout routes relationships horizontally and vertically. This results in relationships bending at 90-degree angles only.

- **Horizontal Spacing –** Sets the minimum horizontal distance between CIs.

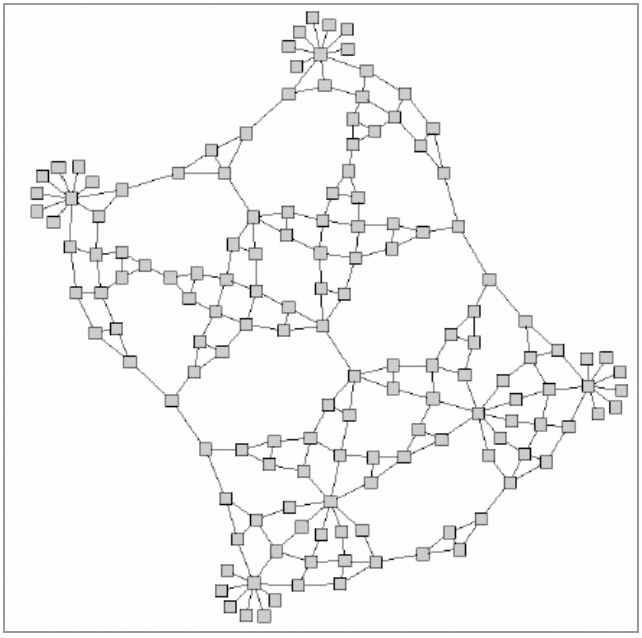


  - **Spacing Between Nodes** – Sets a space around each node in the layer. The default value is 20.
  - **Spacing Between Edges** – Sets a space between elements in the layer.
- **Vertical Spacing –** Sets the minimum vertical distance between CIs.
  - **Spacing Between Nodes** – Sets a space around each node in the layer. The default value is 20.
  - **Spacing Between Edges** – Sets a space between elements in the layer.
- **Layout Quality** – You can adjust the quality of the layout produced to fit your application's needs. The quality reflects the number of steps or the method used to produce the layout (for example: high quality is neater and cleaner but takes more time to layout).
  - **Draft**
  - **Default**
  - **Proof**

- **Aspect Ratio** – The layout's proportions. This only affects the placement of components with respect to each other, not the layout of each individual component. It has no effect if the graph is connected.
  - **Automatic** – The layout is performed automatically.
  - **Custom** – Specify the aspect ratio. The default is 1.0.
  - **Disabled** – Disables the aspect ratio.
- **Fix Node Size –** For future use.

Symmetric Layout

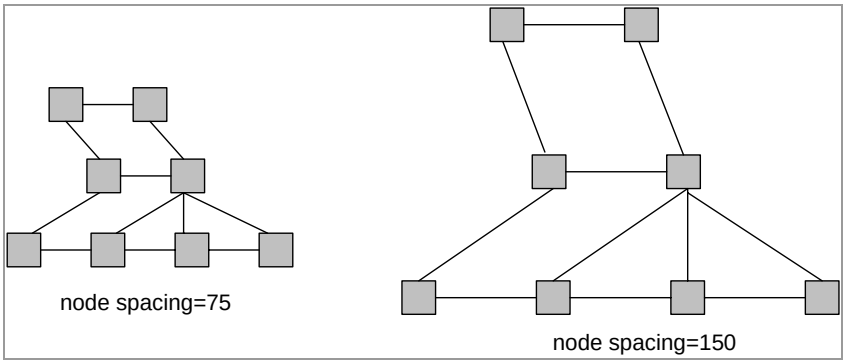
The symmetric layout emphasizes the symmetries that may occur in a graph.



This section describes the symmetric layout options that are available.

- **Spacing Options** – The symmetric layout allows you to adjust the spacing of nodes.

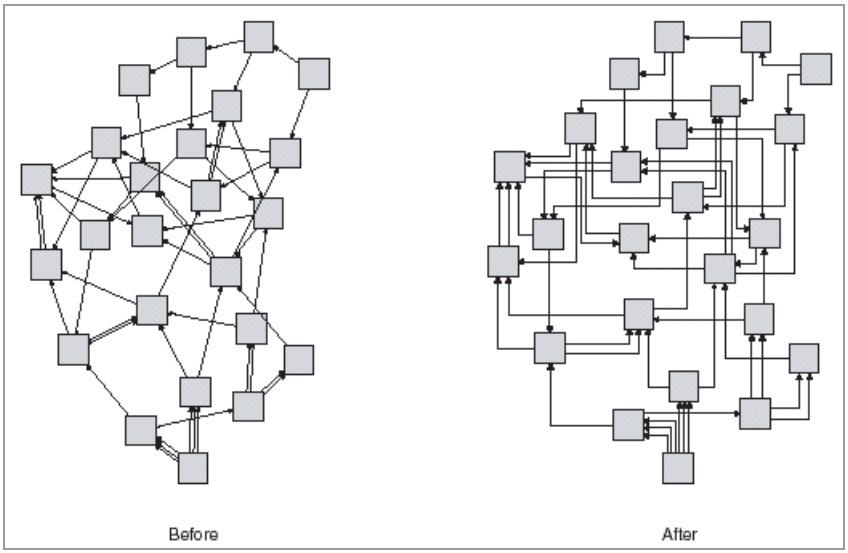
- **Node Spacing** – Allows you to set constant horizontal and vertical spacing around each node in the layer. The value you specify is a guideline for the layout, so that it is possible that the spacing for a particular pair of nodes might be different from the one you specified. The larger the node spacing you specify, the farther apart the nodes will be in the final layout. The default is 10.



- **Layout Quality** – You can adjust the quality of the layout produced to fit your application's needs. The quality reflects the number of steps or the method used to produce the layout (for example: high quality is neater and cleaner but takes more time to layout).
  - **Draft**
  - **Default**
  - **Proof**

Routing Layout

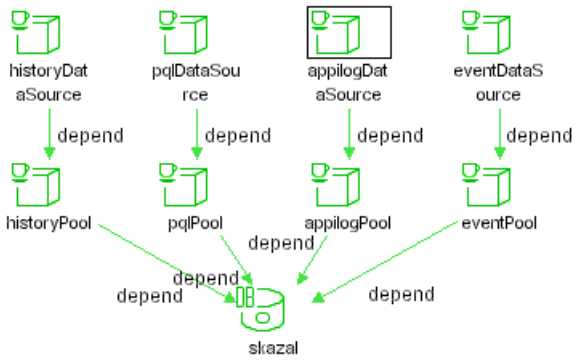
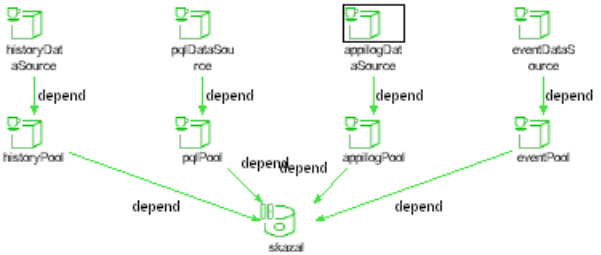
Routing produces drawings that leave nodes essentially where they are, and reroutes the edges orthogonally.



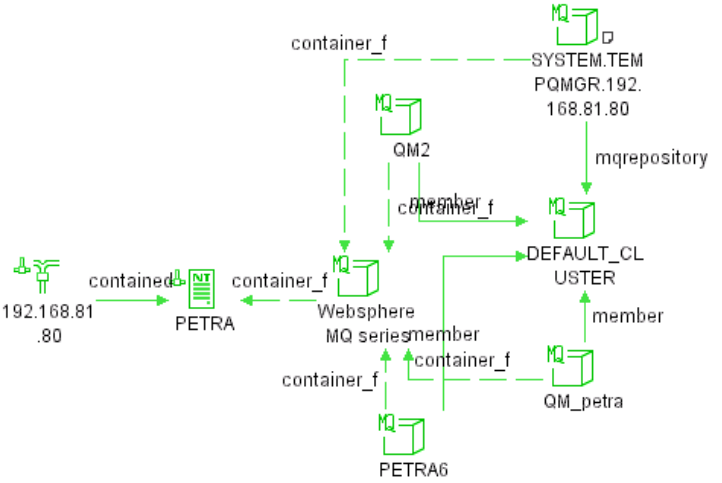
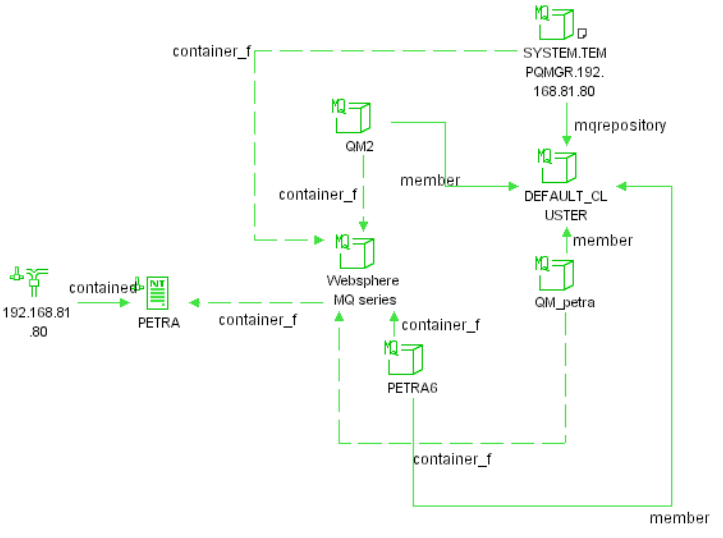
This section describes the routing layout options that are available.

➤ **Horizontal Spacing** – Sets the horizontal space around each node or each CI.

➤ **Spacing Between Nodes** – Sets the horizontal space around each node in the layer. The default value is 20.

Spacing Between Nodes = 20	
Spacing Between Nodes = 75	

➤ **Spacing Between Edges** – Sets the horizontal space between any two neighboring parallel edges. Edge spacing can also affect the size of a node to which many edges are attached, unless the node is set to remain fixed at its current size. The settings for edge spacing are constant values. For example:

Spacing Between Edges = 20	
Spacing Between Edges = 50	

- **Vertical Spacing** – Sets the vertical space around each CI.
  - **Spacing Between Nodes** – Sets the vertical space around each node in the layer. The default value is 20.
  - **Spacing Between Edges** – Sets the vertical space between any two neighboring parallel edges. Edge spacing can also affect the size of a node to which many edges are attached, unless the node is set to remain fixed at its current size. The settings for edge spacing are constant values.
- **Fix Node Sizes** – Select **Fix Node Sizes** to maintain the size of a node. Clear this option if you want the layout to increase the size of a node if it is necessary to maintain the specified relationship spacing when more than one relationship is attached to the same node side.
- **Fix Node Positions** – Select **Fix Node Positions** to maintain the position of all the nodes in the display. Clear the option if you want the layout to move the nodes if it is necessary to improve the layout. The movement is minimal, and avoids overlaps and minimizes bend points.

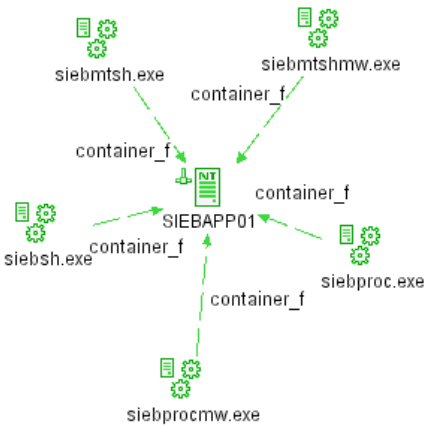
Understanding Layout Options

You can display the contents of the topology map using one of the following layout options. This section includes the following topics:

- Symmetric Layout – for details, see page 106
- Hierarchical Layout – for details, see page 106
- Orthogonal Layout – for details, see page 107
- Manual Layout – for details, see page 107

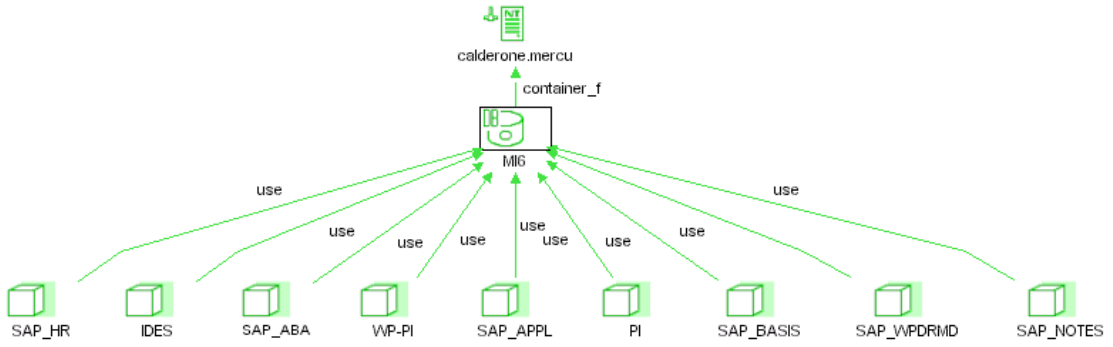
Symmetric Layout

Symmetric layout enables you to display clear representations of complex networks:



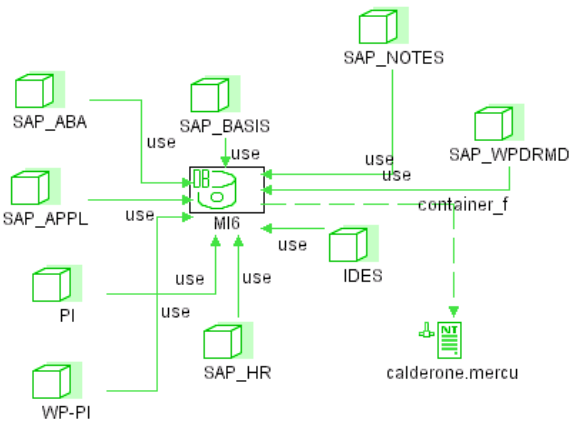
Hierarchical Layout

Hierarchical layout enables you to display the precedence relations in the topology map:



Orthogonal Layout

Orthogonal layout enables you to display views of outstanding clarity that are achieved by employing only horizontal and vertical edge routing.



Manual Layout

Save manual layout is relevant only to the topology map in Dashboard (see "Dashboard Topology Map" in *Using Dashboard*) and the topology map in IT Universe Manager (see "Working in Topology Map Mode" in *IT Universe Administration*).

Manual Layout enables you to manually change the topology map display, by dragging the displayed CIs and relationships to the desired location.

You can save the manual changes you made, by clicking the **Save** button on the toolbar. Thus, you will be able to reuse the layout you created.

Note: You can only save one manual layout. If another manual layout is already saved, after you would click the **Save** button a message would appear, asking you whether you want to replace the saved layout. Click **Yes** to save your new layout.

6

Working with Dynamic Node Factory

This chapter describes the concept of the Dynamic Node Factory (DNode) in Mercury Business Availability Center. It also describes how to define and edit a Dynamic Node Factory configuration item (CI), whether this is done from the IT Universe Manager tab or in the source adapters in the Source Manager tab.

This chapter describes:	On page:
What Is Dynamic Node Factory?	110
How Dynamic Node Factory Works	111
When to Use Dynamic Node Factory	112
Defining/Editing Properties for a Dynamic Node Factory CI	113
DNode Definition Key Property	120

What Is Dynamic Node Factory?

The Business Logic Engine is able to dynamically create CI hierarchies in the Mercury Universal CMDB in run time, based on incoming data samples on the Mercury Business Availability Center Bus. These hierarchies are created using the **Dynamic Node Factory** CI.

The Dynamic Node Factory CI includes a template that defines the rules for creating and managing the generated hierarchy, and a selector that defines which data samples are relevant.

When a Dynamic Node Factory CI is added to a service view, it can immediately start generating CIs from the incoming data samples. The generated CIs are added to the IT universe model in the CMDB, and to the subtree under the Dynamic Node Factory CI in the view. These generated CIs are then available to add to other views like any other CI in the CMDB.

A Dynamic Node Factory CI is added to a view and the IT universe model in one of the following ways:

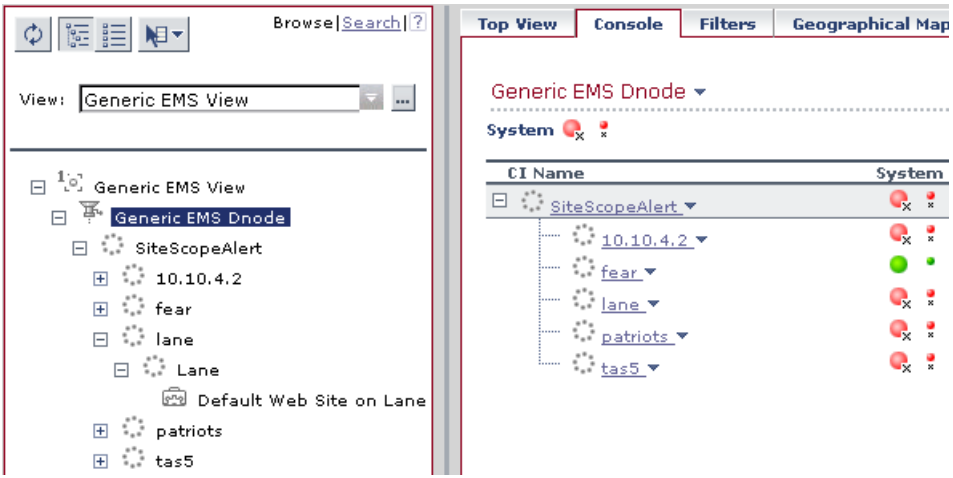
- **Automatically, by a source adapter** – Some of the Mercury Business Availability Center custom source adapters, for example, HP OpenView, include a Dynamic Node Factory definition. When the source is added in Source Manager, the Dynamic Node Factory CI is added to the out-of-the-box view for HP OpenView.

  For details on adding source adapters, see *Source Manager Administration*.

- **Manually, in IT Universe Manager** – You can create a new CI of type Dynamic Node Factory when building your view in the IT Universe Manager tab.

After a Dynamic Node Factory CI is added to a view, the CI properties can be edited in the IT Universe Manager tab. For details on creating a Dynamic Node Factory CI or editing its properties, see "Defining/Editing Properties for a Dynamic Node Factory CI" on page 113.

KPI status for the CIs in the Dynamic Node Factory hierarchy can be viewed in the Dashboard application as soon as the CIs are created.



How Dynamic Node Factory Works

The XML definition file generated by the Dynamic Node Factory CI defines rules for listening to the samples that arrive to the system, using selectors to define areas of interest. The XML file guides the Business Logic Engine on how to build the topology in the CMDB based on the incoming data, and produces its own definitions for the generated CIs.

