Cisco Network Planning Solution – SPM Design and Analysis Planning and Design Module User Guide

Software Release 11.5

Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

Text Part Number: OL-8610-01

Cisco Network Planning Solution – SPM
Design and Analysis
Design Module User Guide

Copyright

Document Copyright

Document Title: Planning and Design User Guide
Document Part Number: D00267
Version: 6

Software Copyright

Product Name: SP Guru
Product Release: 11.5

Documentation Conventions

OPNET documentation uses specific formatting and typographic conventions to present the following types of information:

- Objects, examples, and system I/O
- Object hierarchies, notes, and warnings
- Computer commands
- Lists and procedures

Objects, Examples, and System I/O

- Directory paths and file names are in plain Courier typeface:

  ```
  opnet\release\models\std\ip
  ```

- Function names in body text are in italics:

  op_dist_outcome()

- The names of functions of interest in example code are in bolded Courier typeface:

  ```
  /* determine the object ID of packet’s creation module */
  src_mod_objid = op_pk_creation_mod_get (pkptr);
  ```

- Variables are enclosed in angle brackets (< >):

  ```
  <opnet_user_home>/op_admin/err_log
  ```

Object Hierarchies, Notes, and Warnings

Menu hierarchies are indicated by right angle brackets (>); for example:

Open File > Print Setup > Properties...

Attribute hierarchies are represented by angled arrows (➘) that indicate that you must drill down to a lower level of the hierarchy:

Attribute level 1 ➘ Attribute level 2 ➘ Attribute level 3

Note—Notes are indicated by text with the word Note at the beginning of the paragraph. Notes advise you of important supplementary information.

WARNING—Warnings are indicated by text with the word WARNING at the beginning of the paragraph. Warnings advise you of vital information about an operation or system behavior.

Computer Commands

These conventions apply to Windows systems and navigation methods that use the standard graphical-user-interface (GUI) terminology such as click, drag, and dialog box.

- Key combinations appear in the form “press **`<button>+x`**”; this means press the **`<button>`** and **`x`** keys *at the same time* to do the operation.
- The mouse operations *left-click* (or *click*) and *right-click* indicate that you should press the left mouse button or right mouse button, respectively.

Lists and Procedures

Information is often itemized in bulleted (unordered) or numbered (ordered) lists:

- In bulleted lists, the sequence of items is not important.
- In numbered lists, the sequence of items is important.

Procedures are contained within procedure headings and footings that indicate the start and end of the procedure. Each step of a procedure is numbered to indicate the sequence in which you should do the steps. A step may be followed by a description of the results of that step; such descriptions are preceded by an arrow.

Procedure FM-1 Sample Procedure Format

1 Procedure step.

➥ Result of the procedure step.

2 Procedure step.

End of Procedure FM-1

For more information about using and maintaining OPNET documentation, see the Documentation Guide.

Document Revision History

Release Date	Product Version	Chapter	Description of Change
August 2005	11.5	OPNET Design Actions	• Updated Design Action Log section for new log viewer.
		MPLS Explicit Route Configlet Generation	• Updated Generated Configlet Files on page DA-10-2 to include support for exporting bandwidth-request and LSP-destination data
		MPLS Traffic Engineering	• Updated Design Action Log section for new log viewer.
		IP QoS Design Actions	• New chapter with descriptions of the ip_qos_configuration and ip_qos_queue_sizing design actions
		Project Pricing Reporter	• New chapter
		MPLS Differentiated-Service-Aware Traffic Engineering	• New chapter
November 2004	11.0 PL1	MPLS TE	• Added Delay Constraints, Hop Limits, 3 new reports (Primary Route, Secondary Route, and Contained LSPs) and related changes.
		MPLS Explicit Route Configlet Generation	• New chapter
		MPLS Minimum Fast Reroute Design	• New chapter
		MPLS Minimum Cost Topology Design	• New chapter
January 2004	10.5	Link Dimensioning	• New chapter
		Link Dimensioning Resilient	• New chapter
		Ring Backbone	• New chapter
		Spanning Tree	• New chapter
		Dual Tree	• New chapter
		MPLS TE	• Moved from Protocols menu
		Design Actions	• Moved chapter from Guru *User Guide*

Contents

List of Figures

List of Tables

List of Procedures

1 OPNET Design Actions

A *design action* is a high-level operation that automates the process of changing a network model, usually to achieve a design goal. With a design action, you can modify your network models with one operation.

The OPNET Model Library includes a set of standard design action models. Each design action combines internal design logic with a set of configurable attributes. You can configure a design action in the same way that you configure other types of objects. If you own a license to the OPNET Development Kit, you can also extend the standard design action models to embed your custom logic or create completely new custom design actions.

Workflows for Design Actions

Design actions provide a very flexible way to make automated changes to your network. The workflow for a design action varies in complexity, depending on the extent of customization required.

Table 1-1 Workflows for Design Actions (Part 1 of 2)

Extent of Customization	Workflow / References
No customization	• Run action (Design > Run Design Action) • View results (Viewing Results on page DA-1-10)
Configure action	• Select action (Design > Configure/Run Design Action) • Configure attributes (Configuring a Design Action on page DA-1-3) • Run action (Running a Design Action on page DA-1-8) • View results (Viewing Results on page DA-1-10)

Table 1-1 Workflows for Design Actions (Part 2 of 2)

Extent of Customization	Workflow / References
Select subactions / program blocks	• Select action (Design > Configure/Run Design Action) • Select subactions to fill specific functional roles (Compound Actions, Roles, and Subactions on page DA-1-4) • Configure attributes (Configuring a Design Action on page DA-1-3) • Run action (Running a Design Action on page DA-1-8) • View results (Viewing Results on page DA-1-10)
Modify / customize existing action	• Open existing action in Design Action Editor (Custom Design Actions on page DA-1-11) • Add new control attributes • Change embedded C/C++ code • Rebuild action
Create custom action	• Create entirely new action in Design Action Editor (Custom Design Actions on page DA-1-11) • Specify all information, including attributes and C/C++ code
End of Table 1-1	

Design Menu

The Design menu includes operations for selecting, configuring, running, and the viewing the results of design actions. Table 1-2 lists the Design menu operations.

Table 1-2 Design Menu Operations

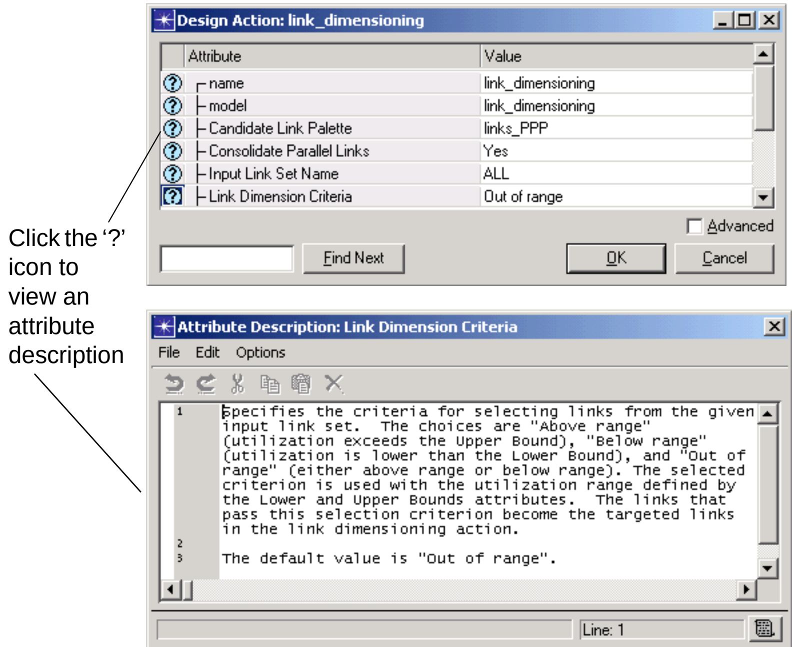
Menu item		**Reference**
Configure/Run Design Action	Select, configure, and run a design action	Configuring a Design Action on page DA-1-3
Run Design Action	Select and run a design action without configuring it	Running a Design Action on page DA-1-8
Quick Actions	Run a design action directly from the menu, without configuring it	Quick Actions on page DA-1-8
Results	View Latest Log—Displays the log messages generated by the last design action run. View Latest Reports—Displays the results of the last design action run. View Logs—Displays a list of log files generated by design action runs that were performed on the current project-scenario. From the list, you can select a log file to view. View Reports—Displays a list of results generated by design action run that were performed on the current project-scenario. From the list, you can select a set of results to view.	—
End of Table 1-2		

Configuring a Design Action

To configure a design action, choose Design > Configure/Run Design Action. The Configure/Run Design Action dialog box (Figure 1-3) shows a treeview of all available actions, a description of the selected action, and the list of visible attributes for the selected actions.

Every design action has one or more attributes that you can configure from this dialog box. You can also save a configured action under a new name, and thus create your own set of configured actions.

Figure 1-1 Attribute Table for a Design Action (Example)



Viewing Information about a Design Action

Because a design action can cause extensive changes to a scenario, it is good practice to research the effects and options of a design action thoroughly before you run it. There are three primary sources of information about a specific design action:

- Documentation—This manual includes chapters about specific design actions such as link dimensioning.
- Action description—To view the action description, open the design action in the Configure / Run Action Dialog box and click the View Comments button.
- Attribute descriptions—To view descriptions of individual attributes, open the design action in the Configure / Run Action Dialog box; then click Edit Attributes. To view information about a specific attribute, click the Help ('?') icon as illustrated in Figure 1-1

Compound Actions, Roles, and Subactions

A *compound action* is a design action that can run other actions. A compound action has the following characteristics:

- Each compound action has one or more *roles* that define function blocks in the flowchart of the action. The function block is known by its role name, and runs other actions according to the internal flowchart of the compound action.

- The order in which the roles are listed in the user interface is not significant. Each compound action has its own internal flowchart that defines when and how each role is used.
- When an action is run by a role, it is called a *subaction*. If a compound action needs to execute a function block, it retrieves the subactions assigned to that role and runs each subaction in sequence.
- You can assign zero, one, or multiple subactions to a role. You might not want or need to assign a subaction to every role; a compound action skips over any role that has no subaction assigned. If a role has multiple subactions assigned, it runs subactions in the order in which they are listed.