Dynamic Node Factory CIs use the **DNode definition key** property, which defines how the samples are processed by providing example values for the XML file. These values are used as the template for the dynamic hierarchy in the CMDB. When required, you can customize the DNode definition key to produce a hierarchy that displays the information you need. For details, see “DNode Definition Key Property” on page 120.

If there is no existing CI for a received sample in the CMDB, then the Business Logic Engine creates the CI and adds it to the subtree for the Dynamic Node Factory CI. If the CI already exists in the CMDB (for example, if a BPM Transaction CI that corresponds to the sample was already created by the Business Process Monitoring source adapter), then the existing CI is added to the Dynamic Node Factory subtree, creating a new relationship for that CI.

Once a CI is created, the CI definitions are stored as for any other CI, and it receives update samples from the Dashboard matcher based on the selector for the Dynamic Node Factory CI. The Dynamic Node Factory hierarchy is rebuilt only as necessitated by the sample contents. (You can also define additional composite selectors for the Dynamic Node Factory XML definitions, so that the generated CI is updated independently from the Dynamic Node Factory generators. For details, see "Defining Composite Selectors in the DNode Definition Key" on page 129.)

You can add the generated CIs to other views, modify the CI properties, and delete the generated CIs from the CMDB. (However, after deleting a CI, it may be recreated by the Dynamic Node Factory based on incoming data samples.)

When to Use Dynamic Node Factory

Dynamic Node Factory hierarchies are useful for the integration and mapping of samples from external systems (generally EMS systems), where the database structure for the source is unknown, and Mercury Business Availability Center does not have a predefined source adapter for handling the data. However, the decision of whether to set up a Dynamic Node Factory hierarchy or a new source adapter to import the information depends on the particular situation and your data requirements.

The Dynamic Node Factory CI may also be used to bring alternative hierarchies from adapter sources.

You should take into consideration the following points:

- The adapter provides more precise results.
- Implementation of a Dynamic Node Factory template to import the information is usually simpler than setting up a new adapter for the source.
- The Dynamic Node Factory is dynamic, reflecting changes that generate new CIs in run time, while the adapters reflect the changes only after synchronization. For example, if you create a Dynamic Node Factory CI for alerts, you can generate a hierarchy that displays alerts with event children, without knowing in advance what type of alert will be received. The Dynamic Node Factory hierarchy will display a new alert when it occurs, while an adapter for alerts will display a new alert only after synchronization occurs (by default, once an hour).

  However, if an alert is removed at the source, the adapter will reflect this change at synchronization, while the Dynamic Node Factory may take up to 24 hours to reflect the change (depending on the Dynamic Node Factory cleaning definitions). Similarly, if a transaction name is changed or moved to a new profile in Monitor Administration, then the Dynamic Node Factory will create a new CI for it, but the old CI will remain in the hierarchy until it has not had any corresponding samples for 24 hours (default value, can be changed to a different value).

Defining/Editing Properties for a Dynamic Node Factory CI

You can edit the properties for a predefined Dynamic Node Factory CI in your view, or define a new Dynamic Node Factory CI to add to your view, in order to generate a hierarchy that fits your organization's needs. You can modify the mapping to the samples that create the hierarchy, and the rules and logic used to import the data and present it in Dashboard.

(It is also possible to customize the template for a predefined Dynamic Node Factory CI in the custom source adapter, but it is recommended that you customize the CI in the view, rather that in the adapter.)

Keep in mind that you affect two levels of CIs when you modify properties for a Dynamic Node Factory CI:

- The general properties for the Dynamic Node Factory CI, for example, location and context menu, apply only to that CI. Also, when you attach KPIs to the Dynamic Node Factory CI in Dashboard Administration, they are defined only for that CI. Modifying these properties has no influence on the generated hierarchy.
- The CIT-specific properties for the Dynamic Node Factory CI, for example, selector expression and KPI status for removal, define information relating to the CIs in the generated hierarchy under the Dynamic Node Factory CI. Modifying these properties changes the templates and handling rules for the hierarchy.

  Changes to these properties should be correlated with the definitions within the DNode definition key. For example, the selector expression must identify samples that contain the keys defined in the generators, otherwise the Dynamic Node Factory hierarchy will be empty.

Note: If you change the DNode definition key or selector expression for an active (enabled) Dynamic Node Factory CI, then the previous Dynamic Node Factory hierarchy is deleted from the view and the hierarchy is recreated.

If the catchment area for a Dynamic Node Factory hierarchy is too wide, you can modify the selector expression for the Dynamic Node Factory CI to narrow down the definition, for example, to include only samples coming from a particular machine.

The following procedure describes how to define properties for a Dynamic Node Factory CI in the IT Universe Manager tab, whether you are creating or editing the CI.

Note: There are no default KPIs defined for the Dynamic Node Factory CI; you must manually attach KPIs after creating the CI, or no status will be displayed in Dashboard.

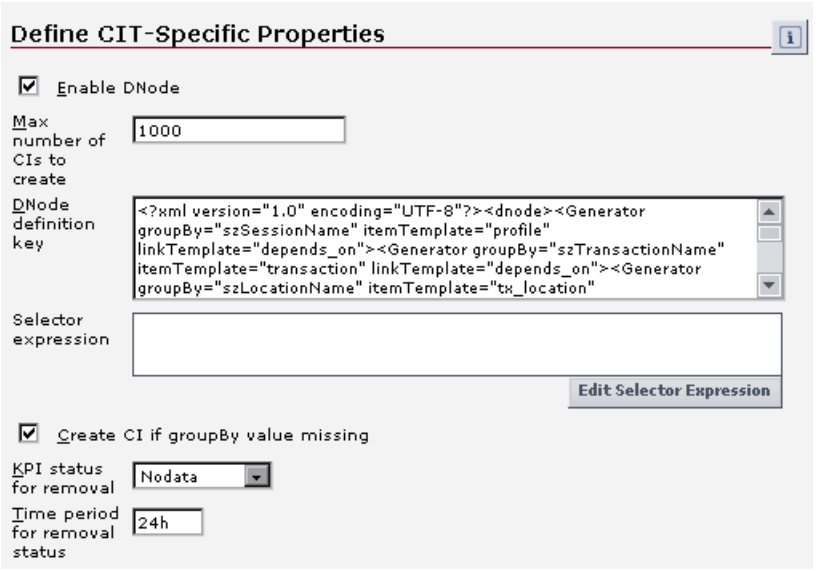
To define/modify a Dynamic Node Factory CI:

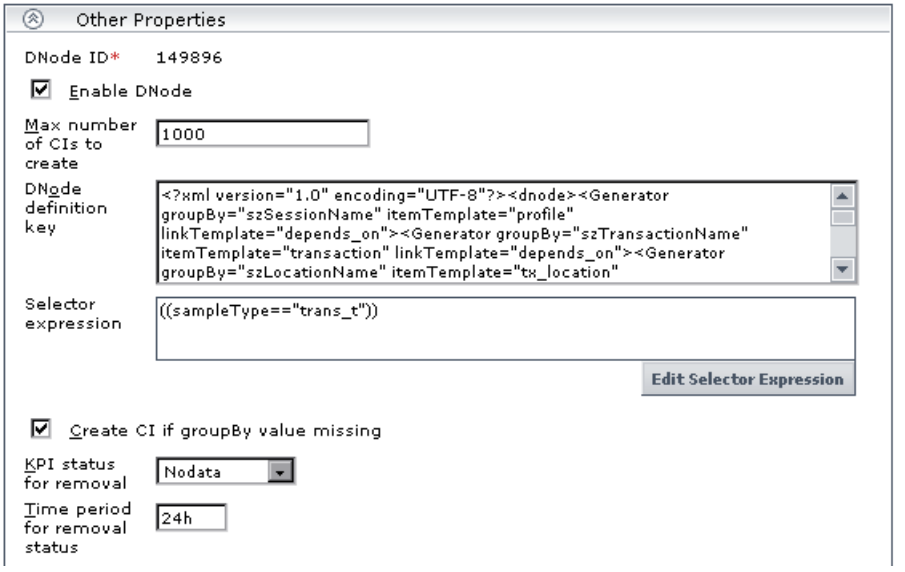
1. To create a new Dynamic Node Factory CI: In the **IT Universe Manager** tab, open the New CI Wizard as described in "Creating a New CI" on page 40, and select **Dynamic Node Factory** in the Select Configuration Item Type page.

   To edit properties for an existing Dynamic Node Factory CI: In the **IT Universe Manager** tab, select the CI in View Explorer and open the **Properties** tab.

2. Define properties in the Define General Properties page (or edit in the **General Properties** area), as required. These properties apply only to the Dynamic Node Factory CI, not to the CIs that will be created in the hierarchy. For details on these properties, see "General Properties for CIs" on page 66.

3 Define properties in the Define CIT-Specific Properties page (or edit in the **Other Properties** area), as required:





- **Enable DNode** – (Selected by default.) When the check box is selected, the Dynamic Node Factory hierarchy is updated as required based on the incoming samples. If the check box is deselected, the hierarchy is not updated (but the KPIs for the CIs will still receive status updates in Dashboard).
- **Max number of CIs to create** – Defines the limit for creating CIs in the Dynamic Node Factory hierarchy, default is**1000**. When a sample is received and the number of CIs in the hierarchy is below 1000, then the Dynamic Node Factory will add new CIs to the hierarchy as required by the sample, even if the number of CIs then exceeds 1000. However, if the hierarchy already has more than 1000 CIs when a sample is received, no new CIs will be created.

  If you think that the number of CIs might be close to or exceeding the defined limit, check the **<Mercury Business Availability Center root directory>\MercuryAM\log\EJBContainer\Trinity.log** file. If there is a problem, either change the limit in the **Max number of CIs to create** property, or modify the selector expression to narrow down the definition. (Also take into consideration the overall CI limit for the IT Universe Manager tab.)
- **DNode definition key** – Defines how the received data samples are processed to generate the hierarchy. For details, see "DNode Definition Key Property" on page 120.

Note: The default DNode definition key is for a Business Process Monitor source.

- **Selector expression** – Defines the filter that catches the samples on the Bus that are required for the Dynamic Node Factory. These samples are processed according to the generator definitions in the DNode definition key property. For example, if you want the generators to be applied against all Business Process Monitor samples, you would define a selector for sampleType "trans_t". If you want to narrow the catchment area, narrow the filter by adding additional selector expression rows.

To define or edit the selector, click **Edit Selector Expression**. For details on defining selector expressions, see "Selectors for KPIs" in *Application Administration*.

Note: If you change values in the DNode definition key property, you should change the selectors accordingly. The selectors must identify samples that contain the keys defined in the generators, otherwise the Dynamic Node Factory hierarchy will be empty.

- **Create CI if groupBy value missing** – (Selected by default.) Defines how to handle samples that are missing the grouping (**groupBy**) key. When the check box is selected a CI will still be created for the sample. (If the **nameBy** key is also missing, the CI is given the default name **Empty**.)
- **KPI status for removal** – Defines at which KPI status the generated CIs can be considered for removal from the CMDB (default is **Nodata**). If all KPIs for a CI have held this status for at least 24 hours (default period), the CI is eligible to be removed from the CMDB. Once an hour, the Dynamic Node Factory cleaner removes the CIs that are marked as removable.

  This option enables automatic cleaning of CIs that have not received corresponding data samples for 24 hours. You can select a different status from the **KPI status for removal** list according to the characteristics of the generated CIs. For example, you may have generated ticket CIs that change to status green (**OK**) when the ticket is closed; after 24 hours at OK status these CIs are considered no longer relevant for your views.
- **Time period for removal status** – Defines the time period during which all KPIs for a CI must have the status defined in **KPI status for removal**, before the CI can be marked as removable (default is 24 hours). If you want to change the value, enter a number followed by one of the following suffixes: **s** - for seconds, **m** - for minutes, **h** - for hours, **d** - for days.

Note: The Dynamic Node Factory cleaner is set to review Dynamic Node Factory content and remove expired CIs once an hour. If the value set for the **Time period for removal status** property is less than an hour, then the actual removal time may still be up to an hour, when the next cleaner review takes place.

If required, you can change the default value for the **Dynamic Node Factory clean interval** parameter on the Infrastructure Settings page (**Admin** > **Platform** > **Setup and Maintenance** > **Infrastructure Settings** > context **Foundations** > select **Online Business Logic Engine** from the dropdown list). However, the cleaning values should be changed with care, to avoid disrupting the sensitive balance of the cleaning mechanism.

---

4 If you are defining a new CI, complete the wizard as described in "Creating a New CI" on page 40.

If you are editing a CI, click **Apply**.

5 To define KPIs for the Dynamic Node Factory CI, move to **Admin** > **Dashboard** > **KPIs**. Define KPIs and business rules for the Dynamic Node Factory that are logical for the data source, for example, **Availability** and **Performance** for a Business Process Monitor source. You should select a group-type business rule for teach KPI.

For information on defining KPIs, see "Configuring KPIs" in *Application Administration*.

Note: The dimensions defined for a parent Dynamic Node Factory CI should correlate with the dimensions defined within the DNode definition key property.

DNode Definition Key Property

The DNode definition key property defines how the received data samples are processed to generate the hierarchy under the Dynamic Node Factory CI. This section describes the DNode definition key structure and contents, and how to customize the property.

This section contains the following topics:

- "About the DNode Definition Key" below
- "Components of the DNode Definition Key" on page 121
- "Defining Composite Selectors in the DNode Definition Key" on page 129

Important note: Only advanced users should make changes to templates. An incorrect definition may lead to loss of data, the wrong hierarchy build, and redundancy in the CMDB.

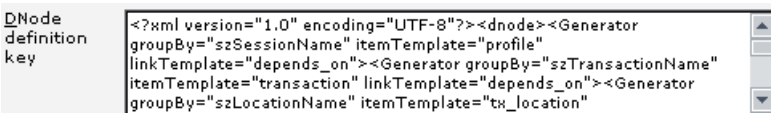
About the DNode Definition Key

The DNode definition key defines the properties for the CIs created by the received samples. This includes generators to determine how to group the incoming samples, how many CI levels to create, and which templates to use when creating each CI (KPIs, business rules, and so forth). Each different data type needs an appropriate definition in the Dynamic Node Factory file.

You can customize the definitions in the DNode definition key to make them relevant for a specific data source and to build the hierarchy structure you require. This customization can be done by:

- editing the Dynamic Node Factory CI in the IT Universe Manager tab (recommended method). The procedure for editing a Dynamic Node Factory CI is described in "Defining/Editing Properties for a Dynamic Node Factory CI" on page 113.
- editing the template for the source adapter in Source Manager (only relevant for predefined Dynamic Node Factory CIs). For information on editing adapter templates, see "Working with Templates" in *Source Manager Administration*.

The default DNode definition key property (used when you create a new Dynamic Node Factory CI in the IT Universe Manager tab) contains definitions for a Business Process Monitor hierarchy:



If you are adding a Dynamic Node Factory CI for a different data source, for example, SiteScope, you must replace the definitions with ones appropriate for the source.

For explanations of the data samples and their fields, see “Samples” in *Reference Information*.

For descriptions of the definitions contained in the DNode Definition Key, see the following section.

Components of the DNode Definition Key

In order to customize a DNode Definition Key, you need some understanding of its various components.

All changes to the parameters in the DNode definition key property must be logical and relevant for the source.

Generators

The first part of the DNode definition key contains the **Generators** defining the hierarchy for the CIs. Each generator in the DNode definition key defines a level to be built in the Dynamic Node Factory hierarchy (the XML provides for definitions of up to three levels). For example, for Business Process Monitor, the Dynamic Node Factory may build levels for profile, transaction, and location.

For example, the generators in the default DNode definition key for Business Process Monitor are as follows:

```
<Generator groupBy="szSessionName" itemTemplate="profile"
linkTemplate="depends_on">
<Generator groupBy="szTransactionName" itemTemplate="transaction"
linkTemplate="depends_on">
<Generator groupBy="szLocationName" itemTemplate="tx_location"
linkTemplate="monitored_by"/>
```

- **groupBy** – defines the property to look for in the samples, to see if the sample is relevant for building this level of the hierarchy. A CI is created for each unique value of this key. For example, all samples with a unique value for szSessionName will create a CI at the profile level of the hierarchy.

Note: If samples are missing this key, a CI is still created for the sample if the **Create CI if groupBy value missing** is selected for the parent Dynamic Node Factory CI. For more information, see "Defining/Editing Properties for a Dynamic Node Factory CI" on page 113.

- **itemTemplate** – defines the name of the CI template to be used for creating each CI at this level. The CI templates are defined in the DNode definition key after the generators.
- **linkTemplate** – defines the name of the relationship template to be used for each CI created at this level. The relationship template defines the relationship between the created CI and its parent CI. The relationship templates are defined in the DNode definition key after the CI templates.

Note: It is possible to use the DNode definition key property to build more complex dynamic hierarchies, for example, by using more than one generator at the same level. For more information, contact Mercury Customer Support.

CI Templates

The second part of the DNode definition key contains the **ItemTemplates**, the templates for creating the CIs. Each template defines attributes, KPIs, and business rules to be used for each generated CI. (If required, you can also add a composite selector to a monitor ItemTemplate. For details, see "Defining Composite Selectors in the DNode Definition Key" on page 129.)

For example, the following ItemTemplate creates Business Process Group CIs for the profiles in the default Business Process Monitor hierarchy:

```
<ItemTemplate template="profile" namespace="Mercury" class-
name="bp_group">

    <Attributes>
        <Attribute name="data_name" ref_value="szSessionName"/>
        <Attribute name="id" ref_value="u_iSessionId"/>
        <Attribute name="app_id" value="-1"/>
    </Attributes>

    <KPI>
        <Attributes>
            <Attribute name="type" value="6"/>
            <Attribute name="rule" value="1"/>
        </Attributes>
    </KPI>

    <KPI>
        <Attributes>
            <Attribute name="type" value="7"/>
            <Attribute name="rule" value="1"/>
        </Attributes>
    </KPI>

    <!--Specify objectives only if you want to override the default values for
objectives-->

</ItemTemplate>
```

Item Template Line

The first line of the ItemTemplate defines the following parameters:

```
<ItemTemplate template="profile" namespace="Mercury" class-
name="bp_group">
```

- **template** – defines the name of the template.
- **namespace** – defines the context for the class-name: **Mercury** is used for all CIs.
- **class-name** – defines the configuration item type (CIT) name for the CIs generated by this template. This must be a valid CMDB CIT. The attributes for the created CIs are taken from the CIT definition.

  This is the internal name used for the CIT, not the display name. For example, the internal name for Business Process Group CIs is **bp_group**.

  To view the internal name for each CIT: Open the **Admin** > **CMDB Administration** > **CI Type Manager** page and select **Edit CIT** from the right-click menu for a CIT to display the Edit Configuration Item Type window. In the **Information** tab, the internal name is displayed in the **Configuration Item Type Name** field.

Attributes Section

Most of the CITs have mandatory attributes. These attributes must be included in the **Attributes** section of the ItemTemplate (in the following example, the **id** and **app_id** are mandatory attributes).

The Attributes section defines the following parameters:

```
<Attributes>
        <Attribute name="data_name" ref_value="szSessionName"/>
        <Attribute name="id" ref_value="u_iSessionId"/>
        <Attribute name="app_id" value="-1"/>
</Attributes>
```

- **name** – defines the internal name of an attribute included in the attributes list for the relevant CIT. For example, **data_name** is the internal attribute name corresponding to the **Name** property for the created CI.

To view the list of attributes defined for a CIT: Open the **Admin** > **CMDB Administration** > **CI Type Manager** page and select **Edit CIT** from the right-click menu for a CIT to display the Edit Configuration Item Type window. Click the **Attributes** tab to see the list of attributes—the **Name** column lists the attributes by their internal name.

- **ref_value** – defines a key in the data sample that will provide the value for the attribute. In the example <Attribute name="id" ref_value="u_iSessionId"/>, the **id** attribute for each CI is given the value from the property **u_iSessionId** in the incoming samples.
- **value** – defines a static value for the attribute. This enables you to use a predefined value for all CIs generated by this ItemTemplate. For example, you can define a static value attribute for **data_name**, where you know that there will only be one CI at this level: <Attribute name="data_name" value="my_BPM_profile"/>

Note: The Business Process Group CI template includes the static value attribute <Attribute name="app_id" value="-1"/>. This **app_id** attribute corresponds to the Business Process Monitor source adapter, and is only required for Business Process Monitor hierarchies.

Every ItemTemplate should contain attributes defining a unique name and an appropriate ID for each generated CI (otherwise the CIs will be created on top of each other). You can define additional attribute keys for the template, as required.

If you want to define a context menu in the ItemTemplate, so that it will apply for all CIs generated by the template, you need to use a special string format. The value for the context menu attribute is presented as XML, and so must be “escaped” from the DNode definition key XML, as in the following example:

<Attribute name="contextmenu" value="<Values><Value>**groupMenu**</Value></Values>"/>

The value in bold shows the internal context menu name. (You can find the internal context names in: Mercury Business Availability Center server\AppServer\webapps\site.war\bam\conf\repositories\contextMenu\def.xml)

KPI Section

The **KPI** section of the ItemTemplate defines the following parameters:

```
<KPI>
    <Attributes>
        <Attribute name="type" value="6"/>
        <Attribute name="rule" value="1"/>
    </Attributes>
</KPI>
```

- **name="type"** – defines the KPI to be attached to the CIs generated by this ItemTemplate. The value corresponds to the ID number for a KPI, for example, **6** is the ID number for the **Performance** KPI. The ID numbers for the KPIs are listed in the **Admin** > **Dashboard** > **Repositories** > **KPIs** page, under the **ID** column.

  Each ItemTemplate in the DNode definition key should specify the same KPIs, and these should correlate with the KPIs defined for the Dynamic Node Factory CI itself.

- **name="rule"** – defines the business rule to be used with the KPI for the generated CIs. The value corresponds to the ID number for a rule, for example, **1** is the ID number for the **Worst Child Rule**. The ID numbers for the business rules are listed in the **Admin** > **Dashboard** > **Repositories** > **Business Rules** page, under the **ID** column.

  The specified business rule should be a monitor rule if the ItemTemplate is defining leaf CIs, and a group rule if the ItemTemplate is defining non-leaf CIs. For more information on rule types, see "About Business Rules" in *Application Administration*.

  If required, you can define additional parameters in the KPI section. Attributes can use **value** or **ref_value**, as described in the attributes section on page 124.

The following additional parameters are relevant for leaf monitor CIs defined in the DNode definition key, and should be declared as part of the KPI:

- **RuleArguments** – defines parameters used by the business rule. When you add a rule argument to a KPI definition in the DNode definition key, it overrides the default value for the parameter defined in the Rules Repository. For example, the following definition will override the default value for the **No data timeout** parameter (used with, for example, the Transaction Availability Rule), and the generated CIs will use the new value for the KPI:

  ```
  <RuleArguments><parameter name="No data timeout" type="Integer"
  value="1200"/></RuleArguments>
  ```

  For details on the parameters used by the business rules, see "Business Rules Repository" in *Application Administration*.

Note: If you add definitions for **RuleArguments**, make sure to define the exact key name and to use the correct format for the value.

- **Objective** – defines objective values that override the default objective parameter values declared in the Rules Repository. The objective values should be relevant for the rule.

For example, the following definition will override the default value for the **Critical** (red) objective level (Objectives section shown in bold):

```
<KPI>
    <Attributes>
        <Attribute name="type" value="6"/>
        <Attribute name="rule" value="13"/>
    </Attributes>
    <RuleArguments/>
    <Objective>
        <Attributes>
            <Attribute name="critical" value="12"/>
        </Attributes>
    </Objective>
</KPI>
```

Note: The KPI definitions should be in the order shown above: KPI, business rule, rule arguments, objectives.

Relationship Templates

The final part of the DNode definition file contains the **LinkTemplates**, the templates for creating the relationships between the nodes.

For example, the LinkTemplates in the default DNode definition key for Business Process Monitor are as follows:

```
<LinkTemplate template="depends_on" namespace="Mercury" class-
name="depends_on"/>
</LinkTemplate>

<LinkTemplate template="monitored_by" namespace="Mercury" class-
name="monitored_by">
</LinkTemplate>
```

- **template** – defines the name of the template.
- **namespace** – defines the context for the class-name: **Mercury** is used for all CIs.

➤ **class-name** – defines the internal name for the relationship generated by this template, for example, **depends_on**. The attributes for the created relationship are taken from the definitions in the CI Type Manager.

To view the internal name for each relationship: Open the **Admin** > **CMDB Administration** > **CI Type Manager** page and select **Edit CIT** from the right-click menu for a relationship to display the Edit Configuration Item Type window. In the **Information** tab, the internal name is displayed in the **Configuration Item Type Name** field.

If required, you can define attributes for the relationships; the values will override the default values for the relationship properties.

Defining Composite Selectors in the DNode Definition Key

The Dynamic Node Factory uses the selector expression defined for the Dynamic Node Factory CI to receive samples from the Bus and create the CIs in the hierarchy. If required, you can define a **composite selector** for the monitor ItemTemplate (leaf level) in the DNode definition key, so that the new selector is used to receive update samples for KPI status, rather than the original selector that generated the CI.

Composite selectors are used when you want the created CI to be independent from the Dynamic Node Factory generators. This may be required when the original selector expression information is not included in later samples.

For information on defining selector expressions, see “Selectors for KPIs” in *Application Administration*.

You can use static or dynamic attributes in composite selector expressions. When using static attributes, you define a known value for a known referenced property, for example:

```
<Selector key="sampleType" op="EQ" type="String" value="ss_t" />
```

Note: The attributes in a selector are case sensitive, so take care to define the attributes exactly as in the samples.

If you do not know the value of a selector attribute in advance, you can use a dynamic selector expression. The value for the dynamic attribute is defined only when the sample is received.

You make an attribute dynamic by using the prefix **ref_** with the attribute name. For example:

```
<Selector key="u_iSessionId" op="EQ" type="Long" ref_value="u_iSessionId" />
```

In this example, the **value** attribute is dynamic, so the value for the referenced property will be taken from the **u_iSessionId** field in the received sample.

You can define **ref_value**, **ref_key**, **ref_type**, and **ref_op** in dynamic selector expressions, and each dynamic selector expression can contain more than one dynamic attribute.

Note: The dynamic selector expression option is not relevant when working with custom data (fully descriptive samples are received from the source, so there is no problem of insufficient information).

The following example shows part of the ItemTemplate for a SiteScope monitor with an added composite selector (shown in bold), including dynamic selector expressions:

```
<ItemTemplate template="monitor" namespace="Mercury" class-
name="sitescope_monitor">
    <Attributes>
        …
    </Attributes>
    <CompositeSelector logicalOp="AND">
        <Selector key="sampleType" op="EQ" type="String" value="ss_t" />
        <Selector key="u_iSessionId" op="EQ" type="Long"
        ref_value="u_iSessionId" />
        <Selector key="u_iMonitorId" op="EQ" ref_value="u_iMonitorId" />
    </CompositeSelector>
    <KPI>
        …
    </KPI>
</ItemTemplate>
```

Note: If necessary, you can define a different composite selector for a KPI in the ItemTemplate, if the KPI requires a different selector from the default for the monitor.

7

Example Scenario for an Instance View

This chapter provides a scenario for setting up an instance view to use in Dashboard and Service Level Management.

This chapter describes:	On page:
About the Example Scenario	134
Stage 1: Create the View	136
Stage 2: Build the View	136
Stage 3: Customize KPI Settings	160
Stage 4: Define Viewing Permissions	168
Stage 5: See KPI Status in Dashboard	173

About the Example Scenario

The example scenario is for creating an instance view that shows KPI status for lines of business.

In this scenario, a new view is created for A.A. Web Tours to show real-time performance for the company's main lines of business, and over time impact of network problems on major customers.

Note that this scenario provides an example for setting up a line of business view with customized KPI settings. The actual steps performed and parameters defined when setting up a view will vary according to your organization's requirements.

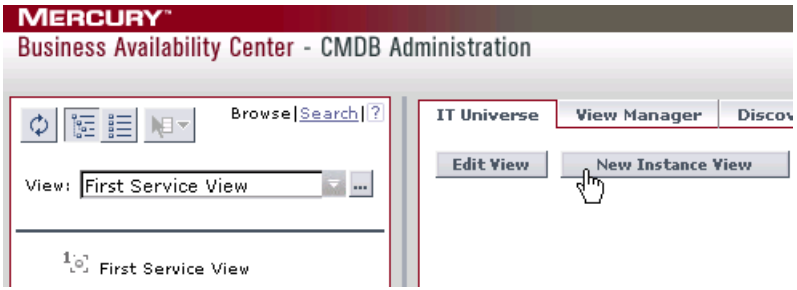
Stage 1: Create the View

The first stage is creating the framework for the view. In this scenario, the new view is created from the IT Universe Manager tab.

Note: It is also possible to create instance views from the View Manager tab.

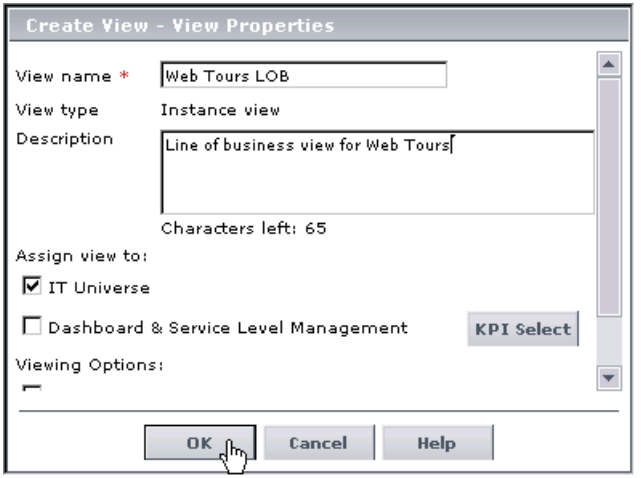
To create the new view in the IT Universe Manager tab:

1 In the **Admin** > **CMDB** > **IT Universe Manager** tab, click **New Instance View**.



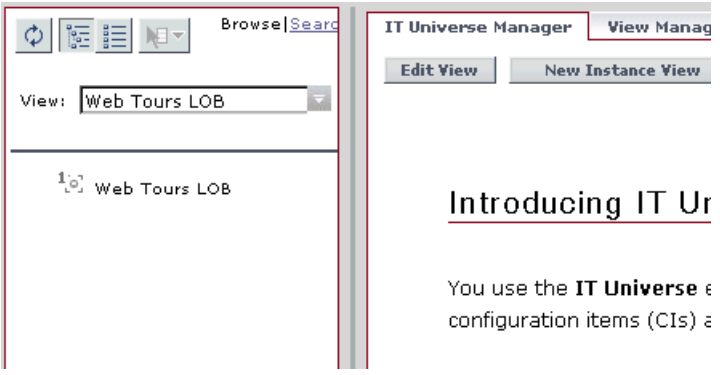
The Create View - View Properties window opens.

2 Enter a logical **View name** (for example, Web Tours LOB) and a **Description**. In the **Assign view to** options, select **IT Universe** (so that the view can be edited in the IT Universe Manager tab) and deselect **Dashboard & Service Level Management** (so that the view will not be available in these applications until it is ready).



Click **OK**.

3 The view is created and automatically opens in View Explorer.



You can immediately continue with building the view, as described in "Stage 2: Build the View" on page 136.

Note: The new view is created by default in the same folder as the original view that you had open in View Explorer. At any stage after creating the view, you can open the View Manager tab and move the view into a different folder.

Stage 2: Build the View

The second stage of the scenario is populating the view with CIs.

For this scenario, the hierarchy created for A.A. Web Tours lines of business is:
Line of business > customer > service > host > monitor

The first three levels are logical CIs that are manually created in the view. The fourth level are CIs added to the CMDB by the discovery process. The final level are monitor CIs added to the CMDB by the source adapters.

This stage of the scenario describes how to add the CIs to the view using the Default viewing mode (all steps could also be performed using Topology Map viewing mode), and using the different edit options available in the IT Universe Manager tab. However, all the steps described in this stage could also be performed from the Instance View Editor (described in "Editing with the Instance View Editor" on page 28).

Building the view is divided into the following sections:

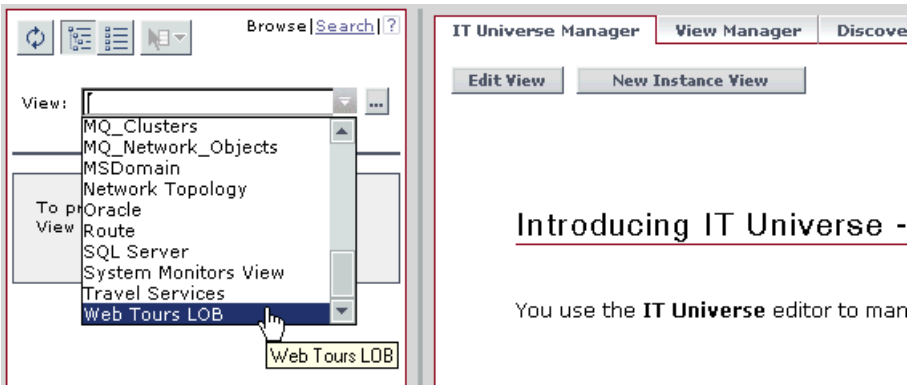
- Create Root CIs in the View, described below
- Create Logical CIs Related to CIs in the View, described on page 141
- Attach CIs from Discovery Process to CIs in the View, described on page 145
- Attach Monitoring CIs to CIs in the View, described on page 152

Create Root CIs in the View

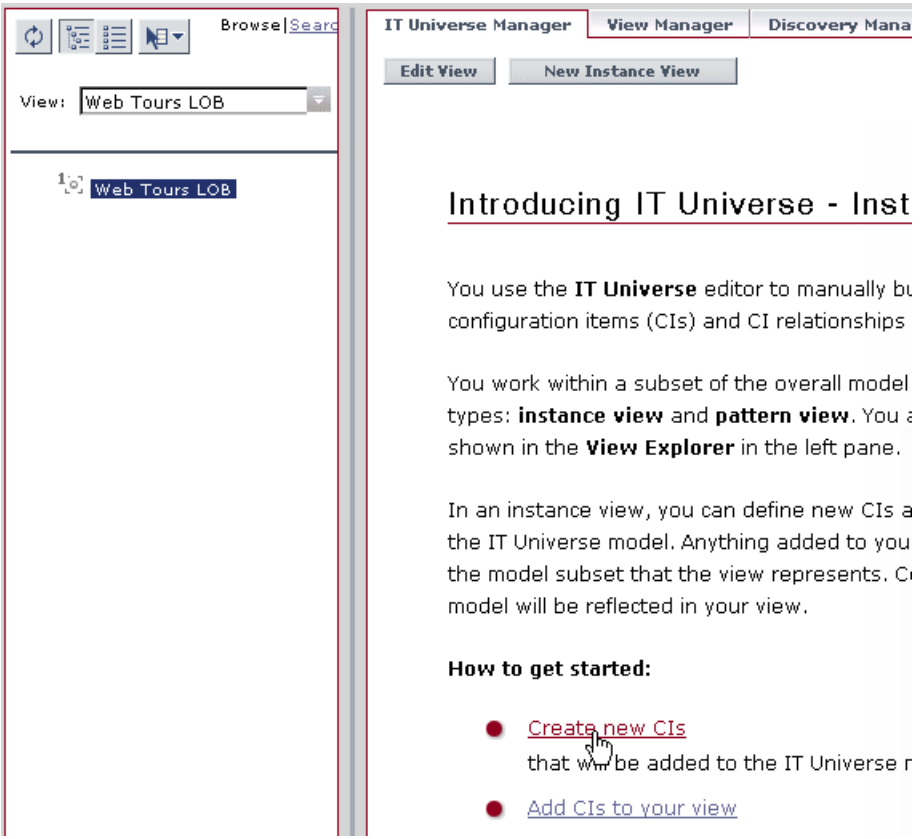
This section describes the creation of the line of business CIs in the views, to be the root CIs for each subtree. These new logical CIs are added to the Web Tours view and the CMDB without any relationship to any other CI.