Figure 1-2 Subactions Table (Example)

Role	Action(s)	
Initializer	link_pricer_2kb_100km	
Link Selector	link_price_selector_geq_25	
Link Action	remove_links	
Link Analyzer	link_price_analyzer	Click to configure...
Finalizer		

The Subactions table appears in the bottom right in the Configure Design Action dialog box. This table lists the roles and subactions defined for a specific compound action. To assign one or more subactions to a role, click in the Action(s) field for that role. In Figure 1-2, the compound action is configured to run four subactions:

- Set the financial cost on each link (“Initializer”)
- Select all links with a financial cost greater than 25 (“Link Selector”)
- Remove these links from the network (“Link Action”)
- Analyze the cost of the remaining links (“Link Analyzer”)

Derived Actions

A *derived action* is an action with customized attribute settings. You might want to create your own derived actions to perform common actions. For example, you might want to create actions to price links according to different criteria. You can derive actions from the link_pricer base action and then run the derived actions as needed.

Procedure 1-1 Creating a Derived Action

1. Choose Design > Configure/Run Design Action and select the action you want to use as a base model.
2. Configure the attributes and subactions (if any) for the base model.

3 Click Save As and save the action under a new name.

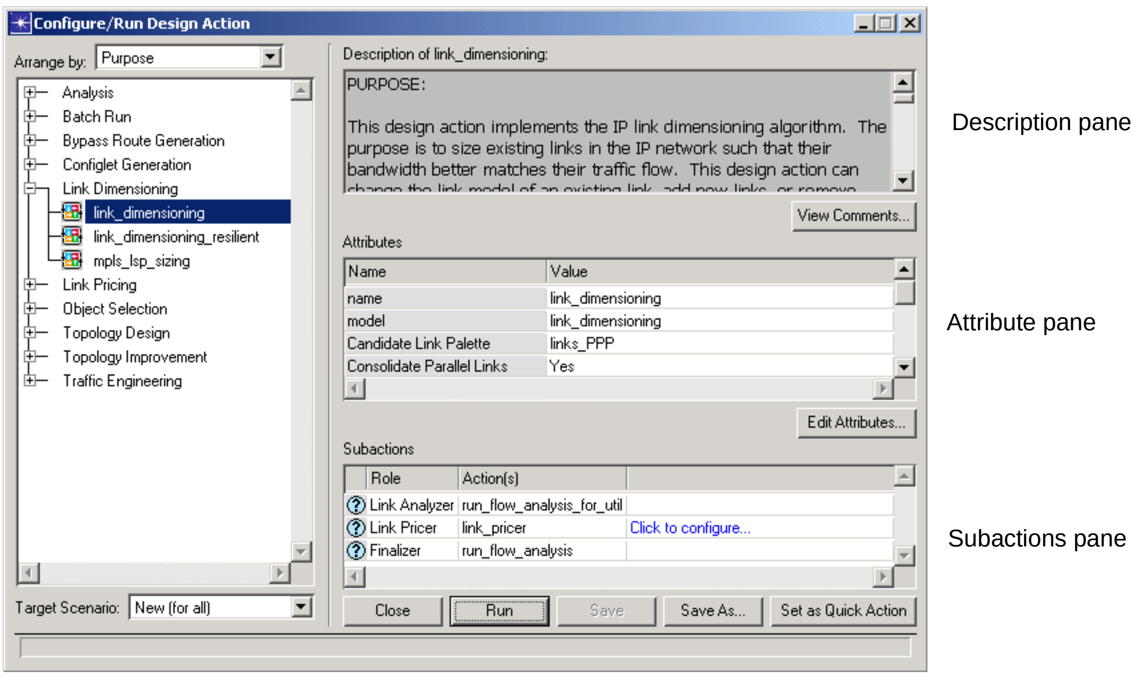
Note—You cannot overwrite a base action in the Project Editor; you can only save it under a new name. For this reason, the Save button is disabled when you configure a base action.

End of Procedure 1-1

Configure/Run Design Action Dialog Box

Figure 1-3 shows the Configure/Run Design Action dialog box and indicates the types of information that appear in the panes of the dialog box. The display and control elements in this dialog box are described in Table 1-3 on page DA-1-7.

Figure 1-3 Configure/Run Design Action Dialog Box



Treeview pane

Table 1-3 Configure/Run, Design Action Dialog Box

Item	Description
Panes: • Treeview (left) • Action description (right top) • Attributes table (right center) • Subactions table (right bottom, compound actions only)	 Select an action in this pane to view, configure, or run. Description of the selected action. Visible attributes of the selected action. Click Edit to configure these attributes. To assign one or more subactions to a role, click in the Action(s) field for that role. The Choose Subactions dialog box lists all assignable action models. To be assignable, an action model must have a Subaction Type self-description value that matches the role type.
Arrange By menu	Arrange the treeview based on the self-descriptions of action models. Thus you can arrange actions based on author, technology, or some other criteria.
Run	Run the selected action. If the action cannot be run directly, this button is disabled.
Save	Save the action based on the current configuration. If you change the configuration of a standard action, this button is disabled. You must choose Save As and save the current configuration under a new name.
Save As	Save the current configuration under a new name.
Set as Quick Action	Create a Quick Action from the current action and configuration. For more information, see Quick Actions on page DA-1-8.
Target Scenario menu	Choose the target scenario for the design action. The target scenario can be either the current scenario or a new scenario. **NOTE—**It is good practice to choose a new target scenario for any design action that can modify the network. This ensures that you retain a copy of the original (pre-design action) network state.
End of Table 1-3	

Running a Design Action

Every design action has an attribute called “runnable,” which specifies whether you can run the action directly. If this attribute is set to FALSE, the action can be run by a compound action only (that is, as a subaction). This means that

- when you choose Design > Run Design Action, only runnable actions appear.
- when you choose Design > Configure/Run Design Action and select an unrunnable action, the Run operation is unavailable.

You can run an action using any of the following operations:

1) Choose Design > Run Design Action (see Table 1-4 Run Action Dialog Box on page DA-1-9)

2) If you want to configure an action before running it, choose Design > Configure/Run Design Action (see Table 1-3 Configure/Run, Design Action Dialog Box on page DA-1-7)

3) If you have an action that you want to run quickly and repeatedly, you might want to create a quick action. You can then run the action from the Design > Quick Actions submenu (see Quick Actions on page DA-1-8)

Quick Actions

A *quick action* is an action that you can run directly from the Design > Quick Actions submenu. When you choose an action in this submenu, the action executes immediately. You can create your own quick actions, and thereby create your own “action toolbox” for a specific scenario.

Note—Each Project Editor session maintains its own Quick Actions list, so the Design > Quick Actions submenu is persistent within the current Project Editor session only.

Procedure 1-2 Creating a Quick Action

1 Choose Design > Quick Actions > Create New Quick Action to open the Configure/Run Design Action dialog box.

2 Select the action you want to use as the base for your quick action.

Note—You can only create a quick action from an action whose “runnable” attribute is set to TRUE. If you select an action and the Run button appears dimmed, this means that the action (as currently configured) is not runnable. If the “runnable” attribute is not visible, then you cannot use that action to create a quick action.

3 Click Edit.

4 Configure the attributes so that the action performs the behavior you want.

Note—Because a quick action executes automatically, you must specify the exact behavior you want the quick action to perform.

5 Verify that the “runnable” attribute is set to TRUE, then click OK to return to the Configure/Run Design Action dialog box.

6 Click Save As and save the action under the name you want to use for the quick action.

7 Click Quick Action.

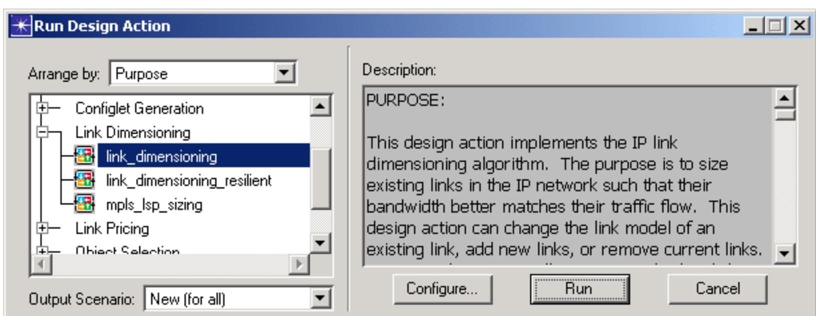
➥ The action appears in the Design > Quick Actions submenu.

End of Procedure 1-2

Run Action Dialog Box

Figure 1-4 shows the Run Action dialog box and indicates the types of information that appear in the panes of the dialog box.

Figure 1-4 Run Action Dialog Box



Treeview pane (only runnable actions appear in this tree)

Table 1-4 lists the display and control elements in the Run Action dialog box.