To create root CIs in the view:

1 In the **Admin** > **CMDB** > **IT Universe Manager** tab, open the created view in View Explorer (left pane).

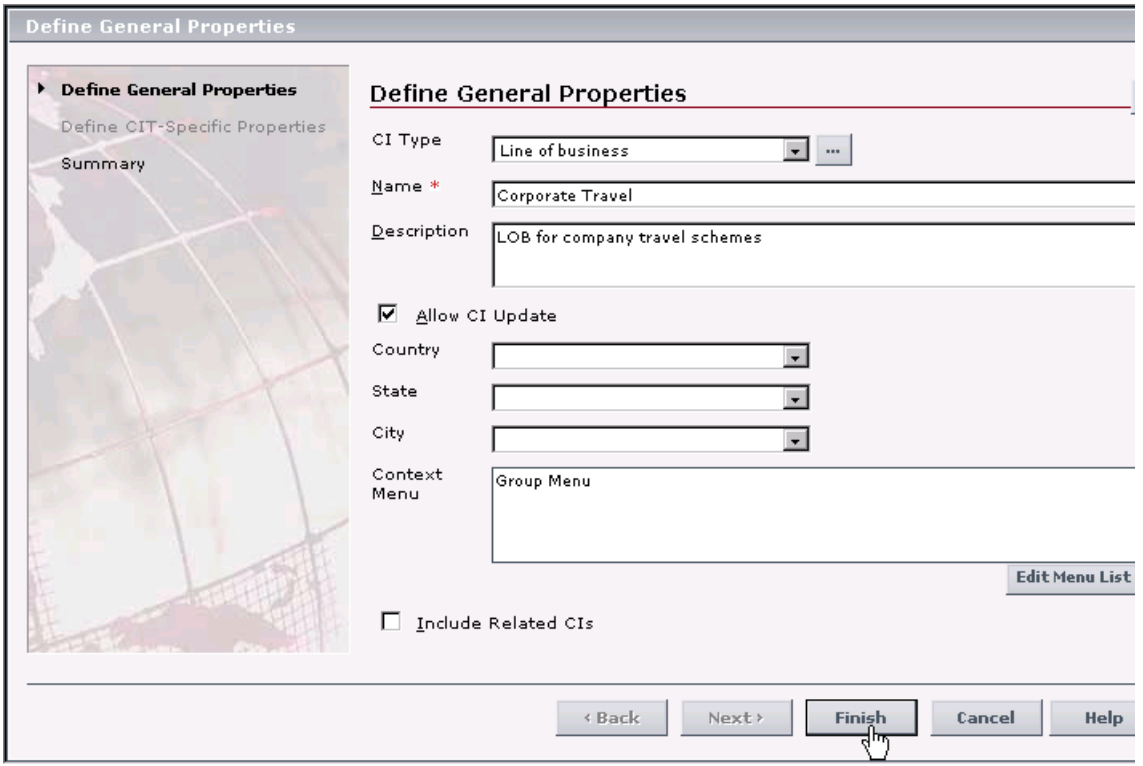


2 Create the Line of Business CIs for the root level of the hierarchy. With the view name selected, either select **New CI** from the right-click menu for the view name, or click **Create New CIs** in the right pane.



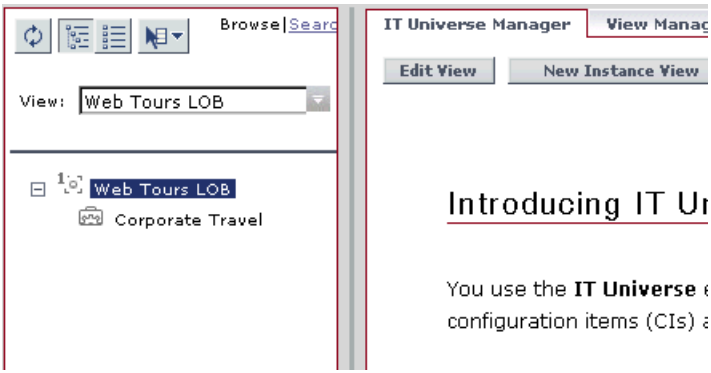
The New CI Wizard opens at the Define General Properties page.

3 Select **Line of business** from the **CI Type** list. Enter the CI **Name** (for example, Corporate Travel) and **Description**.

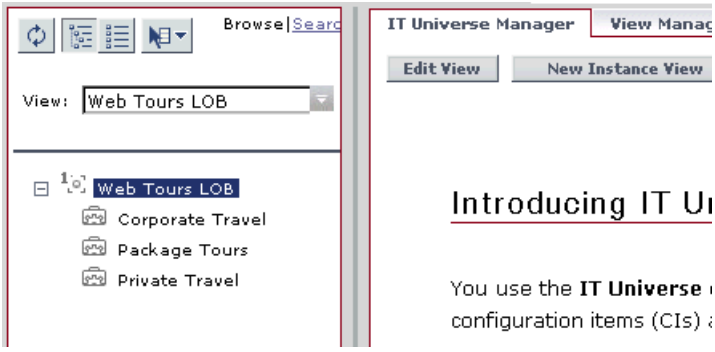


Click **Finish**.

4 The Corporate Travel CI appears as a root CI in the Web Tours LOB view.



Repeat steps 2 and 3 to create root CIs for the other lines of business: Package Tours and Private Travel.



Create Logical CIs Related to CIs in the View

This section describes the creation of the customer CIs that are attached to the existing LOB CIs in the view, creating relationships between the CIs.

To create related logical CIs in the view:

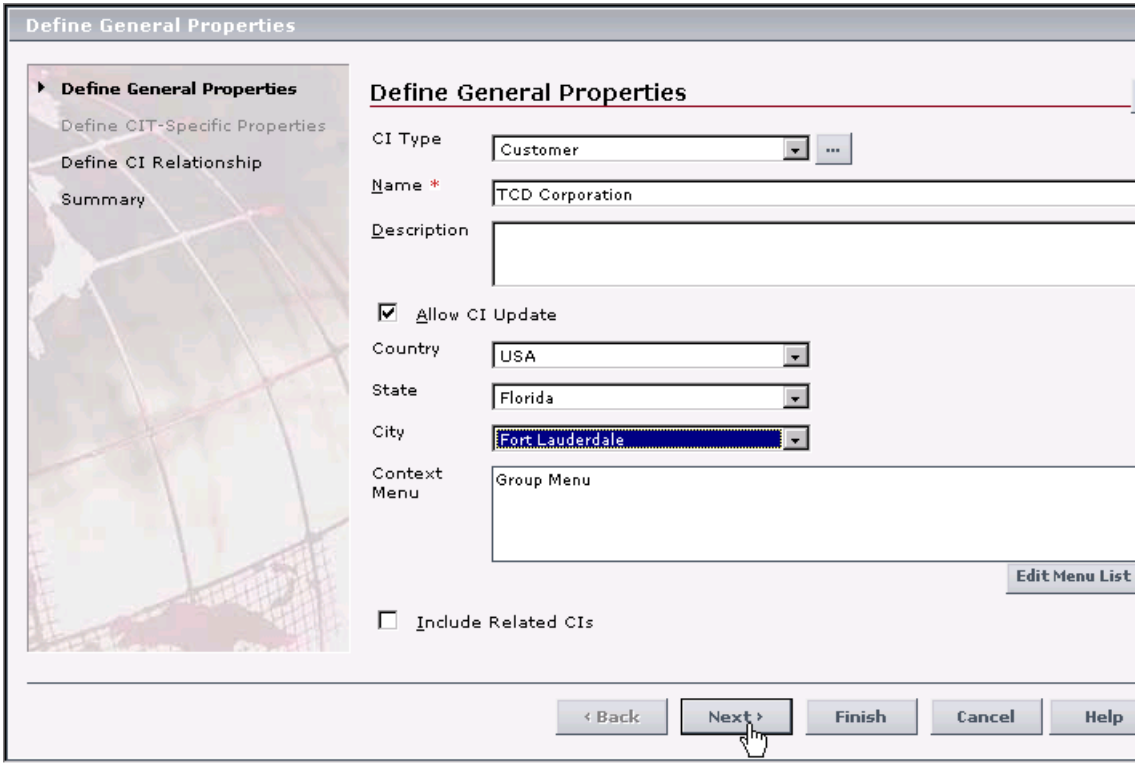
1 Select the CI to attach to in View Explorer (for example, Corporate Travel), and then either select **New Related CI** from the right-click menu for the CI, or click the **New Related CI** button in the right-hand pane.



Note: You must be working in the IT Universe Manager **Related Configuration Items** tab to see the buttons in the right-hand pane.

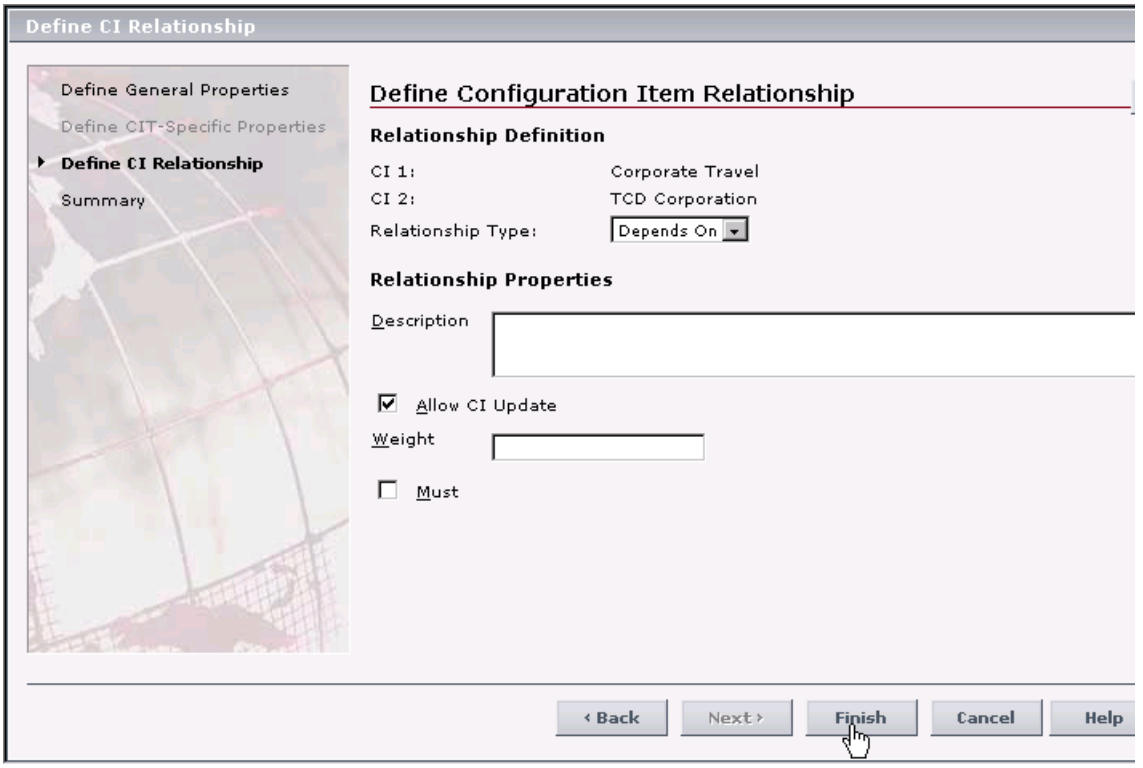
The New CI Wizard opens at the Define General Properties page.

2 Define details for the customer. Select **Customer** from the **CI Type** list. Enter the CI **Name** (for example, TCD Corporation) and select the customer location from the dropdown lists.



Click **Next** to move to the Define CI Relationship page.

3 Check that the details are correct.

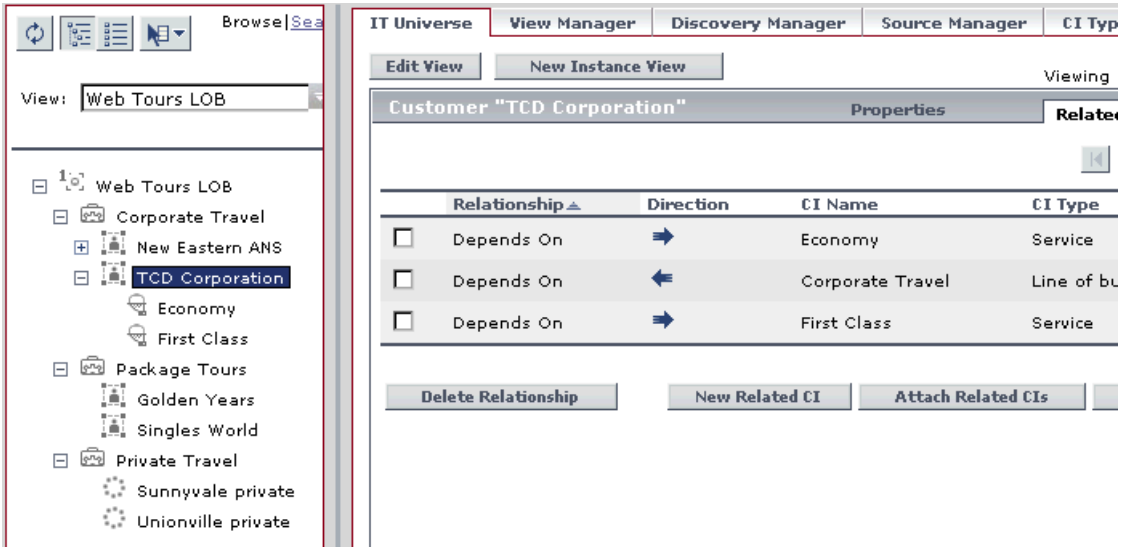


Click **Finish**.

4 The created Customer CI for **TCD Corporation** is shown in View Explorer attached as a child of the parent CI, **Corporate Travel**. The relationship for the attached CI is displayed in the **Related Configuration Items** tab.



Repeat steps 1 through 3 to create additional logical CIs in the view that represent the upper levels of Web Tours LOB hierarchies.



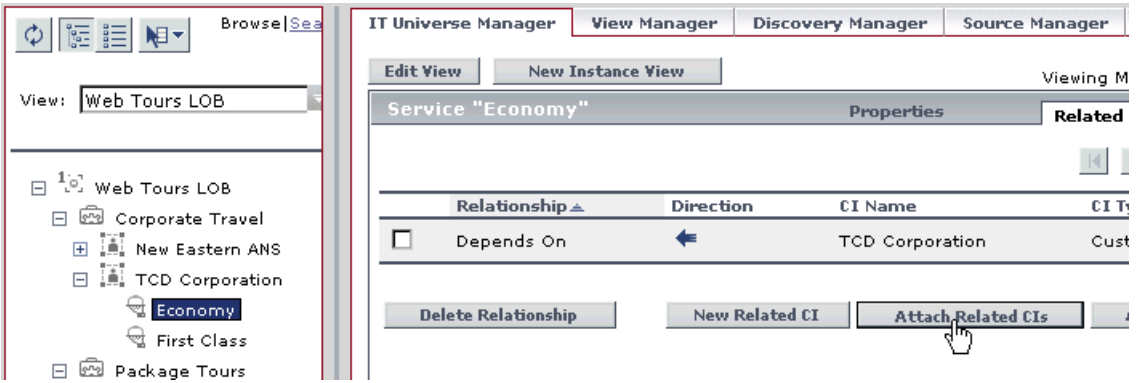
Attach CIs from Discovery Process to CIs in the View

This section describes the attachment of existing CIs in the CMDB to the LOB subtrees. These existing CIs are NT host machines that discovery process discovered in the network and added to the Network Topology view.

The host machines are running the travel service applications that are relevant for the LOB subtrees.

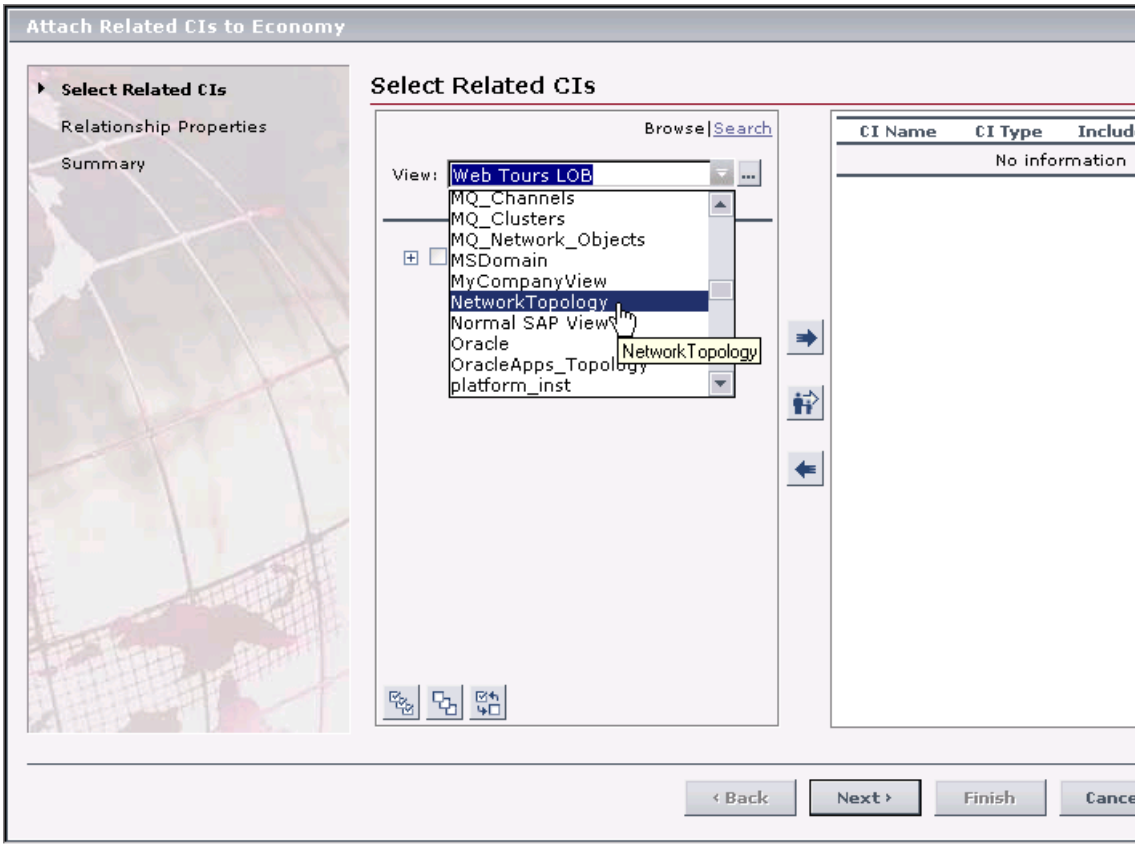
To attach discovery process CIs:

1 Select the CI to attach to in View Explorer (for example, **Economy**), and then either select **Attach Related CIs** from the right-click menu for the CI, or click the **Attach Related CIs** button in the right-hand pane.

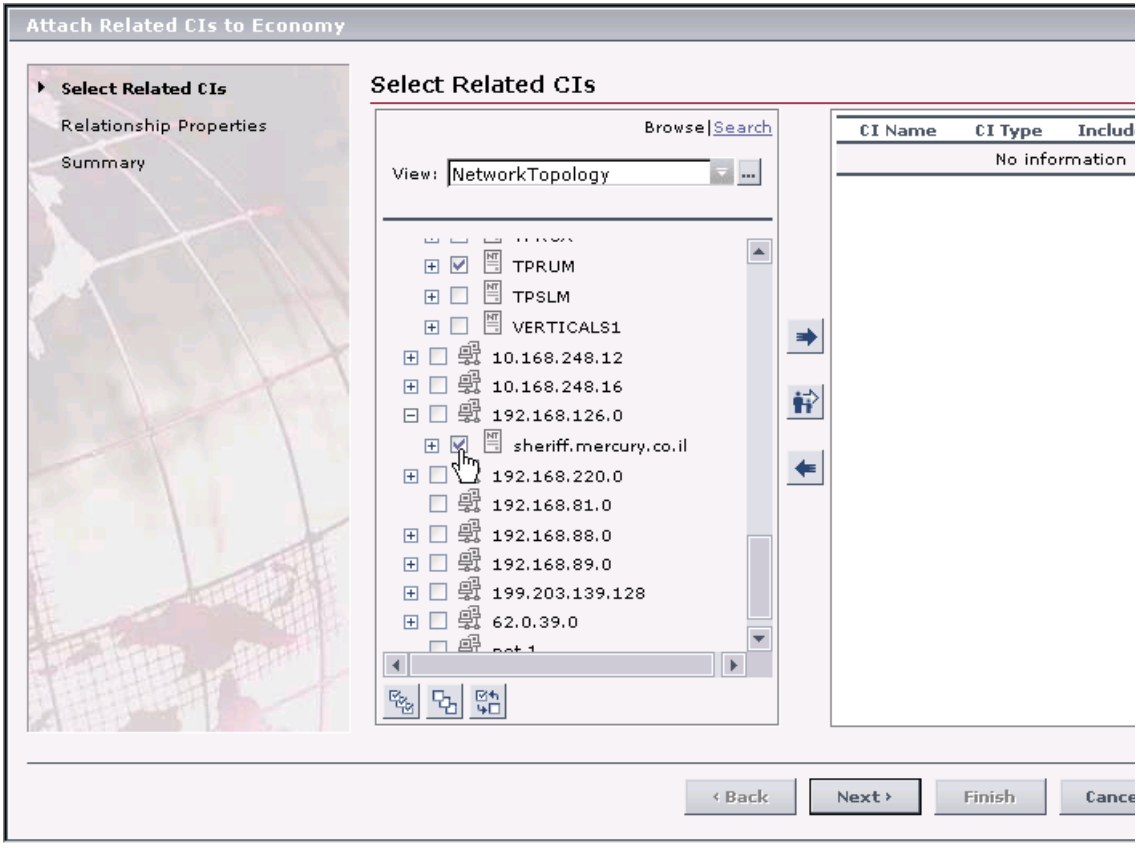


The Attach Related CIs wizard opens at the Select Related CIs page.

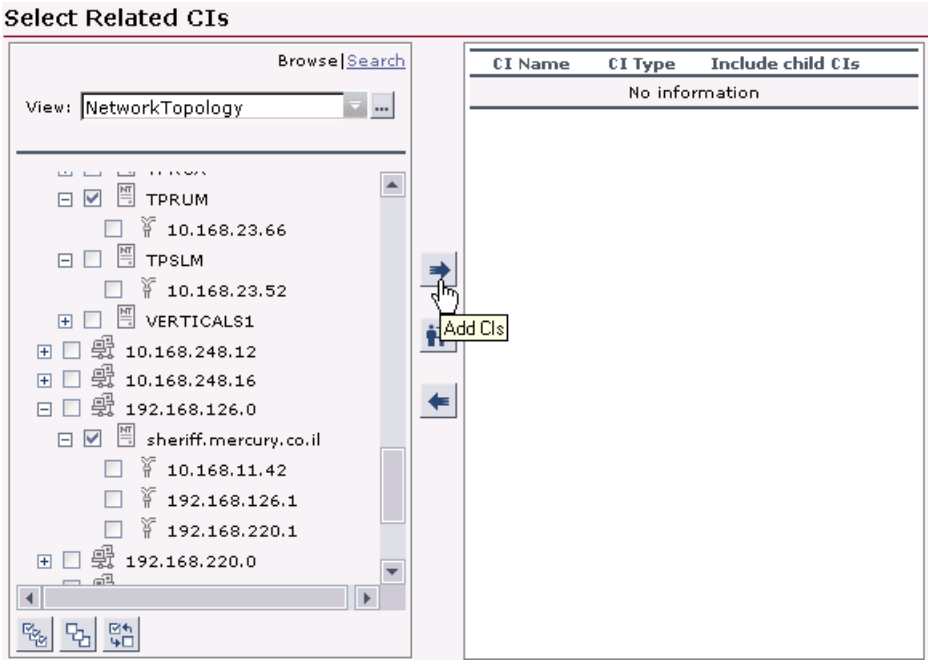
2 The left pane displays the **Web Tours LOB** view. Change to the **Network Topology** view by selecting it from the **View** list.



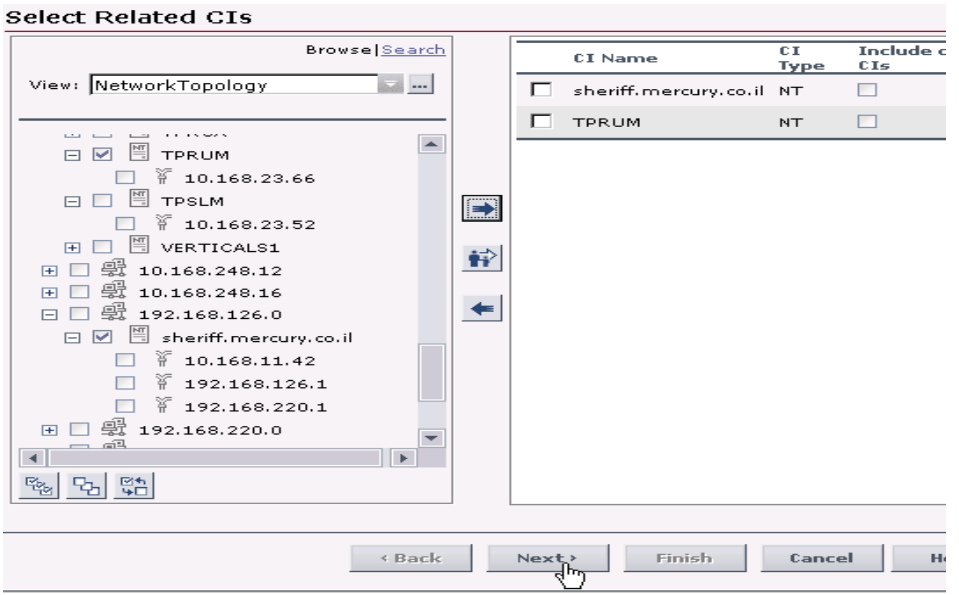
3 In the **Network Topology** view, select the check boxes for the required CIs—CIs representing the host machines that are running the applications relevant for this customer subtree (**TPRUM** and **sheriff**).



4 Click the top arrow button to add the selected CIs (without their child CIs) to the **Web Tours LOB** view.

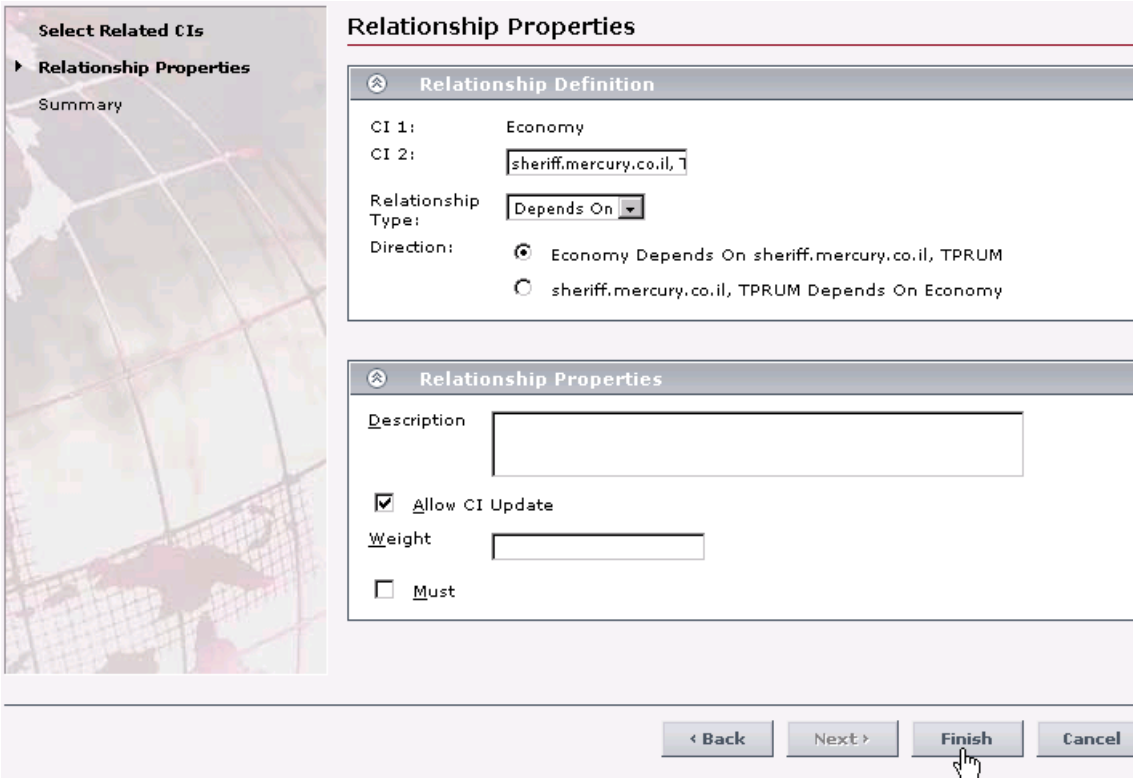


5 The selected CIs are listed in the right-hand pane.



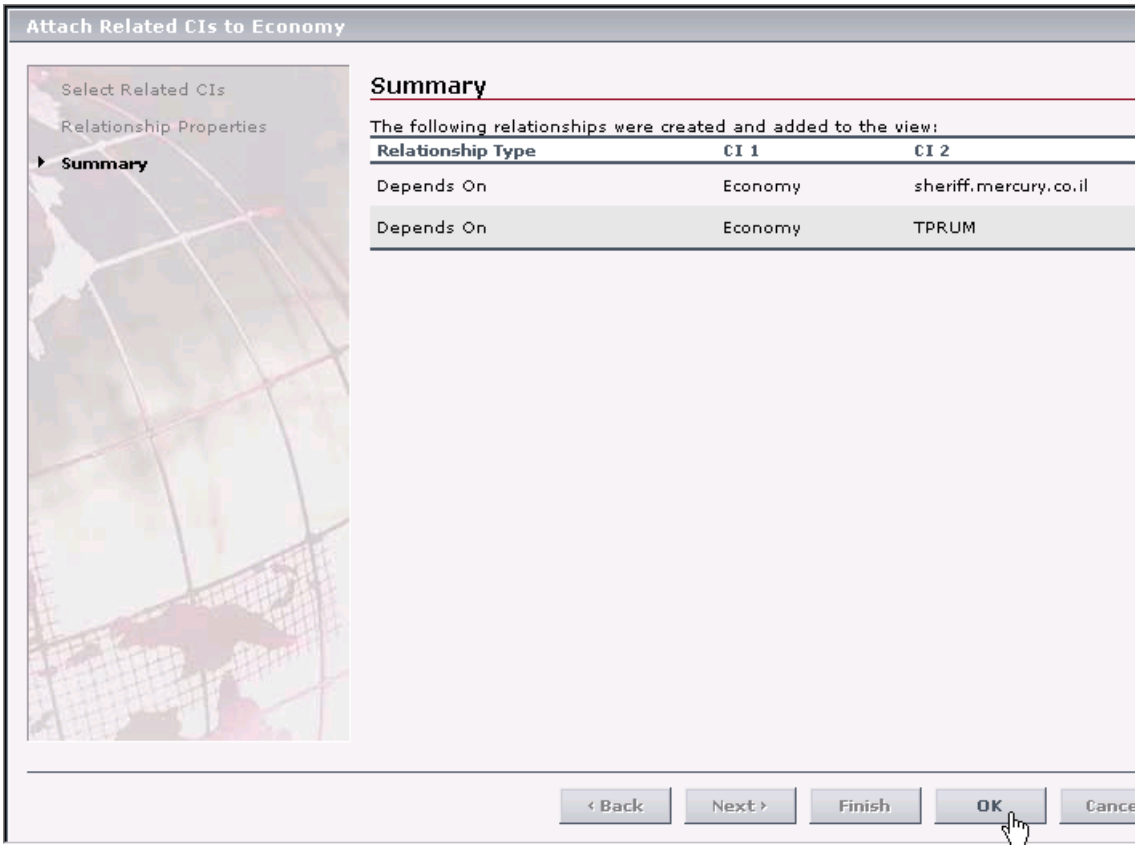
Click **Next** to move to the Relationship Properties page.

6 The Relationship Properties window defines the relationship to be created between the **Economy** CI and the CIs added from the Network Topology view.



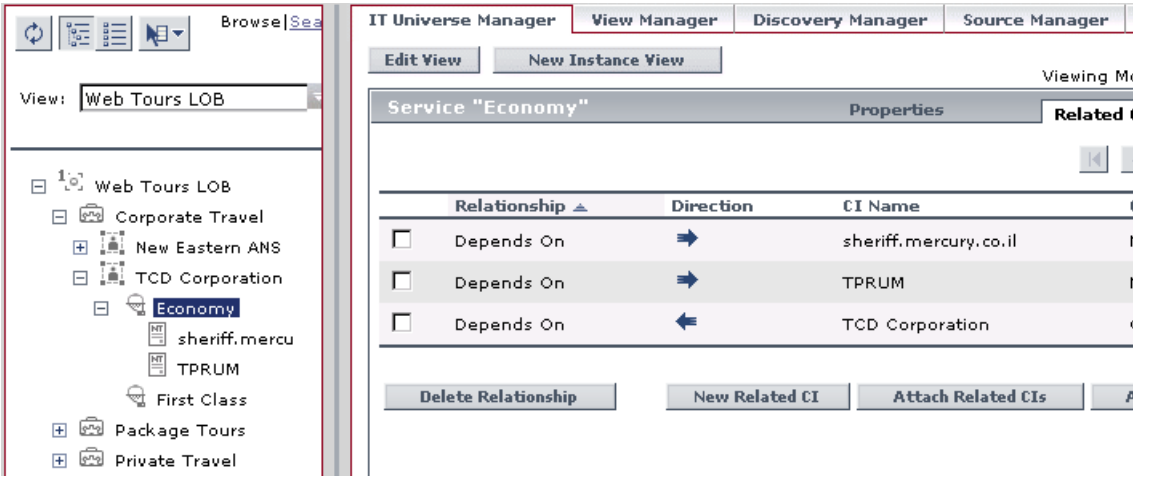
Click **Finish** to move to the Summary page.

7 The Summary page summarizes the CIs added to the view.



Click **OK**.

8 The CIs you added to the view appear in View Explorer attached as children of the parent CI, **Economy**. The relationships for the attached CIs are displayed in the **Related Configuration Items** tab.



Repeat steps 1 through 7 to add additional existing CIs to the view.

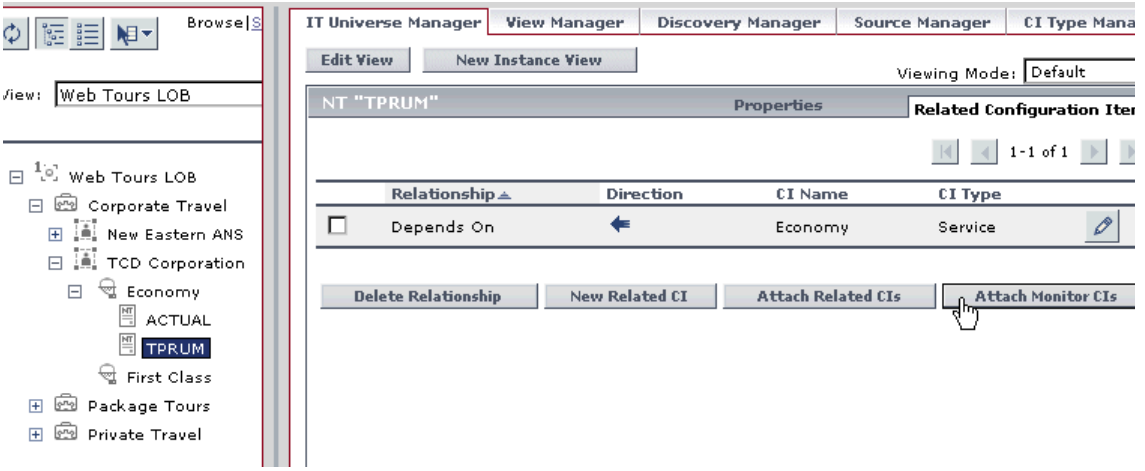
Attach Monitoring CIs to CIs in the View

This section describes the attachment of monitor CIs or monitor subtrees to the host CIs. These monitoring CIs are checking network performance for the host machines, and monitoring system service on the machines.

Note that the individual monitor CIs and the monitor subtrees cannot be attached to a CI in a single operation, because they are added to the parent CI using different relationships (depends on, monitored by).