Table 1-4 Run Action Dialog Box

Item	Description
Treeview pane (left)	Select an action in this pane to configure or run.
Action description (right top)	Description of the selected action
Configure	Configure the attributes and subactions (if any) before running the action.
Run Design Action	Run the selected action.
End of Table 1-4	

Viewing Results

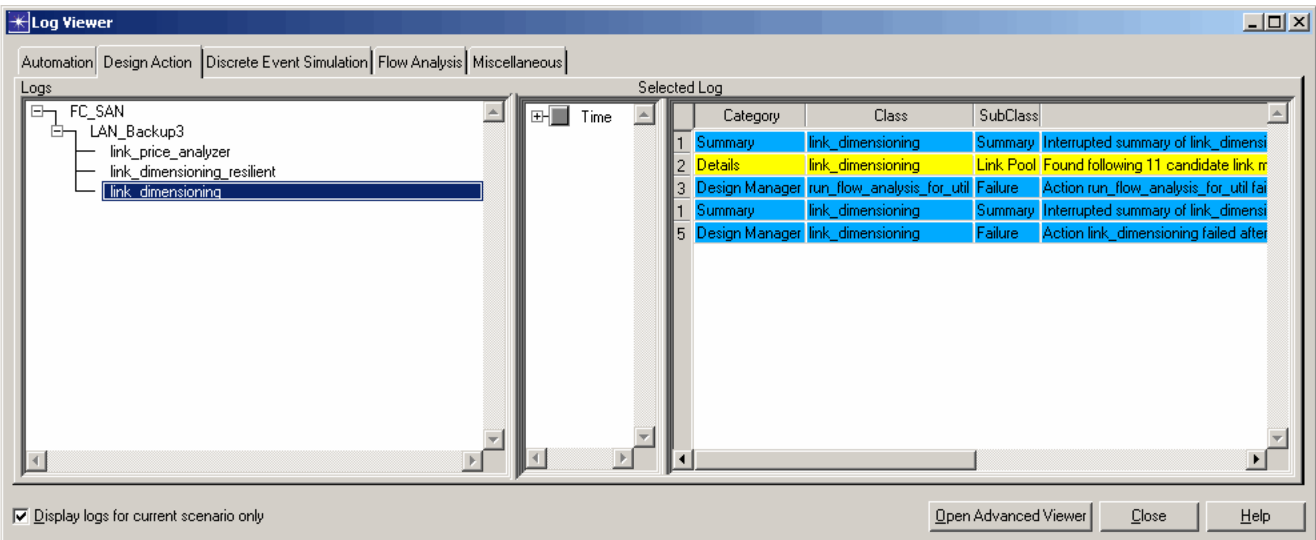
You can view the results of a design action using two different reporting mechanisms:

- The Design Action Log shows all network changes and significant events generated by the last action run; see Design Action Log for more information.
- In some cases, a design action generates an output table of results. You can view these tables by choosing Design > Results > View Last Report or Design > Results > View Reports. For general information about viewing results, see Results Viewer on page SPU-14-46 of the *User Guide*.

Design Action Log

The design action log (Figure 1-5) shows all network changes and significant events generated by the last action run. The messages that appear in this log depend on the specific action. Generally, actions generate Summary messages to provide a high-level snapshot of the action results. Because a design action can generate many types of log messages, you might want to filter the log to show only messages of interest.

Figure 1-5 Design Action Log



Design action logs are displayed in the SP Guru log viewer (as described in Log Viewer on page SPU-4-22). You can view a design action log using one of these methods:

- If a design action completes successfully: click the View Log button in the Action Completed dialog box.
- To view the design action log for the last run: choose Design > Results > View Latest Log.
- To view any available design action log: choose Design > Results > View Logs…

Custom Design Actions

If you have a license for the OPNET Development Kit, you can create your own custom design models using the Design Action Editor. You can open this window in one of two ways:

- To create a new action model, open the New File dialog box (File > New) and select Design Action.
- To edit an existing action model, open the Open File dialog box (File > Open) and select Design Action.

A design action includes the following components:

- Implementation—The implementation is C or C++ code that defines the internal behavior of the model. Each base action must have an implementation for its run function. This implementation is stored within the model file. An action model can also use code defined in external C/C++ or object files. The external files may be shared among multiple action models. Derived models inherit their base models implementation and cannot define their own code.

  The OPNET Development Kit includes a set of Design Action-related functions that you can use to develop your own implementations. To view the declarations for these functions, go to the `<reldir>/sys/include` directory and look in the `odan*.h` set of header files.

- Attributes—To define the attributes for an action, choose Design Action > Edit Attributes.
- Self-Description—The self-description of a model determines how the action appears in the action treeview of the Run Design Action and Configure Design Action dialog boxes. To edit the self-description, choose Design Action > Edit Self-Description.
- Subactions—To edit the roles and subactions for a compound action, choose Design Action > Edit Subactions.

2 Link Dimensioning

The objective of a link_dimensioning run is to determine the best link configuration that accommodates the current traffic patterns in an IP network. This design action can add, remove, reconnect, or replace links, but does not change the existing network connectivity—that is, it will never remove the last link between two nodes or add a link between unconnected nodes.

A link_dimensioning run starts by analyzing the existing network and traffic patterns. Then it upgrades, downgrades, or leaves unchanged the bandwidth for each link bundle in the selection set; a *link bundle* is a set of one or more links between the same node pair. For each link bundle, the design action calculates the best solution based on the following criteria:

1) The link bundle accommodates current traffic patterns.

2) The dimensioned links are neither over-utilized nor underutilized—that is, link utilizations fall within a specified range.

3) If cost parameters are specified, the solution results in the lowest possible cost while still meeting the criteria.

To adjust the bandwidth of a link bundle, link_dimensioning can do any of the operations described in Table 2-1. If link_dimensioning cannot find a satisfactory solution for an individual link bundle, it generates a log message; for more information, see Log Messages on page DA-2-4.

Table 2-1 Operations on Individual Links

Operation Type	Details	Before	After
Replace	Change the data rate or model of the link, but do not change the port/interface connections to its end nodes	Link A	Link A with new model
Move	Change the data rate or model of the link, and also change the port/interface connections to its end nodes. This happens when the new link model cannot use the original ports.	Link A	Link A with new model and ports
Remove	Remove the link from the current scenario	Link A	None
Add	Create a new link and copy its configuration from a parallel link that existed before the current design-action run. Certain configurations such as IP address are not copied.	None	New link object with new model and ports. Configuration is based on the most recent target link.
End of Table 2-1			

Prerequisites

Like all design actions, link_dimensioning is an advanced feature.This section describes the recommended background knowledge you should have before you run this design action.

- Link Dimensioning Tutorial

  The Guru documentation includes a tutorial about link dimensioning. It is recommended that you complete this tutorial before you run a link_dimensioning design action. For more information, see Design Actions: Link Dimensioning tutorial.

- Background Knowledge

  To use this design action most effectively, you should have some familiarity with the following topics:

  — Routers, ports, and their supported links

  — The relationships between link utilizations and IP traffic flows

  — IP routing protocols and their load-balancing options

  — OPNET node and link models

  — The Flow Analysis module

- Design Action and Attribute Descriptions

  In addition to the information in this chapter, you should also view the action description and individual attribute descriptions available in the Configure/Run Design Action dialog box. For more information, see Viewing Information about a Design Action on page DA-1-4.

Configuring and Running the link_dimensioning Design Action

This design action sizes links based on the following inputs:

- In the initial IP network:

  — Nodes, links, and link connectivity

  — Available ports and supported interface types on existing nodes

  — Initial traffic levels (specified by the existing traffic flows)

  — Initial link utilizations (determined by running Flow Analysis)

- Attribute settings in the link_dimensioning design action. Attributes of interest include:

  — The specified target utilization range (upper bound and lower bound)

  — The set of input links (all links in the scenario or a subset of links)

  — The available link models (defined by the candidate link palette for the design action)

  — Link pricing criteria (optional–specified in the link_pricer subaction)

For more information, see

- Configuring a Design Action on page DA-1-3
- Running a Design Action on page DA-1-8

Log Messages

The link_dimensioning design action generates several types of log messages to show the summary and network changes. Table 2-2 and Table 2-3 list these log messages.

Table 2-2 Categories of link_dimensioning Log Messages

Use This Category...	To View...
Detail	Candidate link models considered when adding or replacing individual links.
Design Manager	Which subactions ran successfully.
Network Changes	Which links bundles were upgraded, downgraded, and unchanged.
Warning	Warning messages generated during the run.
Summary	A list of statistics that were generated before and after the run.
End of Table 2-2	

Table 2-3 Classes of link_dimensioning Log Messages

Use This Class...	To View...
Link Pool	Candidate link models considered when adding or replacing individual links.
Unchanged	Link bundles whose bandwidths were unchanged by the design action.
Upgraded	Link bundles whose bandwidths were upgraded by the design action.
Downgraded	Link bundles whose bandwidths were downgraded by the design action.
Summary	Summary statistics about the design action
Success	Whether the design action ran successfully.
End of Table 2-3	

Unsatisfactory Solutions

In some cases, the new solution for a specific link bundle still cannot satisfy the utilization range and criteria. These cases can be categorized in two types:

- Unchanged - A link bundle is selected because it fit in selection criteria, but link dimensioning action cannot change it due to different constraints. Detailed reasons are listed in Table 2-4.

- Upgrade/downgrade but still Invalid - The new solution is not valid because no valid solution can be found within current constraints. A best invalid solution is provided and all possible reasons described in Table 2-4.

Table 2-4 Messages for Unsatisfactory Solutions

ID	Bundle status	Utilization Range	Reason	Message
0	Unchanged		No candidate links are compatible	The initial link bundle has this message as default. For bundles below range, an additional reason is that no smaller links are available (log ID #5). In this case, the initial solution uses small-capacity links and does not meet the utilization range. During the process of finding another appropriate solution, the target link bundle retains this message by default. Whenever a compatible link model is selected and tested, the design action removes this message.
1	Upgraded or unchanged	Above or within	Limited port in the first node	These three reason codes are added during the solution search process. Whenever a compatible link model is considered for inclusion in the solution, these reason codes might be added into reason code of the new solution
2	Upgraded or unchanged	Above or within	Limited port in the second node	
3	Upgraded or unchanged	Above or within	Limited by the maximum number of parallel links	
4	Upgraded or unchanged	Above	Not compliant with the utilization upper bound	This message is generated when the action finds no feasible solution under current constraints, and uses the best infeasible solution: use the largest candidate links to give the lowest maximum link utilization.
5	Downgraded or unchanged	Below	No smaller links available	This message is generated when the action finds no feasible solution under current constraints, and uses the best infeasible solution: use the smallest candidate links to give the highest minimum link utilization.
6	Upgraded or Unchanged	Above	Link Analyzer problem	Current solution should be feasible but the Link Analyzer subaction might give different results than expected. For examples that illustrate how this can occur, see the Design Actions: Link Dimensioning tutorial
7	Unchanged	Above or Below	Limited by algorithm attribute	Current solution is unchanged because the maximum number of links dimensioned per iteration is reached.
8	Warning	Any state	Unspecified reason	Unknown reason, please report to OPNET technical support.
End of Table 2-4				

Viewing Reports

After a link_dimensioning run finishes, you can view reports about the run from the View Results window (Design > Results > View Output Reports). Table 2-5 describes the available reports.