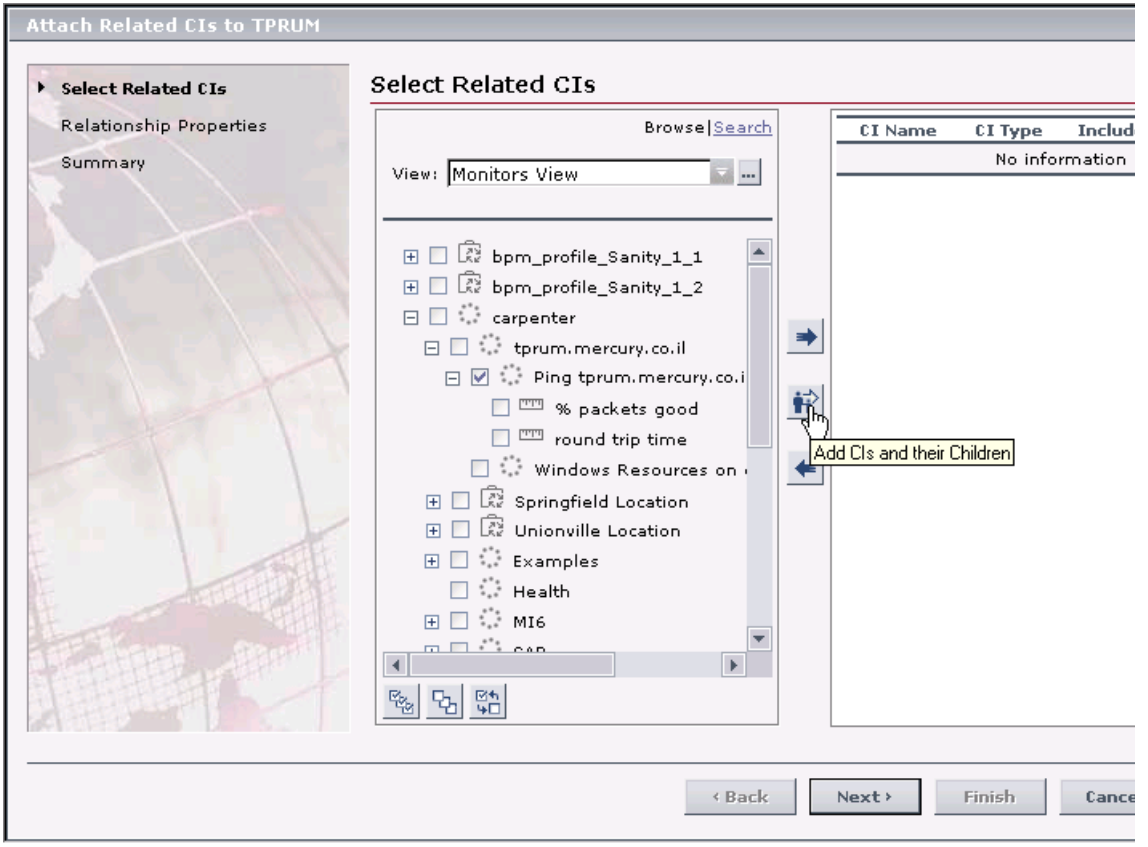
To attach monitoring CIs:

1 Select the CI to attach to in View Explorer (for example, **TPRUM**), and then either select **Attach Monitor CIs** from the right-click menu for the CI, or click the **Attach Monitor CIs** button in the right-hand pane.

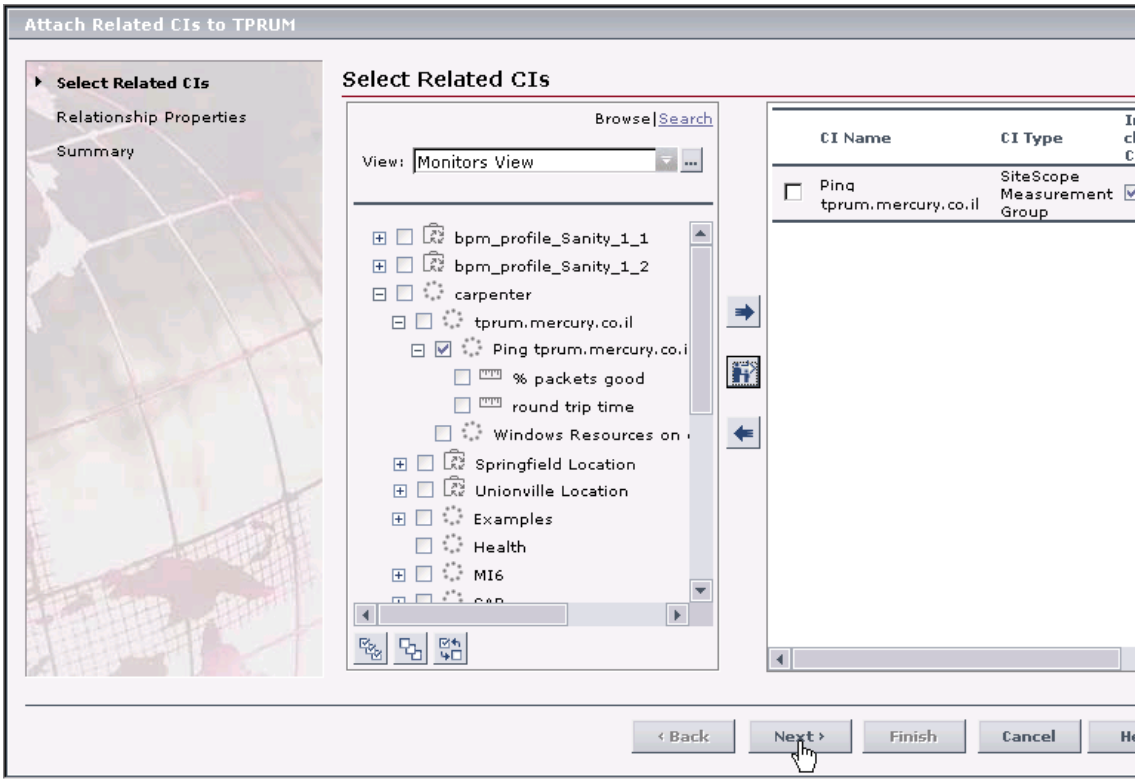


The Attach Related CIs wizard opens at the Select Related CIs page.

2 The Select Related CI page displays the **Monitors View** in the left-hand pane. Select group CIs to attach the required monitoring subtrees to **TPRUM**, for example, the SiteScope Measurement Group CI **Ping tprum.mercury.co.il**, then click the recursive (middle) arrow button (to include the child CIs).

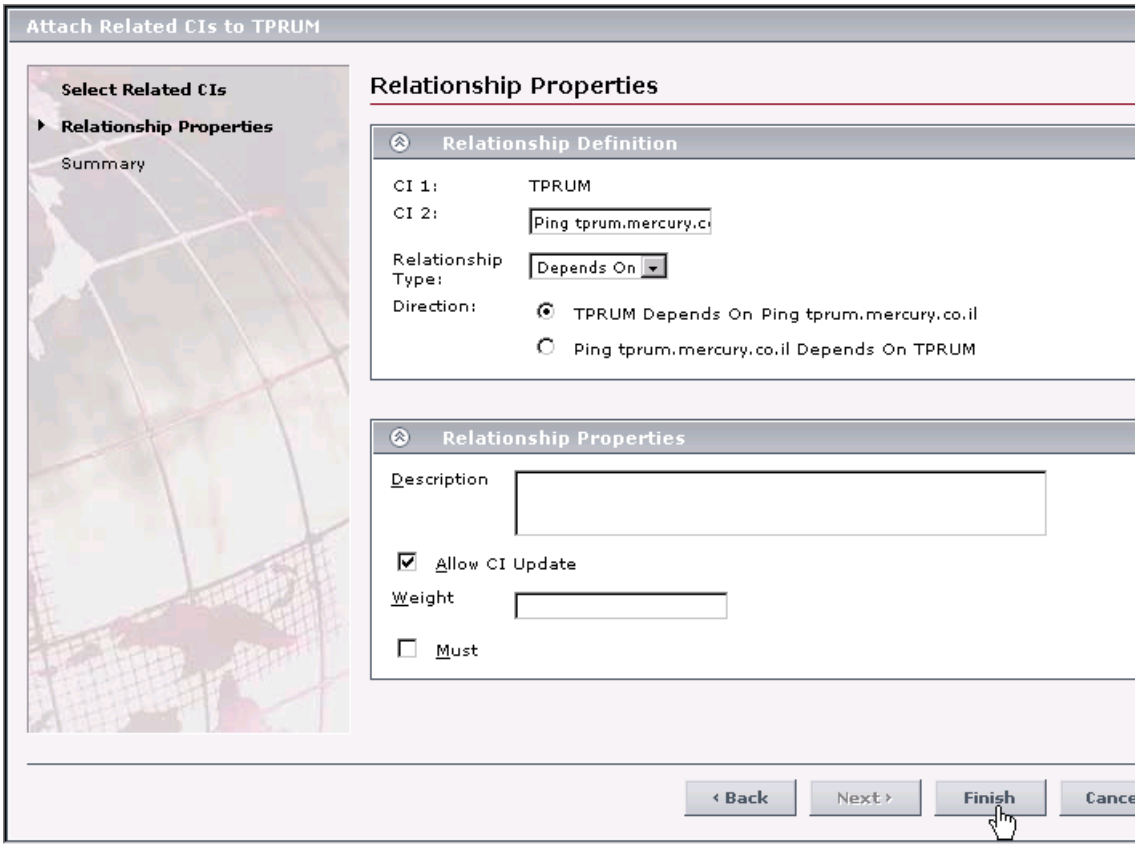


3 The selected CIs are listed in the right-hand pane.



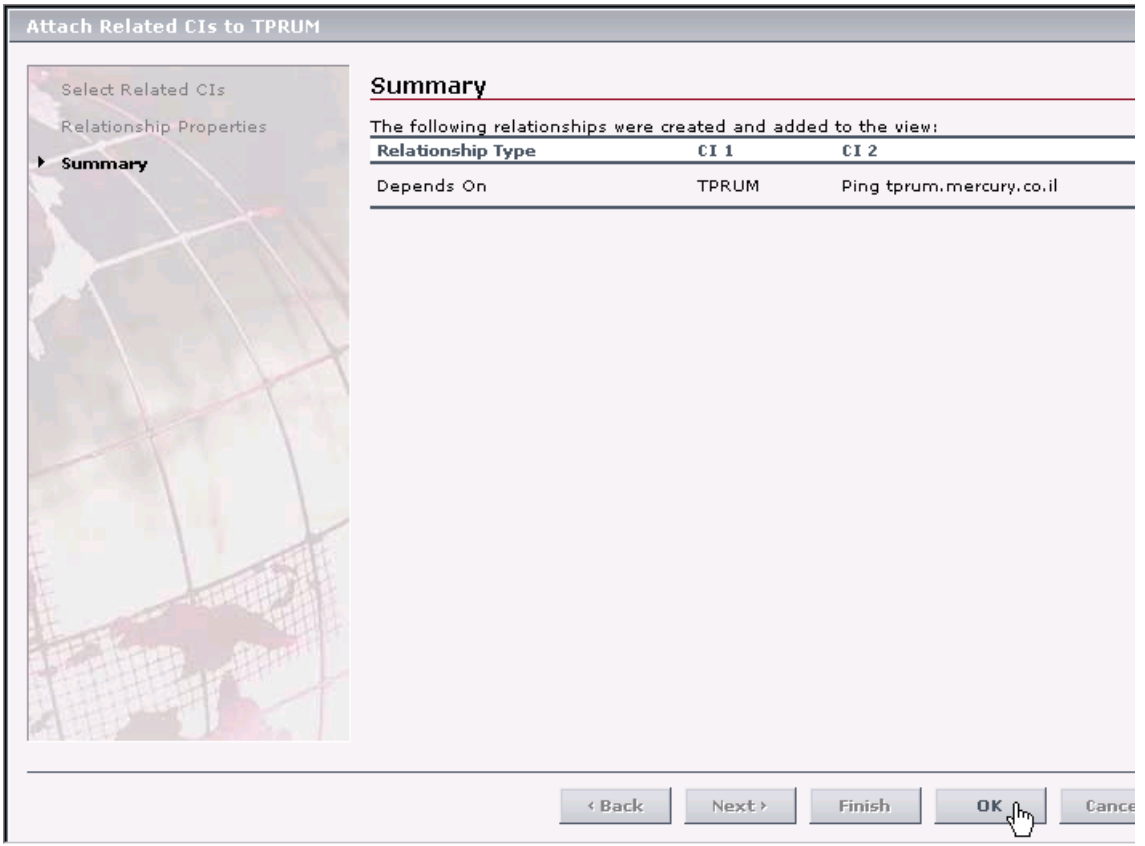
Click **Next** to move to the Relationship Properties page.

4 The Relationship Properties window defines the **Depends On** relationship to be created between the **TPRUM** CI and the **Ping tprum.mercury.co.il** CI.



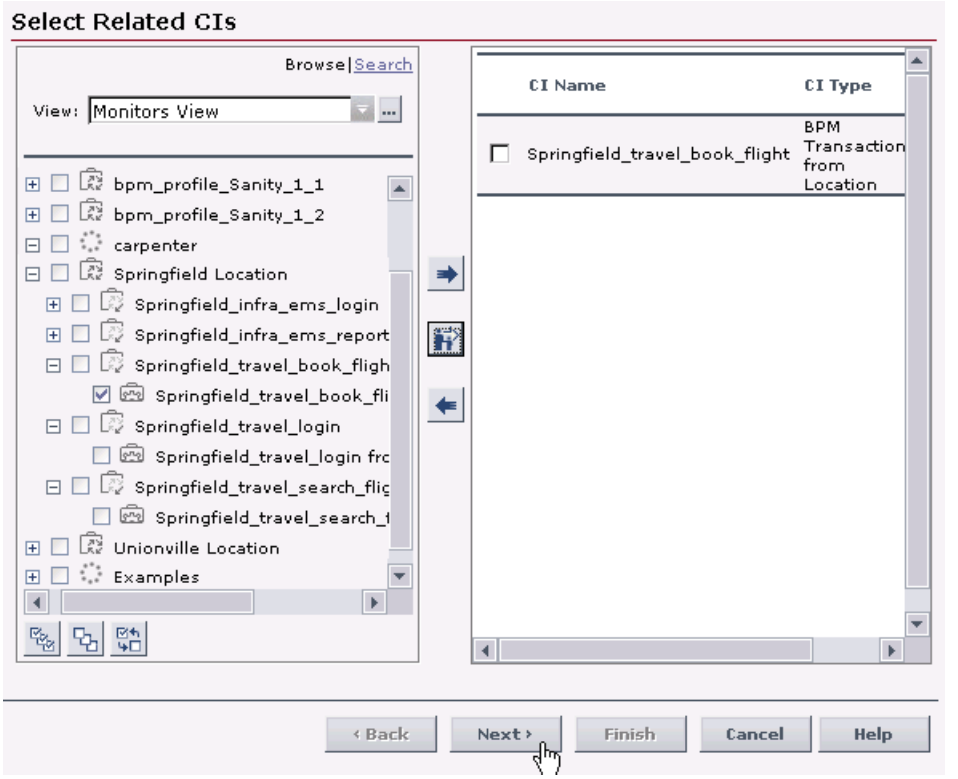
Click **Finish** to move to the Summary page.

5 The Summary page summarizes the CIs added to the view.



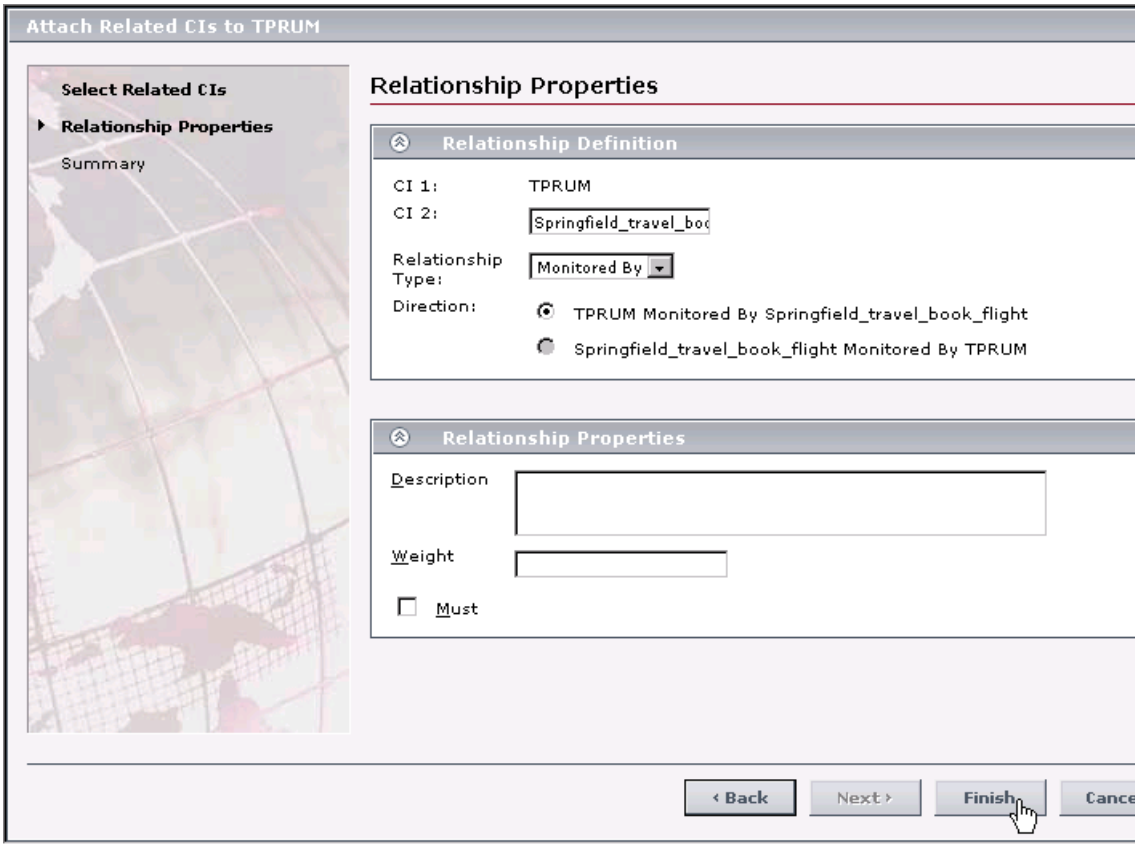
Click **OK**.

6 Repeat steps 1 and 2 to attach individual monitor CIs to **TPRUM**, for example, the BPM Transaction CI **Springfield_travel_book_flight**. These CIs are added as leaf CIs for the subtree.



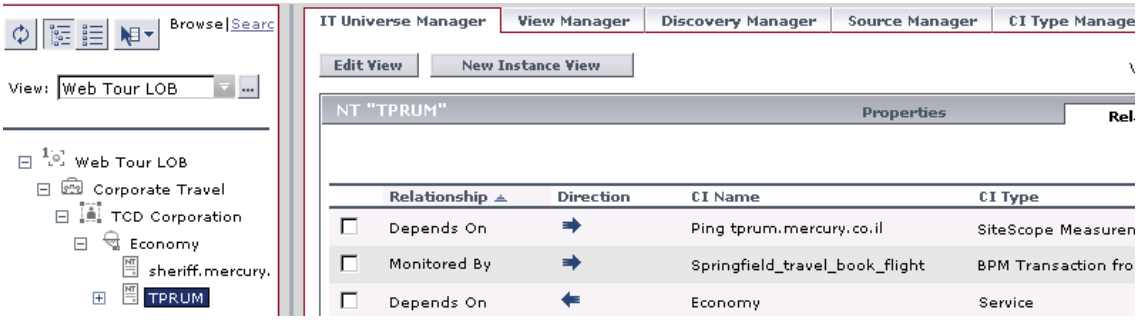
Click **Next** to move to the Relationship Properties page.

7 The Relationship Properties window defines the **Monitored By** relationship to be created between the **TPSLM** CI and the monitor CI.



Click **Finish** to move to the Summary page, then click **OK**.

8 The monitoring CI you added to the view appear in View Explorer as a child of the parent CI, **TPRUM**. The relationships for the attached CIs are displayed in the **Related Configuration Items** tab.



Repeat steps 1 through 7 to add additional existing CIs to the view.

Stage 3: Customize KPI Settings

The third stage of the scenario involves customizing the KPI setup for the view, to add an additional KPI category to the CIs, and to change the default rule calculation for some KPIs.

The KPIs for the monitor CIs automatically propagate up through the hierarchy as soon as the monitor CIs are attached to the view.

Customizing KPI settings is divided up into two sections:

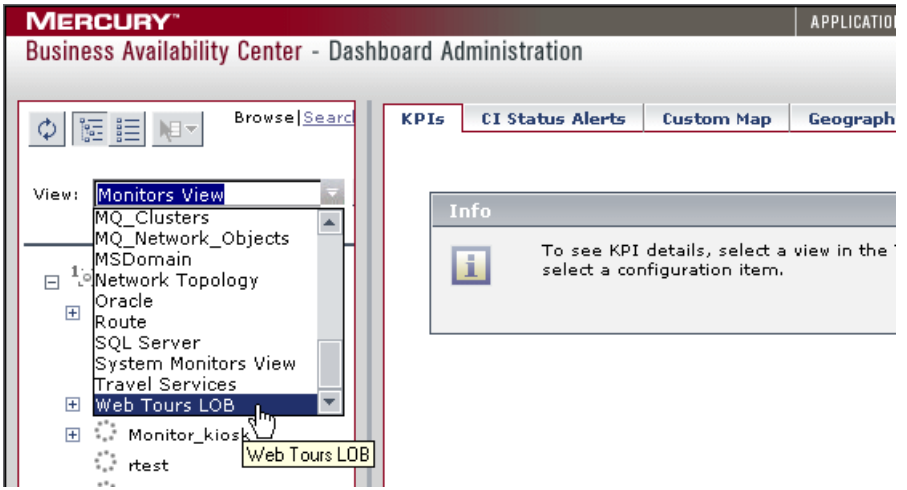
- Define Over Time Impact KPI for CIs, described below
- Define Weighted Percentage Rule, described on page 165

Define Over Time Impact KPI for CIs

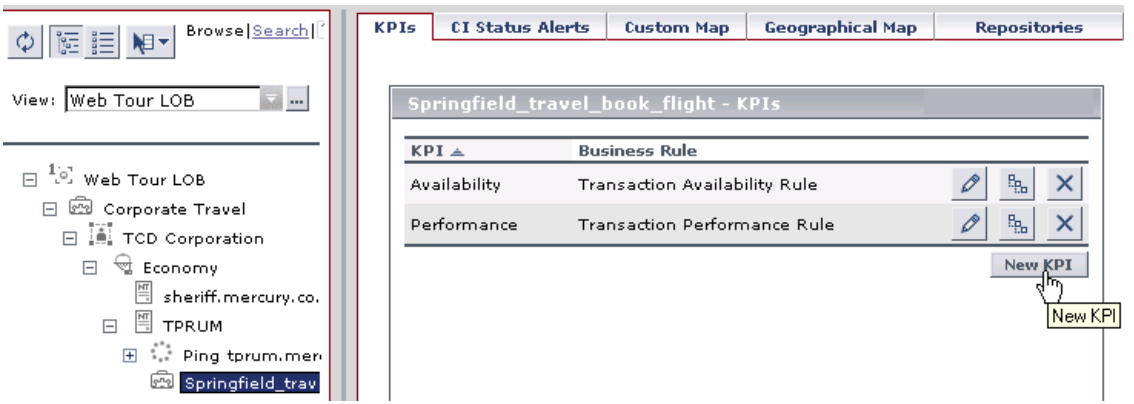
For this scenario, A.A. Web Tours want the Web Tours LOB view to show over time impact metrics, based on availability of their network resources. The OT Impact KPI is defined for the BPM Transaction monitors (which by default have the Availability KPI attached). The OT Impact KPI settings are customized to represent the cost to the company when each resource is unavailable.

To define the OT Impact KPI:

1 In the **Admin** > **Dashboard** > **KPIs** tab, open the Web Tours LOB view in View Explorer (left pane).

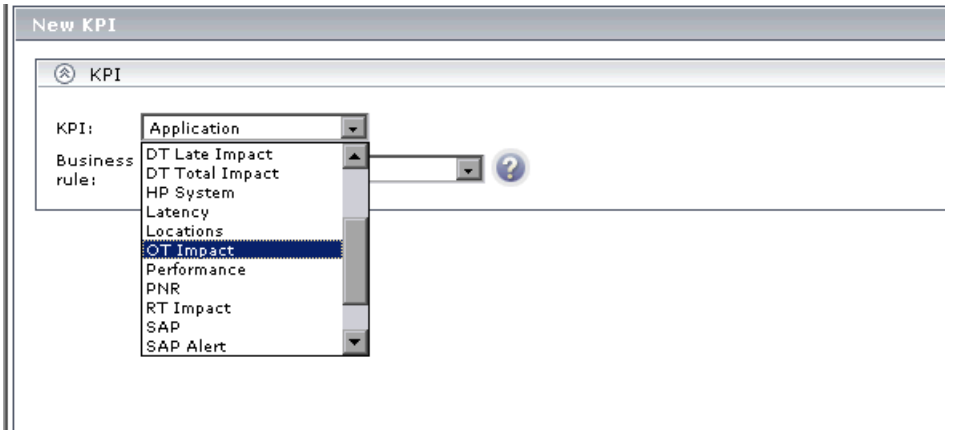


2 Select a monitor CI in the view, for example, **Springfield_travel_book_flight**, then click **New KPI** in the right-hand pane.



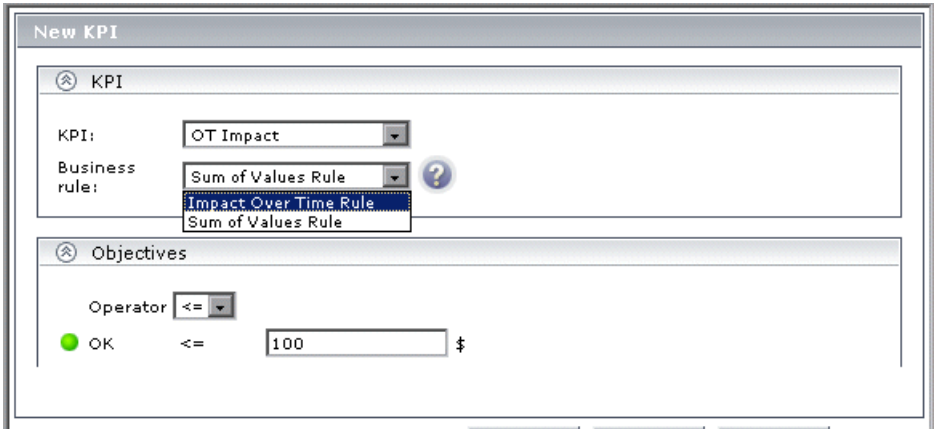
The New KPI window opens.

3 Select **OT Impact** from the **KPI** list.



The window is updated to show business rules and objectives for the OT Impact KPI.

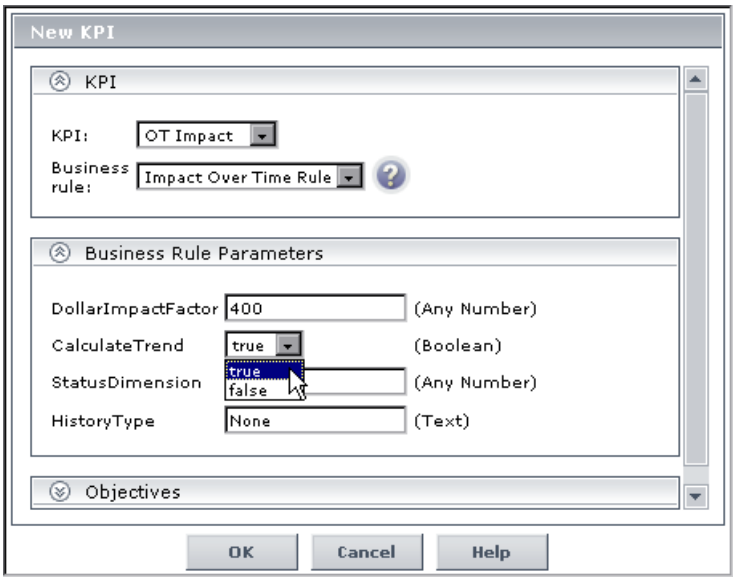
4 Select the monitor rule for OT Impact, **Impact Over Time Rule**, from the **Business rule** list.



The window is updated to include parameters for the rule.

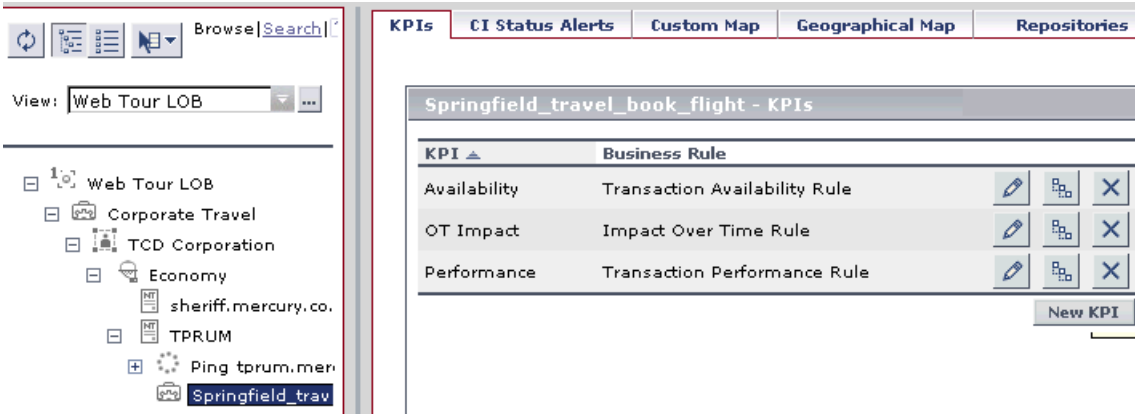
5 In the **Business rule parameters** area, modify the parameters as required:

- In the **DollarImpactFactor** box, enter the dollar financial loss per hour for non-availability of this CI - **100**.
- In the **CalculateTrend** box, change the value from **false** to **true**, so that trend status will be displayed for the KPI.

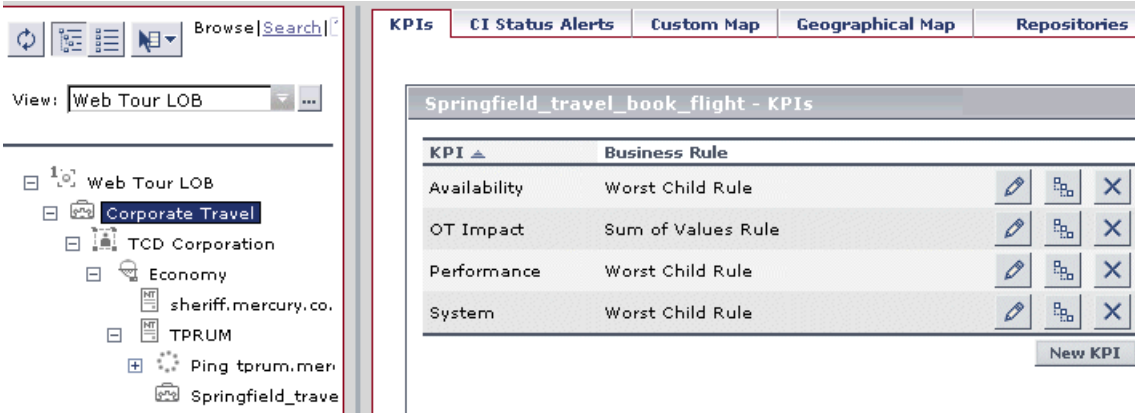


Click **OK** to save the changes.

6 The OT Impact KPI is added to the list of KPIs for the CI.



The KPI also automatically propagates up the hierarchy to the top level. For all CIs above the monitor CI, the OT Impact KPI uses the **Sum of Values Rule** by default.

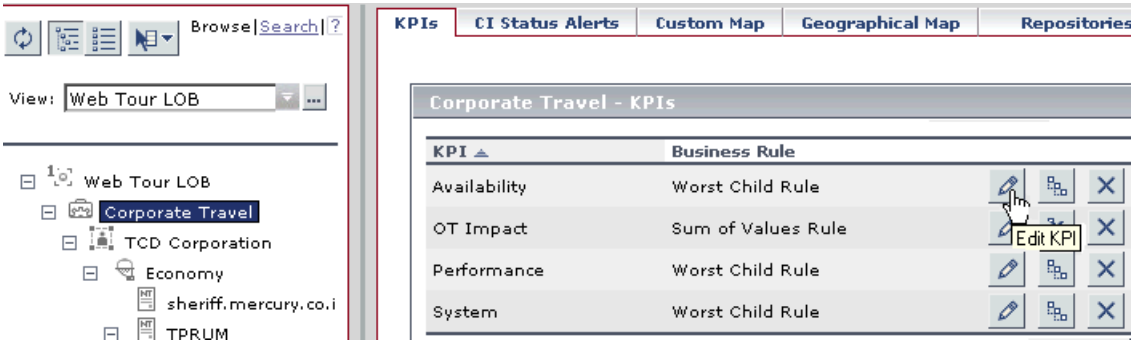


Define Weighted Percentage Rule

For this scenario, A.A. Web Tours considers TCD Corporation to be a key customer, and they want TCD Corporation to have greater impact on the overall status calculations for the Corporate Travel line of business. To achieve this, the Percentage Rule is defined for Corporate Travel KPIs, and the metrics for TCD Corporation are given a heavier weight than the metrics of the other customers attached to Corporate Travel in the view.

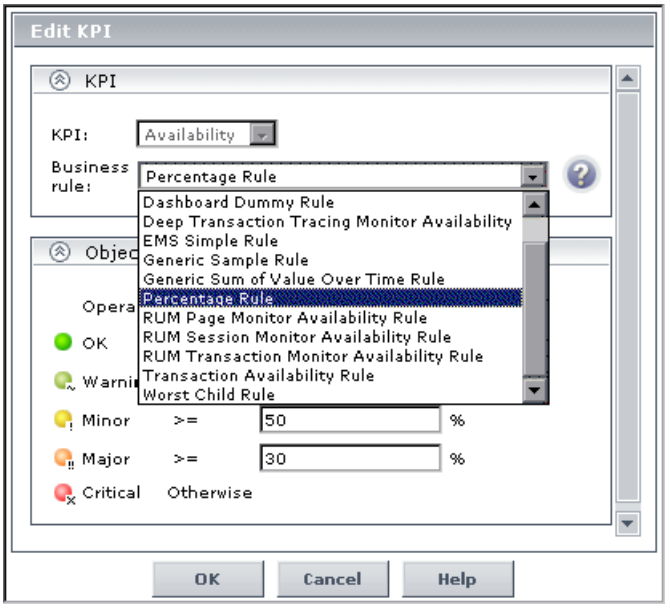
To define weighted percentage rule:

1 In the **Admin** > **Dashboard** > **KPIs** tab, select **Corporate Travel** in View Explorer and then click the **Edit** button for the **Availability** KPI.



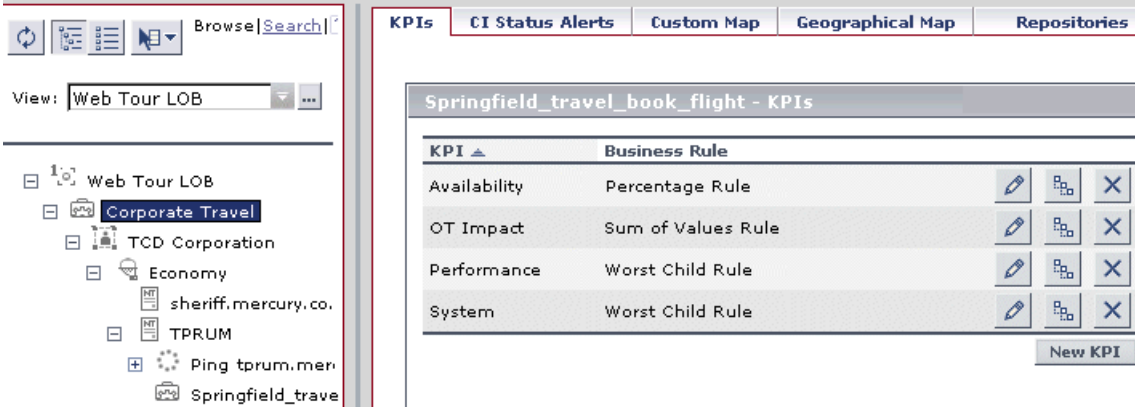
The Edit KPI window opens.

2 Select **Percentage Rule** from the **Business rule** list.



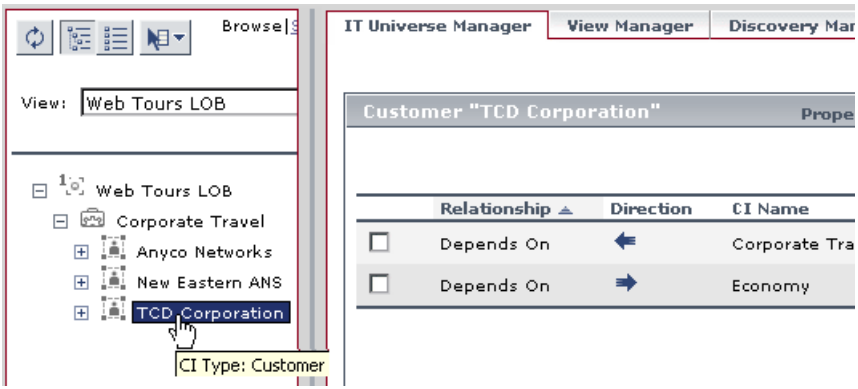
Click **OK**.

3 The new business rule is displayed for the **Availability** KPI in the KPIs list.

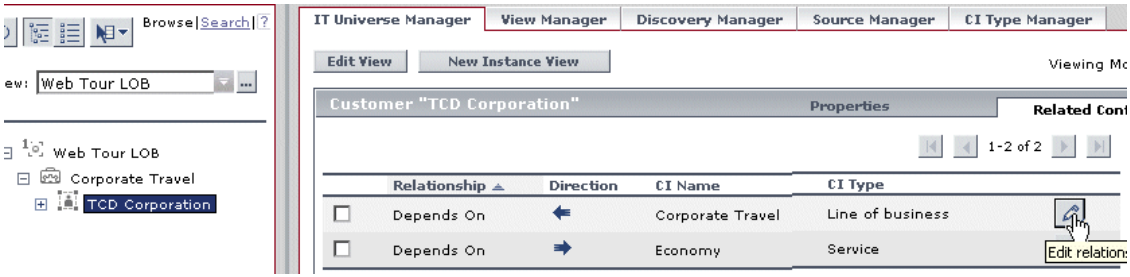


Repeat the procedure for each KPI for **Corporate Travel** that is to use weighted percentage.