Table 2-5 link_dimensioning Reports

Report Name	Description
Link Bundle Summary	Shows the operations performed on each link bundle during the run. The Link Bundle Operation column includes links that enable you to see details about an individual operation (how a link bundle was upgraded, or why a link bundle was unchanged).
Design Action Statistics	A list of statistics that were generated during the run.
Object Tables	Intermediate statistics for individual failure cases.
End of Table 2-5	

3 Link Dimensioning Resilient

The objective of a link_dimensioning_resilient run is to determine the best link configuration that will accommodate current traffic patterns in the normal case (no link failures) and in a set of defined failure cases. Like link_dimensioning, link_dimensioning_resilient can add, remove, reconnect, or replace links, but does not change the existing network connectivity—that is, it never removes the last link between two nodes or adds a link between unconnected nodes. link_dimensioning and link_dimensioning_resilient differ in the following ways:

- link_dimensioning considers the normal case (no failures) only; link_dimensioning_resilient considers the normal case and a set of failure cases, and sizes the links to accommodate traffic for all failure cases.
- link_dimensioning can upgrade or downgrade links, or leave links unchanged; link_dimensioning_resilient can upgrade links or leave them unchanged (it never downgrades links).

Prerequisites

Like all design actions, link_dimensioning_resilient is an advanced feature. Before you work with link_dimensioning_resilient, you should be familiar with the link_dimensioning design action. For more information, see the Link Dimensioning.

Configuring and Running the link_dimensioning_resilient Design Action

This design action sizes links based on the following inputs:

- In the initial IP network:
  - — Nodes, links, and link connectivity
  - — Available ports and supported interface types on existing nodes
  - — Initial traffic levels (specified by the existing traffic flows)
  - — Initial link utilizations (determined by running Flow Analysis)
- Attribute settings in the link_dimensioning_resilient design action. Attributes of interest include:
  - — Specified failure cases (For more information, see Failure Cases)
  - — The maximum utilization (upper bound)
  - — The set of input links (all links in the scenario or a subset of links)
  - — The available link models (defined by the candidate link palette for the design action)
  - — Link pricing criteria (optional–specified in the link_pricer subaction)

Failure Cases

A *failure case* is a scenario in which a set of one or more network objects is failed. link_dimensioning_resilient includes attributes that define the failure cases to consider when the design action is run. During a run, link_dimensioning_resilient sizes each link to accommodate current traffic levels in the normal case (no link failures) and in each possible failure case.

You can configure the following characteristics of failure scenarios:

- Failure object type—A failure case can fail links, nodes, and shared risk groups (SRGs). A shared risk group defines a failure relationship between two sets of network objects, where failures in one set cause failures in the other set. For more information, see Shared Risk Groups on page SPU-6-43 of the SP Guru *User Guide*.
- Failure object set—–This is the set of all objects to be failed during the run. You can include failure cases for all eligible objects or a subset of eligible objects.
- Failure object occurrence—You can specify a run to test single failure scenarios (one object failed), pairwise scenarios (two objects failed), or simultaneous scenarios (all eligible objects failed simultaneously).

Note—A link_dimensioning_resilient run can take several minutes or more to complete, depending on the size and complexity of your network. Factors that can increase run times include: a large number of input links, a complex network with many links and node, a run that tests pairwise failures, and a large number of repetitions for each failure case.

For general information about configuring and running design actions, see

- Configuring a Design Action on page DA-1-3
- Running a Design Action on page DA-1-8

Viewing Logs

The link_dimensioning_resilient design action generates several types of log messages to show the summary and network changes. Table 3-1 and Table 3-2 list these log messages.

Table 3-1 Categories of link_dimensioning_resilient Log Messages

Use This Category...	To View...
Detail	Intermediate statistics for individual failure cases
Design Manager	Which subactions ran successfully
Network Changes	Which links bundles were upgraded, downgraded, and unchanged
Warning	Warning messages generated during the run
Summary	A list of statistics that were generated before and after the run
End of Table 3-1	

Table 3-2 Classes of link_dimensioning_resilient Log Messages

Use This Class...	To View...
Link Pool	Candidate link models considered when adding or replacing individual links
Unchanged	Link bundles whose bandwidths were unchanged by the design action
Upgraded	Link bundles whose bandwidths were upgraded by the design action
Summary	Summary statistics about the design action
Statistics	Intermediate statistics for individual failure cases
Success	Whether the design action ran successfully
End of Table 3-2	

Unsatisfactory Solutions

When it determines an unsatisfactory solutions, the link_dimensioning_resilient design action generates messages similar to link_dimensioning; for more information, see Unsatisfactory Solutions on page DA-2-4. (Because it never downgrades links, link_dimensioning_resilient does not generate “Downgrade” messages.)

Viewing Reports

After a link_dimensioning_resilient run finishes, you can view reports about the run by choosing Design > Results > View Latest Reports. Table 3-3 describes the available reports.

Table 3-3 link_dimensioning_resilient Reports

Report Name	Description
Failure Cases	Provides a summary of all failure cases considered in the link_dimensioning_resilient run. If a single failure case has multiple failed objects, the failure case contains multiple rows.
Link Bundle Summary	Shows the operations performed on each link bundle during the run. The Link Bundle Operation column includes links that enable you to see details about an individual operation (how a link bundle was upgraded, or why a link bundle was unchanged).
Summary Statistics	A list of statistics that were generated before and after the run.
Intermediate Statistics	Intermediate statistics for individual failure cases
End of Table 3-3	

4 Ring Backbone Topology Design

The design actions for topology design help you determine where to place links in a network. Three design actions are available, depending on the type of network you are building. You can use the topology design actions to a create a ring-based backbone, a spanning tree, or dual spanning trees. This chapter describes the ring_backbone design action.

The topology design actions work on network topologies that contain only nodes. This type of design is also referred to as a Greenfield design. You can also use the design actions on networks that have existing links or traffic demands. In this type of incremental design, the design action can use the existing links when it determines where to place additional links.

When the design action adds links to a network, it considers the following:

- cost
- connectivity
- hop count
- utilization
- port constraints
- distance

The ring_backbone design action lets you build network designs that are resilient to failure and insensitive to traffic. These network designs provide paths that have low delays and low cost. The design action builds the network designs by constructing a near-symmetric 2-connected or 3-connected topology with a small diameter. The diameter is the maximum number of hops between any node pair.

Symmetric topologies are less sensitive to traffic fluctuations in demands. A symmetric topology can be an effective design for a backbone when you do not have any traffic information, when the traffic is very variable, or when you are planning for future applications. The design action can use existing traffic information to bias the design to have direct links between high-traffic pairs.

Prerequisites

This section describes what you need to know before you run the ring_backbone design action.

Background Knowledge

To use this design action most effectively, you should be familiar with the following topics:

- Topology design, including greenfield and brownfield design
- Routers, ports, and their supported links
- OPNET node and link models

Design Action and Attribute Descriptions

In addition to the information in this chapter, you should also view the action description and individual attribute descriptions available in the Configure/Run Design Action dialog box. For more information, see Viewing Information about a Design Action on page DA-1-4.

Configuring and Running the Ring Backbone Design Action

This design action creates links based on the following inputs:

- In the initial network:
  - Nodes
  - Existing links, if any
  - Traffic demands, if any
- Attribute settings in the ring_backbone design action. Attributes of interest include:
  - The available link models (defined by the candidate link palette for the design action)
  - The target utilization
  - The type of ring topology (defined by the Graph Engine attribute)
  - Link pricing criteria (optional–specified in the link_pricer_custom_db subaction)

For more information, see

- Configuring a Design Action on page DA-1-3
- Running a Design Action on page DA-1-8

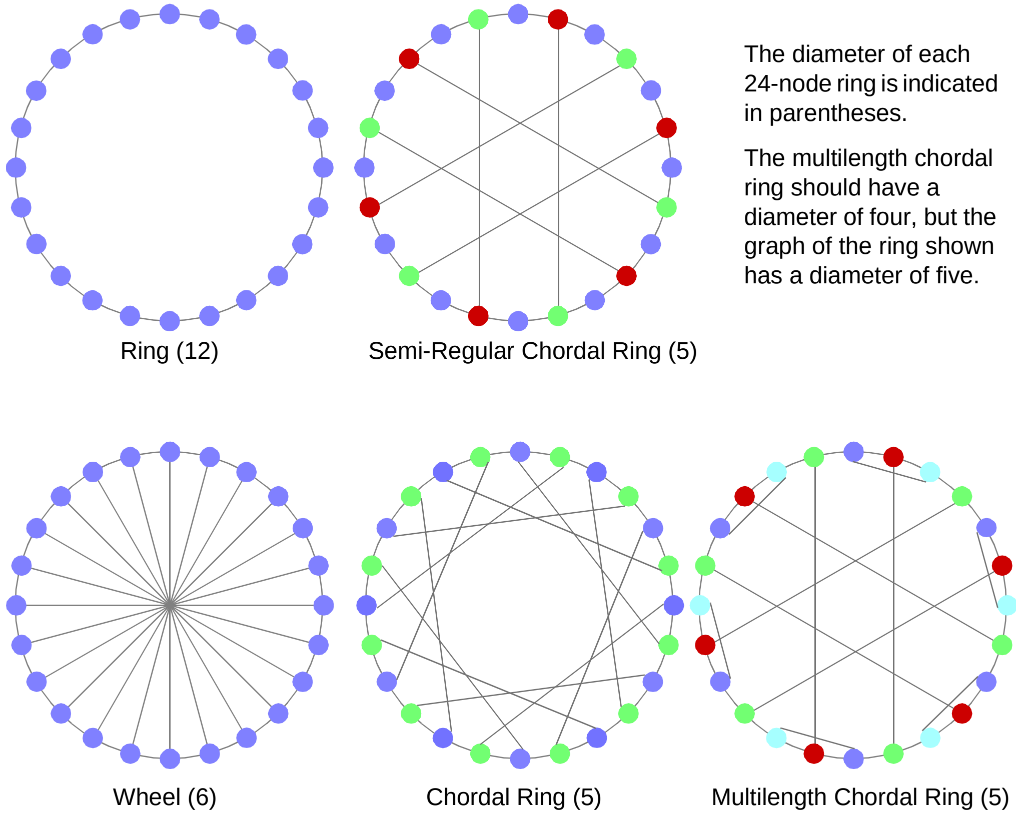
Graph Engine

The design action can create five types of ring backbone designs. All designs consists of a ring that contains all of the input nodes. The degree of connectivity within the ring differentiates ring types.