4 Move to the **Admin** > **CMDB** > **IT Universe Manager** tab and open the **Web Tours LOB** view in View Explorer. Select the **TCD Corporation** CI in the view.

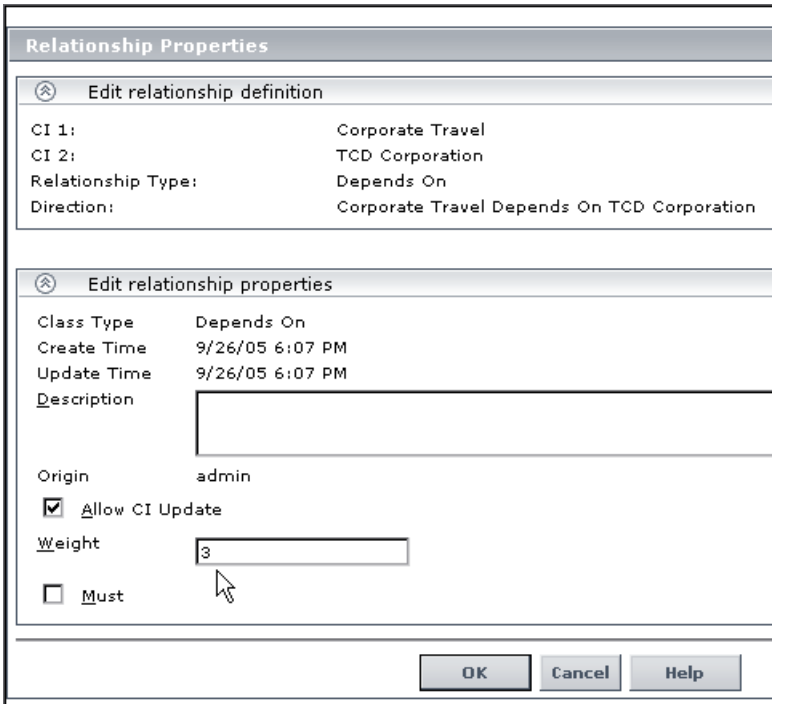


5 In the **Related Configuration Items** tab in the right pane, locate the row in the table for TCD Corporation's relationship with **Corporate Travel**. Click the **Edit relationship** button for the relationship.



The Relationship Properties window opens.

6 Enter the required weight (for example, **3**) in the **Weight** box.



Click **OK**.

Stage 4: Define Viewing Permissions

The fourth stage of the scenario is assigning the completed view to Dashboard, and defining viewing permissions for the view.

Defining viewing permissions is divided up into two sections:

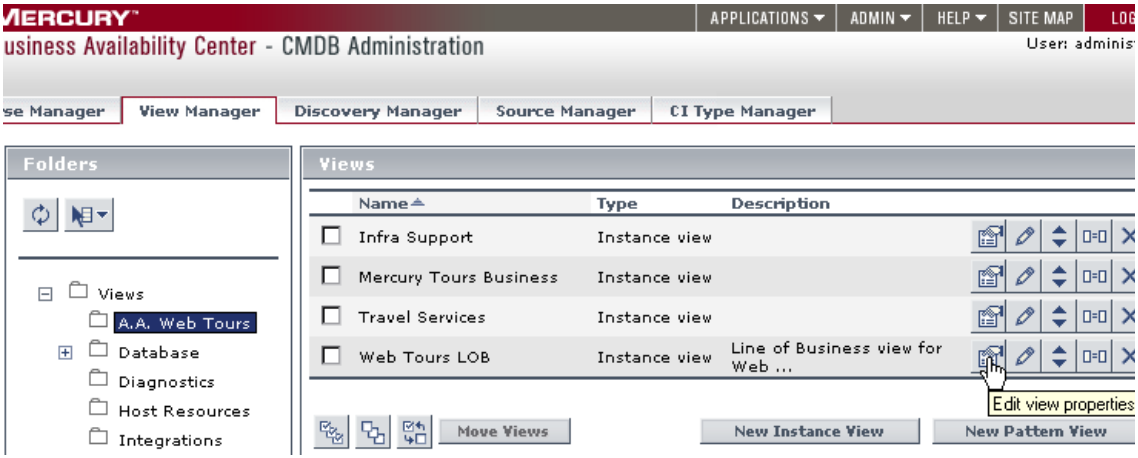
- Assign View to Applications, described below
- Define User Permissions, described on page 170

Assign View to Applications

Now that the view is ready for viewing in Dashboard and Service Level Management, the setting is changed in View Manager so that it is assigned to those applications.

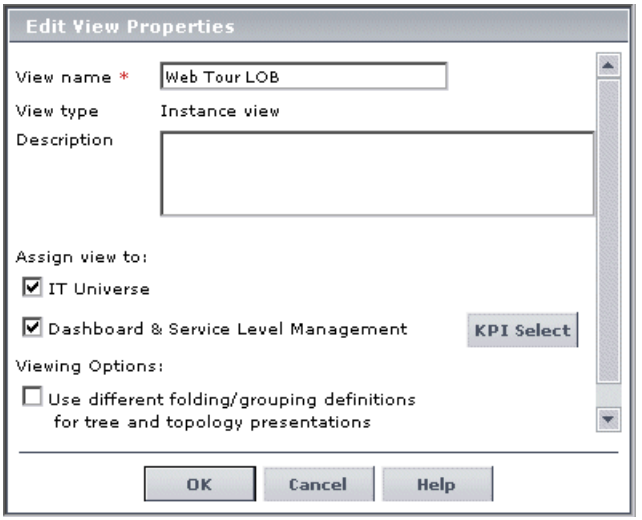
To assign view to Dashboard and Service Level Management:

1 In the **Admin** > **CMDB** > **View Manager** tab, select the **A.A. Web Tours** folder in the **Folders** pane to display the list of views in the folder. Click the **Edit view properties** button for the **Web Tours LOB** view.



The Edit View Properties opens.

2 Select the check box for **Dashboard & SLM**.



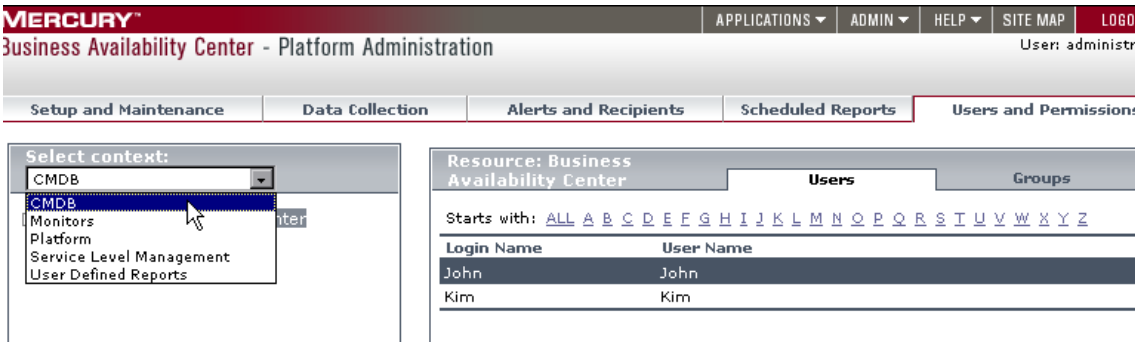
Click **OK**.

Define User Permissions

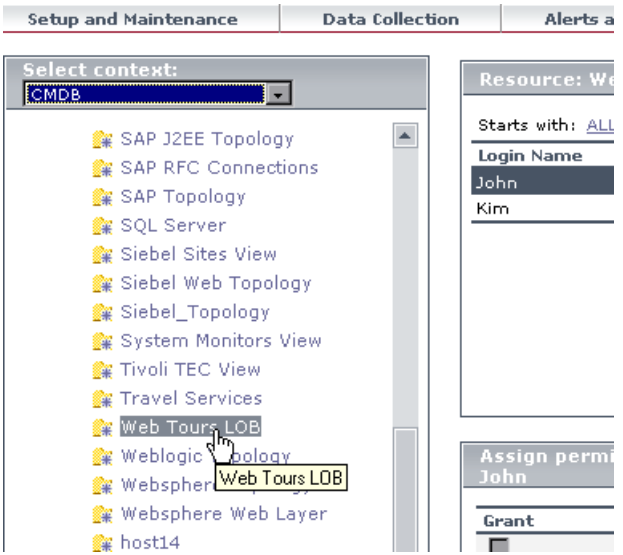
For this scenario, the Web Tours LOB view is set to be available to user Kim for viewing purposes only. This means that Kim can see the view in all applications, but cannot change or delete the view.

To define user permissions for the view:

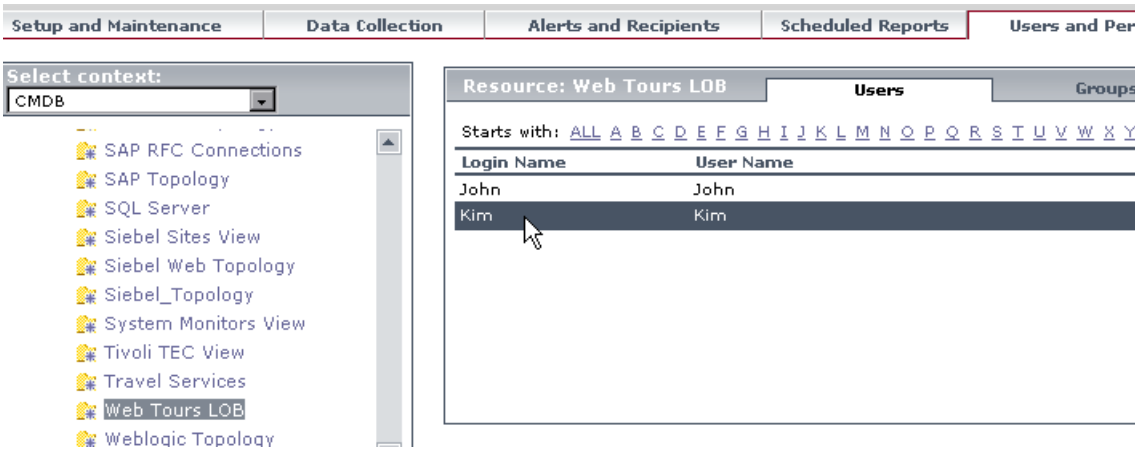
1 In the **Admin** > **Platform** > **Users and Permissions** > **Permissions Management** tab, select **CMDB** from the **Select context** list (left pane).



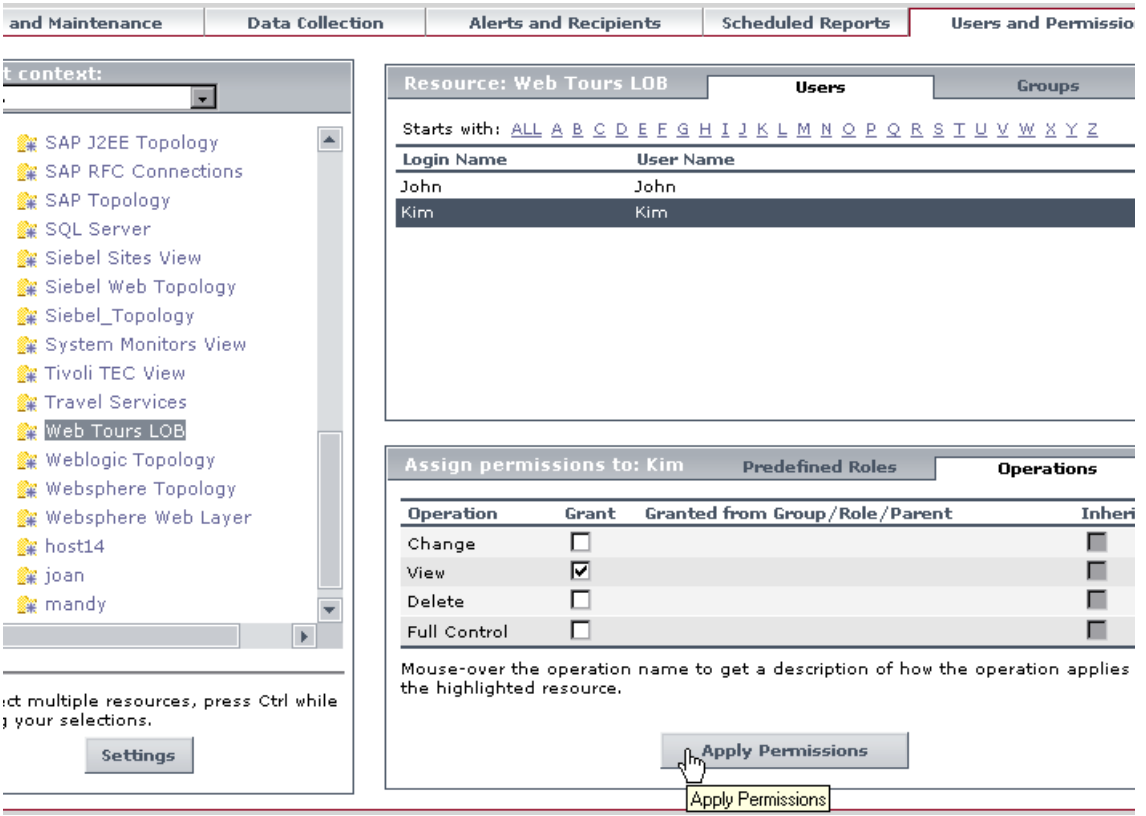
2 Expand the **Views** folder in the **Select context** pane, and select **Web Tours LOB**.



3 In the **Resource** pane (top right), select the user **Kim** from the list of users.



4 In the **Assign permissions to** pane (bottom right), select the **Operations** tab, and then select the **View** check box to assign user Kim permission for viewing only for the Web Tours LOB view.



Click **Apply Permissions** to save the permissions settings.

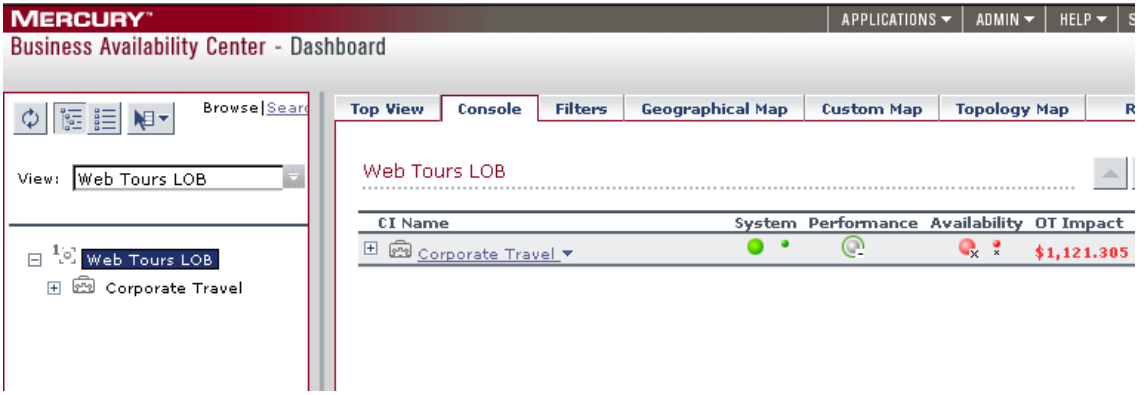
Stage 5: See KPI Status in Dashboard

The final stage of the scenario is using the view in Dashboard to monitor KPI status for A.A. Web Tours' main lines of business.

To view KPI status for the view in Dashboard:

1 Access the **Applications** > **Dashboard** > **Console** tab.

2 Select **Web Tours LOB** from the **View** list in View Explorer.



Index

C

CIs
- add manually to view 27
- attach monitor CIs 49
- attach related CIs 49
- CIT-specific properties 72
- context menus for 68
- creating new in view 40
- creating non-related CIs 40
- creating related CIs 41
- creating with New CI Wizard 42
- deleting 56
- Dynamic Node Factory CI 109
- editing properties 65
- icon composition 78
- include all child CIs 70
- include related elements option 70
- multiple CI selection 79
- properties 65
- property descriptions 66
- relationship properties 49
- selecting multiple 79

CITs
- in IT Universe 13
- properties for CIs 72

CMDB
- creating new CIs 40
- deleting CIs 56
- deleting relationships 62
- populating 2
- relationships for CIs 57

composite selectors
- DNode definition key 129

context menus
- for CIs 68

D

DNode definition key 120
- CI templates 121
- components 121
- defining composite selectors 129
- generators 121
- overview 120
- relationship templates 121
- using 111

Dynamic Node Factory
- definition 110
- properties 113
- using 111
- when to use 112
- without grouping key in sample 118
- working with 109

dynamically created hierarchies 111

G

grouping key
- missing in sample 118

H

hierarchy
- dynamically created 111

I

include child CIs 70
- invisible children 72

include related elements option 70

instance views
- building 18
- creating 20
- example scenario 133

in IT Universe 7
invisible children 72
IT Universe
building an instance view 18
CI properties 65
CITs 13
creating instance views 20
getting started 5
instance views 7
introduction 1
New CI Wizard 42
pattern views 7
populating IT universe model 2
related configuration items tab 24
relationships 57
service views 12
tab, working with 5
toolbar options 81
using 1
working with CIs 23

K

KPIs
scenario for customizing 160

L

layer
containing large number of CIs 77
layer setup
option 86
layer's layout
defining 86
layout
setting layer's layout 86

M

map levels
navigating between 78
monitor CIs, in service view 17

N

New CI Wizard 42

P

pattern views
in IT Universe 7
permissions
scenario for defining for a view 168
properties
CIT-specific 72
for configuration items 65
for Dynamic Node Factory 113
for relationships 60

R

relationships
about 57
defining properties 49
deleting 62
editing 59
properties for 60

S

samples
missing grouping key 118
scenario
building an instance view 136
creating a view 134
customize KPI settings 160
defining view permissions 168
instance view example 133
service views
creating CIs 40
creating in IT Universe 20
creating new CIs 40
creating related CIs 41
defining Dynamic Node Factory CI 109
deleting CIs 56
example scenario for building 133
in IT Universe 12
monitor CIs 17
relationships in 57
tree structure 12
setup
layer layout 86

T

template
    Dynamic Node Factory 120
tooltip
    in IT Universe topology map 79
topology map
    concepts 78
    displaying mode 77
    in IT Universe 76
    simultaneous selection 79

V

viewing mode
    topology map 77