Table 4-1 Ring Backbone Graph Engine Characteristics

Graph Engine	Connectivity	Symmetry	Number of Edges	Description
Ring	2-connected	Symmetric	N	A ring that passes through each node
Semi-Regular Chordal Ring	2-connected	Near symmetric	5N/4	A ring with chords incident to every other node
Wheel	3-connected	Symmetric	3N/2	A ring with chords connecting each node to the node N/2 away in rotational distance around the ring
Chordal Ring	3-connected	Near symmetric	3N/2	A ring with chords connecting each node to the node +/- w away in rotational distance around the ring.
Multilength Chordal Ring	3-connected	Near symmetric	3N/2	Similar to the chordal ring, but with two chord lengths, w1 and w2, which minimize the diameter of the network.
End of Table 4-1				

Figure 4-1 Types of Ring Designs



The diameter of each 24-node ring is indicated in parentheses.

The multilength chordal ring should have a diameter of four, but the graph of the ring shown has a diameter of five.

Log Messages

The ring_backbone design action generates several types of log messages to show the summary and network changes. Table 4-2 and Table 4-3 list these log messages.

Table 4-2 Categories of Ring Backbone Design Log Messages

Use This Category...	To View...
Detail	Candidate link models considered when adding or replacing individual links
Design Manager	A list of subactions that ran successfully.
Network Changes	A list of the links that were created.
Error	Error messages generated during the run.
Warning	Warning messages generated during the run.
Summary	A list of statistics that were generated before and after the run. Statistics include: average cost per link, cost of new links, and the number of links added and reused.
End of Table 4-2	

Table 4-3 Classes of Ring Backbone Design Log Messages

Use This Class...	To View...
Link Pool	Candidate link models considered when adding or replacing individual links.
Filtered	Nodes that were excluded from the design. These are usually configuration objects or nodes with no available ports.
Created	Links that were added by the design action.
Reused	Links that existed in the topology and were reused by the design action.
Summary	Summary statistics about the design action.
Success	Whether or not the design action ran successfully.
End of Table 4-3	

5 Spanning Tree Topology Design

The design actions for topology design help you determine where to place links in a network. Three design actions are available, depending on the type of network you are building. You can use the topology design actions to a create a ring-based backbone, a spanning tree, or dual spanning trees. This chapter describes the spanning_tree design action.

The topology design actions work on network topologies that contain only nodes. This type of design is also referred to as a Greenfield design. You can also use the design actions on networks that have existing links or traffic demands. In this type of incremental design, the design action can use the existing links when it determines where to place additional links.

The spanning tree design action is useful for finding a near minimum cost topology that connects a set of nodes. This can serve as a baseline for understanding the lowest cost network option. The spanning tree may not meet the performance or redundancy requirements for your operational network design, but it may serve as a good starting point for building a final design.

When the design action adds links to a network, it considers the following requirements:

- cost
- connectivity
- hop count
- utilization
- port constraints
- distance

The spanning tree design action is very similar in functionality to the ring_backbone design action, which is described in Chapter 4 Ring Backbone Topology Design on page DA-4-1. The difference is that instead of using a ring-based graph as the network structure, the algorithm builds a spanning tree graph. A spanning tree graph is a connected, acyclic graph.

Prerequisites

This section describes what you need to know before you run the spanning_tree design action.

Background Knowledge

To use this design action most effectively, you should be familiar with the following topics:

- Topology design, including greenfield and brownfield design
- Routers, ports, and their supported links
- OPNET node and link models

Design Action and Attribute Descriptions

In addition to the information in this chapter, you should also view the action description and individual attribute descriptions available in the Configure/Run Design Action dialog box. For more information, see Viewing Information about a Design Action on page DA-1-4.

Configuring and Running the Spanning Tree Design Action

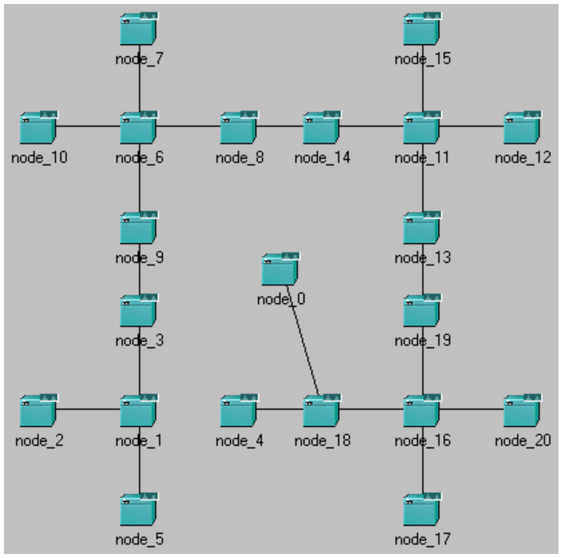
This design action creates links based on the following inputs:

- In the initial network:
  - — Nodes
  - — Existing links, if any
  - — Traffic demands, if any
- Attribute settings in the spanning_tree design action. Attributes of interest include:
  - — The available link models (defined by the candidate link palette for the design action)
  - — The target utilization
  - — The type of spanning tree (defined by the Spanning Tree Weight attribute)
  - — The root node (defined by the Root Criteria attribute)
  - — Link pricing criteria (optional–specified in the link_pricer_custom_db subaction)

For more information, see

- Configuring a Design Action on page DA-1-3
- Running a Design Action on page DA-1-8

Figure 5-1 Types of Spanning Trees



Log Messages

The spanning tree design action generates several types of log messages to show the summary and network changes. Table 5-1 and Table 5-2 list these log messages.

Table 5-1 Categories of Spanning Tree Design Action Log Messages

Use This Category...	To View...
Detail	Candidate link models considered when adding or replacing individual links
Design Manager	A list of subactions that ran successfully.
Network Changes	A list of the links that were created.
Error	Error messages generated during the run.
Warning	Warning messages generated during the run.
Summary	A list of statistics that were generated before and after the run. Statistics include: average cost per link, cost of new links, and the number of links added and reused.
End of Table 5-1	

Table 5-2 Classes of Spanning Tree Design Action Log Messages

Use This Class...	To View...
Link Pool	Candidate link models considered when adding or replacing individual links.
Filtered	Nodes that were excluded from the design. These are usually configuration objects or nodes with no available ports.
Created	Links that were added by the design action.
Reused	Links that existed in the topology and were reused by the design action.
Summary	Summary statistics about the design action.
Success	Whether or not the design action ran successfully.
End of Table 5-2	

6 Dual Tree Topology Design

The design actions for topology design help you determine where to place links in a network. Three design actions are available, depending on the type of network you are building. You can use the topology design actions to a create a ring-based backbone, a spanning tree, or dual spanning trees. This chapter describes the dual_tree design action.

The topology design actions work on network topologies that contain only nodes. This type of design is also referred to as a Greenfield design. You can also use the design actions on networks that have existing links or traffic demands. In this type of incremental design, the design action can use the existing links when it determines where to place additional links.

The dual tree design action is useful for finding a near minimum cost two-connected design for a given set of nodes. It builds on the low cost of the spanning tree design, but guarantees that there are at least 2 disjoint paths between all node pairs. The diameter and average hop distance are also generally significantly lower than the spanning tree.

When the design action adds links to a network, it considers the following requirements:

- cost
- connectivity
- hop count
- utilization
- port constraints
- distance

The dual tree design action is based on a spanning tree design. After constructing the first spanning tree, the design builds a second spanning tree by connecting the leaf nodes from the first tree. This creates a near-minimum cost 2-connected topology.

Prerequisites

This section describes what you need to know before you run the dual_tree design action.

Background Knowledge

To use this design action most effectively, you should be familiar with the following topics:

- Topology design, including greenfield and brownfield design
- Routers, ports, and their supported links
- OPNET node and link models

Design Action and Attribute Descriptions

In addition to the information in this chapter, you should also view the action description and individual attribute descriptions available in the Configure/Run Design Action dialog box. For more information, see Viewing Information about a Design Action on page DA-1-4.

Configuring and Running the Dual Tree Design Action

This design action creates links based on the following inputs:

- In the initial network:
  - — Nodes
  - — Existing links, if any
  - — Traffic demands, if any
- Attribute settings in the dual_tree design action. Attributes of interest include:
  - — The available link models (defined by the candidate link palette for the design action)
  - — The target utilization
  - — The types of spanning trees (defined by the Spanning Tree Weight attribute for the first tree and the Dual Tree Weight attribute for the second tree)
  - — The root node (defined by the Root Criteria attribute)
  - — Link pricing criteria (optional–specified in the link_pricer_custom_db subaction)

For more information, see

- Configuring a Design Action on page DA-1-3
- Running a Design Action on page DA-1-8

Log Messages

The dual tree design action generates several types of log messages to show the summary and network changes. Table 6-1 and Table 6-2 list these log messages.

Table 6-1 Categories of Dual Tree Design Action Log Messages

Use This Category...	To View...
Detail	Candidate link models considered when adding or replacing individual links
Design Manager	A list of subactions that ran successfully.
Network Changes	A list of the links that were created.
Error	Error messages generated during the run.
Warning	Warning messages generated during the run.
Summary	A list of statistics that were generated before and after the run. Statistics include: average cost per link, cost of new links, and the number of links added and reused.
End of Table 6-1	

Table 6-2 Classes of Dual Tree Design Action Log Messages

Use This Class...	To View...
Link Pool	Candidate link models considered when adding or replacing individual links.
Filtered	Nodes that were excluded from the design. These are usually configuration objects or nodes with no available ports.
Created	Links that were added by the design action.
Reused	Links that existed in the topology and were reused by the design action.
Summary	Summary statistics about the design action.
Success	Whether or not the design action ran successfully.
End of Table 6-2	

7 IP QoS Design Actions

This chapter provides high-level descriptions of the following IP QoS design actions:

- IP QoS Configuration
- IP QoS Queue Sizing

For more information about these design actions, open the Configure/Run Design Action dialog box (choose Design > Configure/Run Design Action in the Project Editor).

- For a complete description, select the design action and click View Comments.
- For information about individual attributes, click Edit Attributes. Then click the help (“question-mark”) button next to the attribute of interest.

IP QoS Configuration

You can use ip_qos_configuration to configure IP QoS parameters on routers in the network model. You can define QoS configuration templates and apply them to specific nodes and interfaces. QoS definitions in the design action (of traffic classes, traffic policies, queue profiles, and interface QoS schemes) are added to, but do not replace, existing definitions in the network. The only exception is if two definitions have the same name; in this case, the design action overwrites the existing definition.

IP QoS Queue Sizing

You can use ip_qos_queue_sizing to size the queue bandwidths or weights on IP interfaces based on the queue load and queue configuration rules. This design action uses Flow Analysis to compute the offered load to each queue based on the traffic flow. The design action attributes control how different queues are sized.

8 MPLS Traffic Engineering

The goal of traffic engineering is to align the flow of traffic through a network with the available resources of that network. The mpls_te design action does offline traffic engineering by finding the best primary and secondary explicit routes for the LSPs in an MPLS domain. This functionality is complementary to the MPLS features available with the MPLS specialized model.

The mpls_te design action uses the MPLS model attribute structure and its own design action attributes. Descriptions of all attributes are available in the Attributes dialog box by clicking on the ? icon or on the Details button.

Workflow

The goal of a traffic engineering design study is to determine the best routes for each LSP in the network given the topology of the network and other constraints. The design action computes two routes: a primary explicit route and a secondary explicit route, which is used if a failure occurs along the primary route.

Before you run a TE design, you should:

1) Build the network topology.

2) Create and configure the MPLS LSPs.

The best way to do these initial steps is by importing device configuration files. See Chapter 2 Importing Topologies on page MVI-2-1 in the *MVI User Guide* for additional information.

The best routes for each LSP are computed by examining a capacitated MPLS routing graph. To be included in this graph, a node or link must be configured to be in the MPLS domain. This configuration requires the following settings:

- a node must have its MPLS Parameters ➘ Status attribute enabled
- the interfaces connected to links in the MPLS domain must have the following attributes enabled:
  - RSVP Protocol Parameters ➘ Interface Information ➘ RSVP Status
  - MPLS Parameters ➘ Interface Information ➘ Status

A link-state routing protocol, either OSPF or IS-IS, must be designated in the MPLS domain.

The MPLS model supports two types of LSPs: dynamic and static.

- The MPLS process sets up paths for the dynamic LSPs by using RSVP to signal the setup messages along the path. Dynamic LSPs can also have explicit routes, which are configured at the source LSR and signalled dynamically at set-up.
- For static LSPs, the complete path is configured statically at every hop in the path. Static LSPs are not *routed* by the source LSP. Instead, the routes are completely preconfigured along the path.

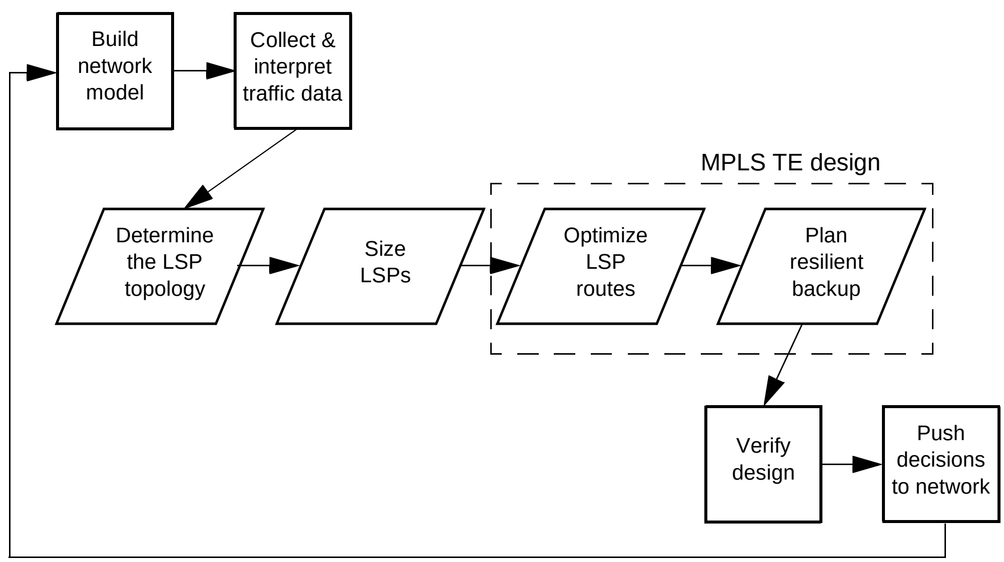
The design algorithm affects only dynamic LSPs. For each dynamic LSP, the design algorithm computes explicit routes—a primary route and optionally a secondary route—that define each hop in the path from the source to the destination.

You can specify which LSPs are included in the MPLS TE design computations. To exclude an LSP from the design, set its Design Constraint attribute to Locked. By default, this attribute is set to Unconstrained, which means that its explicit routes are computed by the design algorithm.

You can use the Connection Browser (Topology > Connection Browser) to show LSP routes calculated during a design action or Flow Analysis run. Each route appears as a child of its associated LSP path object. When you select a route in the tree view, the route is highlighted in the network pane.

Figure 8-1 shows how MPLS TE design fits into a workflow for MPLS TE deployment.

Figure 8-1 Example Workflow for MPLS TE Deployment



The workflow for using MPLS TE design to optimize LSP routes is as follows:

1) Configure the MPLS domain (Procedure 8-1).

2) Configure the parameters used in the optimization (Procedure 8-2).

3) Run the design algorithm to calculate the optimized routes (Procedure 8-3).

4) View the results. Results include the optimal routes and the link subscriptions associated with the routes.

Configuring the MPLS Domain

Procedure 8-1 Configuring the MPLS Domain for TE Design

1 Enable MPLS on all routers in the MPLS domain and on the router interfaces connected to links in the MPLS domain.

2 Configure RSVP on all router interfaces in the MPLS domain:

- Set the RSVP Protocol Parameters ➘ Interface Information ➘ RSVP Status attribute to Enabled
- Define the amount of bandwidth available for MPLS reservations by configuring the RSVP Protocol Parameters ➘ Interface Information ➘ Maximum Reservable BW and RSVP Protocol Parameters ➘ Interface Information ➘ Maximum Bandwidth Per LSP attributes.

3 Lock any LSPs that the TE design should not consider by configuring the Design Constraint attribute on the affected LSPs.

4 Configure the TE Parameters ➘ Min Bandwidth attribute on the LSPs. The other TE parameters are used by CSPF in flow analysis and discrete event simulations, but are not used by TE design.

5 Designate OSPF or IS-IS as the routing protocol on all interfaces in the MPLS domain.

End of Procedure 8-1

Configuring the Design Action Parameters

Design action parameters determine the characteristics or requirements for optimal routes. The parameters can be set locally (for specific LSPs) in the LSP ➘ TE Parameters attribute or globally (for all LSPs) as described in Procedure 8-2. Locally set constraints apply to both primary and secondary routes, and override any global settings for the same parameter.

Delay Constraints

The maximum delay constraint lets you use the MPLS TE design action to provide basic quality-of-service (QOS) guarantees. This constraint is defined as the total one-way path delay from the source node to the destination node. It does not include computed queuing or processing delays. Thus, for links with default delay settings, the one-way path delay equals propagation delay.

Maximum delay is a hard constraint. If it cannot be met for an LSP, the design action will not assign a route to that LSP and the LSP will be considered unrouted.

Hop Limits

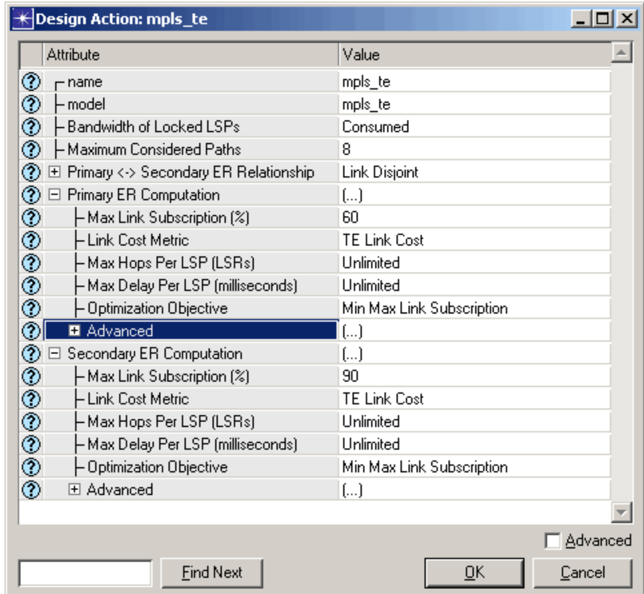
The hop limit constraint is defined as the maximum number of LSRs to allow in an LSP (including the source). Maximum hops is a hard constraint. If it cannot be met for an LSP, the design action will not assign a route to that LSP and the LSP will be considered unrouted.

Procedure 8-2 Configuring the Parameters Used in the Optimization

1. Click on the Configure/Run Design Action toolbar button.
2. In the Configure/Run Design Action dialog box, select the mpls_te design action in the tree view.

   ➥ The Description and Attributes panes display the mpls_te information.
3. Click on the Edit Attributes button.
4. In the Design Action: mpls_te dialog box, set the parameters for computing the primary and secondary explicit routes. You can find a description of each parameter by clicking on the “?” icon.

Figure 8-2 Setting the MPLS TE Design Parameters



5 Click OK.

End of Procedure 8-2

Running the Design Action

Use the following procedure to run the mpls_te design action.

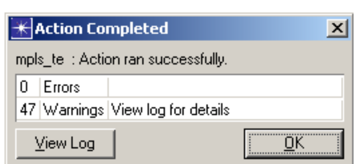
Procedure 8-3 Running the MPLS TE Design Action

1 Open the Configure/Run Design Action dialog box.

2 Under MPLS, select mpls_te.

3 Click Run.

➥ When the run is complete, the following dialog box appears.

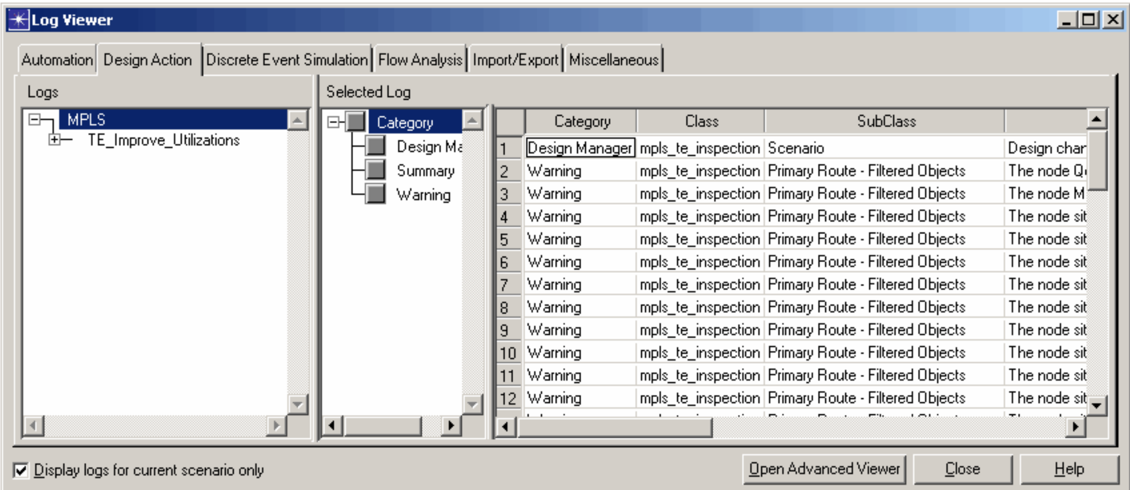


End of Procedure 8-3

Design Action Log

After a design action is run, there might be error or warning messages in the mpls_te design log (Figure 8-3). To examine the log, click the View Log button in the Action Completed dialog box or choose Design > Results > View Latest Log.

Figure 8-3   MPLS TE Design Log



You can expand the treeview in the left pane to sort the log messages. The following tables list the types of log messages that you can view or sort by in the log browser.

Table 8-1   Categories of mpls_te Log Messages

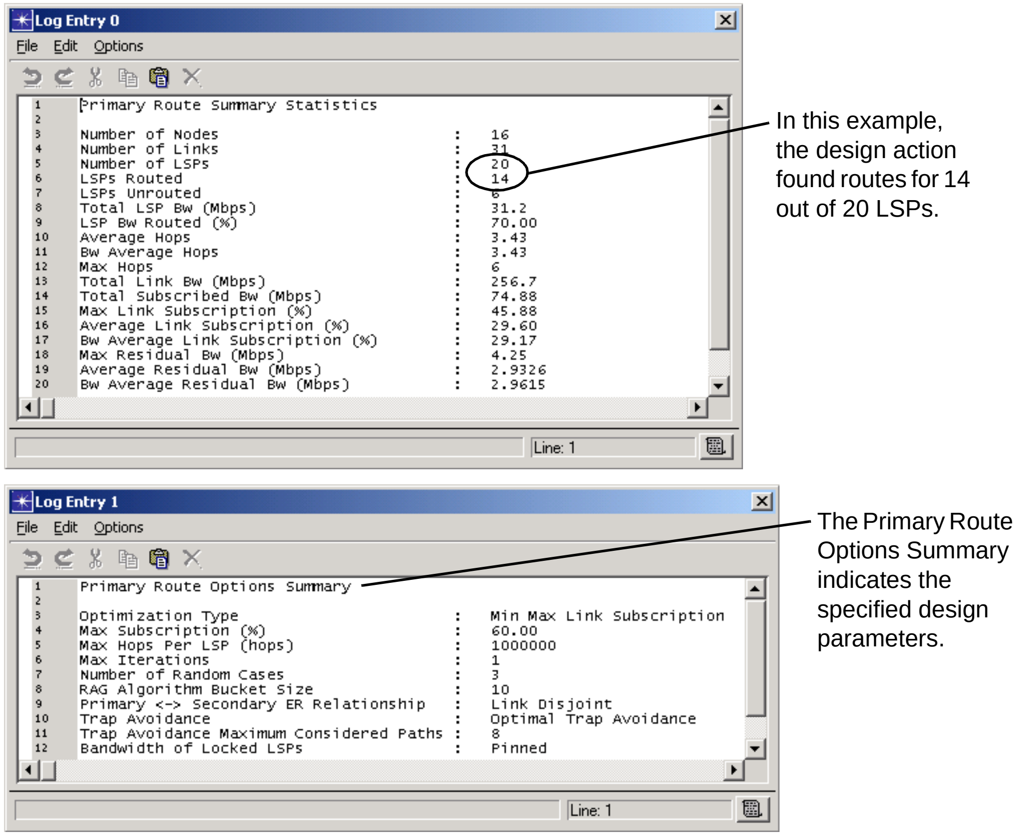
Use This Category...	To View...
Detail	The routed, unrouted, and filtered log messages.
Network Changes	The changes made to the network by the design action.
Summary	General information about the primary and secondary routes that were computed as well as the configuration information that was used to compute the routes.
Error	A list of errors that were generated while computing the routes.
Warning	A list of warnings that were generated while computing the routes.
Design Manager	A list of each design action that has been run and whether the action completed successfully or with failures.
End of Table 8-1	

Table 8-2 Classes of mpls_te Log Messages

Use This Class...	To View...
Primary Route – Filtered Objects Secondary Route – Filtered Objects	A list of objects that were not included in the route computation.
Primary Route – Routed Secondary Route – Routed Primary Route – Unrouted Secondary Route – Unrouted	Whether or not a route was computed.
Primary Route – Locked Secondary Route – Locked	Locked LSPs that are configured with a valid explicit route. The design action does not modify or try to find routes for locked LSPs. mpls_te checks the explicit route configuration to make sure that a route is configured and verifies the route to make sure that it is valid.
Primary Route – Locked Empty Secondary Route – Locked Empty	Locked LSPs that did not have an explicit route configured and locked LSPs whose configured explicit route was invalid.
Primary Route – Locked Overflow Secondary Route – Locked Overflow	Locked LSPs that are oversubscribed for the configured explicit route.
Primary Route – Route Configuration Secondary Route – Route Configuration	Whether or not the LSP was successfully configured. These messages contain information about the configuration changes made in the network.
Primary Route – Summary Secondary Route – Summary	General information about the primary or secondary routes that were computed.
Primary Route – Options Summary Secondary Route – Options Summary	General information about the configuration information that was used to compute the primary or secondary routes.
End of Table 8-2	

The information contained in many of the log messages are also contained in the web reports generated after a design is run.

Figure 8-4 Summary Log Messages



Available Reports

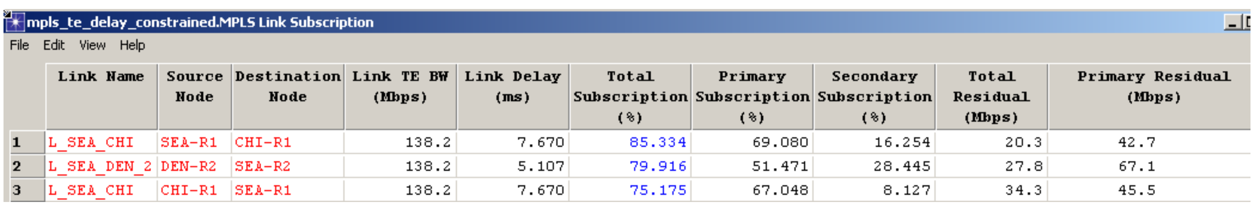
Detailed results are displayed as output tables in the Results Viewer, as described in Viewing Results on page DA-1-10. The mpls_te design action generates the following reports:

- MPLS Link Subscription on page DA-8-9
- MPLS Link Subscription Bidirectional on page DA-8-10
- MPLS LSP Explicit Routes Summary on page DA-8-12
- Primary Route on page DA-8-13
- Secondary Route on page DA-8-14
- Contained LSPs on page DA-8-15

MPLS Link Subscription

The link subscription report contains results for each link in the MPLS domain. The results given in the MPLS Link Subscription report are described in Table 8-3.

Figure 8-5 MPLS Link Subscription Report



	Link Name	Source Node	Destination Node	Link TE BW (Mbps)	Link Delay (ms)	Total Subscription (%)	Primary Subscription (%)	Secondary Subscription (%)	Total Residual (Mbps)	Primary Residual (Mbps)
1	L_SEA_CHI	SEA-R1	CHI-R1	138.2	7.670	85.334	69.080	16.254	20.3	42.7
2	L_SEA_DEN_2	DEN-R2	SEA-R2	138.2	5.107	79.916	51.471	28.445	27.8	67.1
3	L_SEA_CHI	CHI-R1	SEA-R1	138.2	7.670	75.175	67.048	8.127	34.3	45.5

Table 8-3 MPLS Link Subscription Report Results (Part 1 of 2)

Result	Description
Link Name Source Node Destination Node	Each row in the report represents a link in the network. Since the information can be different on the link in each direction, each link is shown in two different rows. The Source Node Name and Destination Node Name indicates the direction of the link shown in this row. Links are sorted in descending order of Total Subscription.
Link TE BW (Mbps)	The amount of bandwidth on the link that is available for MPLS traffic engineering. This is based on the data rate of the link and the percentage of the bandwidth available to MPLS configured in the RSVP Protocol Parameters ➘ Interface Information for the source interface.
Link Delay (ms)	The total one-way delay over the link.

Table 8-3 MPLS Link Subscription Report Results (Part 2 of 2)

Result	Description
Total Subscription (%)	The total percentage of this link's Link TE Bw that is reserved by LSP routes computed by the MPLS TE design. The total is the sum of the primary and secondary subscription percentages. It is based on the worst-case failure scenario. For example, on a link disjoint design, the total subscription is the highest subscription on this link that would occur under any single link failure. Click on a value in this column to open the Contained LSPs report for a link.
Primary Subscription (%)	The percentage of this link's Link TE Bw that is reserved by primary LSP routes computed by the MPLS TE design. This is the expected working subscription of the link when each LSP is using its primary route.
Secondary Subscription (%)	The percentage of this link's Link TE Bw that is reserved to provide for secondary LSP routes. This number does not include the primary subscription. The secondary routes are computed using a shared protection scheme that allows multiple LSPs to share secondary route bandwidth as long as their primary routes do not share a common failure condition. The Secondary Subscription is based on a worst-case failure scenario.
Total Residual BW (Mbps)	The residual bandwidth is the amount of bandwidth not reserved for LSPs on a link. Total residual bandwidth is bandwidth on a link not used under any failure scenario. Total Residual Bw = Link TE Bw - Total Subscription * Link TE Bw/100.0.
Primary Residual BW (Mbps)	The bandwidth on a link not reserved by primary routes. Primary Residual Bw = Link TE Bw - Primary Subscription * Link TE Bw/100.0
End of Table 8-3	

MPLS Link Subscription Bidirectional

The bidirectional link subscription report contains results for each link in the MPLS domain. This report combines the information given for each direction in the unidirectional report so that you can view the worst-case values for each link.

The results given in the link subscription report are described in Table 8-4. You can also refer to the corresponding descriptions of the unidirectional report in Table 8-3 for additional information.

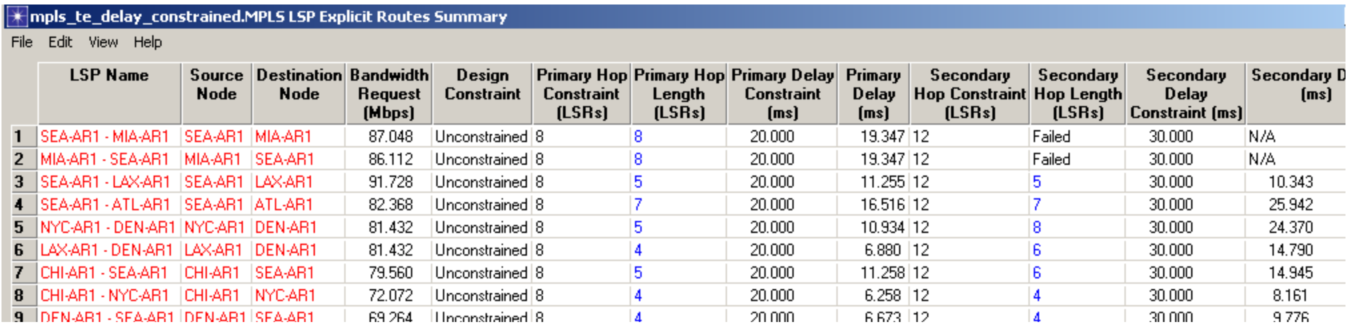
Table 8-4 MPLS Link Subscription Bidirectional Report Results

Result	Description
Link Name Node 1 Node 2	Each row in the report represents a link in the network. Since the information can be different on the link in each direction, each link is shown in two different rows. The Source Node Name and Destination Node Name indicates the direction of the link shown in this row. Links are sorted in descending order of Max Total Subscription.
Min Link TE BW (Mbps)	The lower of the reported values for Link TE Bw in each direction on this link. See Table 8-3 MPLS Link Subscription Report Results on page DA-8-9 for a description of Link TE Bw.
Link Delay (ms)	The total one-way delay over the link.
Max Total Subscription (%)	The higher of the reported values for Total Subscription in each direction on this link. See Table 8-3 MPLS Link Subscription Report Results on page DA-8-9 for a description of Total Subscription.
Max Primary Subscription (%)	The higher of the reported values for Primary Subscription in each direction on this link. See Table 8-3 MPLS Link Subscription Report Results on page DA-8-9 for a description of Primary Subscription. Click on a value in this column to open the Contained LSPs report for a link.
Max Secondary Subscription (%)	The higher of the reported values for Secondary Subscription in each direction on this link. See Table 8-3 MPLS Link Subscription Report Results on page DA-8-9 for a description of Secondary Subscription.
Min Total Residual BW (Mbps)	The lower of the reported values for Total Residual Bw in each direction on this link. See Table 8-3 MPLS Link Subscription Report Results on page DA-8-9 for a description of Total Residual Bw.
Min Primary Residual BW (Mbps)	The lower of the reported values for Primary Residual Bw in each direction on this link. See Table 8-3 MPLS Link Subscription Report Results on page DA-8-9 for a description of Primary Residual Bw.
End of Table 8-4	

MPLS LSP Explicit Routes Summary

This report includes information available in the Design Log summary messages. The results given in the MPLS LSP Explicit Routes Summary report are described in Table 8-5.

Figure 8-6 MPLS LSP Explicit Routes Summary Report



mpls_te_delay_constrained.MPLS LSP Explicit Routes Summary

File Edit View Help

	LSP Name	Source Node	Destination Node	Bandwidth Request (Mbps)	Design Constraint	Primary Hop Constraint (LSRs)	Primary Hop Length (LSRs)	Primary Delay Constraint (ms)	Primary Delay (ms)	Secondary Hop Constraint (LSRs)	Secondary Hop Length (LSRs)	Secondary Delay Constraint (ms)	Secondary D (ms)
1	SEA-AR1 - MIA-AR1	SEA-AR1	MIA-AR1	87.048	Unconstrained	8	8	20.000	19.347	12	Failed	30.000	N/A
2	MIA-AR1 - SEA-AR1	MIA-AR1	SEA-AR1	86.112	Unconstrained	8	8	20.000	19.347	12	Failed	30.000	N/A
3	SEA-AR1 - LAX-AR1	SEA-AR1	LAX-AR1	91.728	Unconstrained	8	5	20.000	11.255	12	5	30.000	10.343
4	SEA-AR1 - ATL-AR1	SEA-AR1	ATL-AR1	82.368	Unconstrained	8	7	20.000	16.516	12	7	30.000	25.942
5	NYC-AR1 - DEN-AR1	NYC-AR1	DEN-AR1	81.432	Unconstrained	8	5	20.000	10.934	12	8	30.000	24.370
6	LAX-AR1 - DEN-AR1	LAX-AR1	DEN-AR1	81.432	Unconstrained	8	4	20.000	6.880	12	6	30.000	14.790
7	CHI-AR1 - SEA-AR1	CHI-AR1	SEA-AR1	79.560	Unconstrained	8	5	20.000	11.258	12	6	30.000	14.945
8	CHI-AR1 - NYC-AR1	CHI-AR1	NYC-AR1	72.072	Unconstrained	8	4	20.000	6.258	12	4	30.000	8.161
9	DEN-AR1 - SEA-AR1	DEN-AR1	SEA-AR1	69.264	Unconstrained	8	4	20.000	6.673	12	4	30.000	9.776

Table 8-5 MPLS LSP Explicit Routes Summary Results (Part 1 of 2)

Result	Description
LSP Name Source Node Destination Node	Each row in the report corresponds to an LSP. The LSPs are unidirectional from a source node to a destination node. The LSPs are sorted based on multiple keys. First all LSPs with a Failed Primary ER are listed. Next all LSPs with a Failed Secondary ER are listed. Successful LSPs are listed last. Among LSPs in the same category, the LSPs are sorted by Bandwidth Request in descending order.
Bandwidth Request (Mbps)	The amount of bandwidth reservation required by the LSP. This is set in the TE Parameters ➤ Min Bandwidth attribute on the LSP.
Design Constraint	This column indicates the value for the LSP's Design Constraint attribute (either "Locked" or "Unconstrained"). Only "Unconstrained" LSPs are modified by the design action.
Primary Hop Constraint (LSRs)	The hop constraint for this LSP's primary route (number of LSRs, including the source LSR). If there is no constraint, the value is "Unlimited".
Primary Hop Length (LSRs)	If the primary explicit route was successfully computed, this column gives the number of hops in the path (number of LSRs, including the source LSR). If the route computation was unsuccessful, the value is "Failed". For locked LSPs, this column displays "Locked" if a valid route is configured and "Empty" if no route is configured or if the configured route is not valid. Click on a value in this column to open the Primary Route report for an LSP.
Primary Delay Constraint (ms)	The delay constraint for this LSP's primary route. If there is no constraint, the value is "Unlimited".

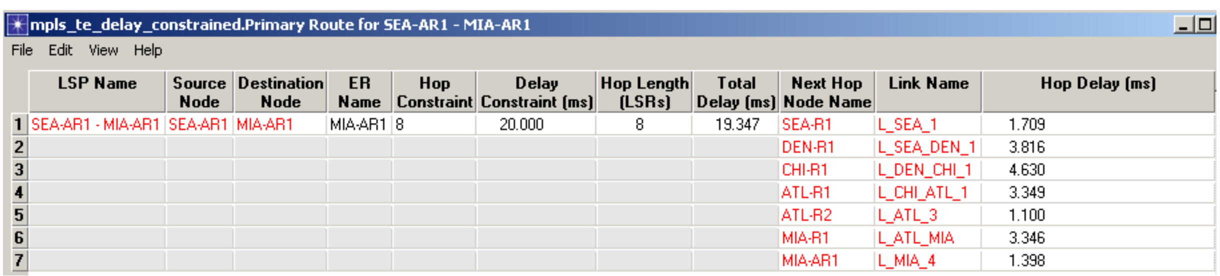
Table 8-5 MPLS LSP Explicit Routes Summary Results (Part 2 of 2)

Result	Description
Primary Delay (ms)	The total route delay for this LSP's primary route.
Secondary Hop Constraint (LSRs)	The hop constraint for this LSP's secondary route (number of LSRs, including the source LSR). If there is no constraint, the value is "Unlimited".
Secondary Hop Length (LSRs)	If the secondary explicit route was successfully computed, this column gives the number of hops in the path (number of LSRs, including the source LSR). If the route computation was unsuccessful, the value is "Failed". For locked LSPs, this column displays "Locked" if a valid route is configured and "Empty" if no route is configured or if the configured route is not valid. Click on a value in this column to open the Secondary Route report for an LSP.
Secondary Delay Constraint (ms)	The delay constraint for this LSP's secondary route. If there is no constraint, the value is "Unlimited".
Secondary Delay (ms)	The total route delay for this LSP's secondary route.
End of Table 8-5	

Primary Route

This report includes information about a primary route. It is a drilldown accessed by clicking in the Secondary Hop Length column of the Explicit Routes Summary. The results given in the Primary Route report are described in Table 8-6.

Figure 8-7 Primary Route Report



	LSP Name	Source Node	Destination Node	ER Name	Hop Constraint	Delay Constraint (ms)	Hop Length (LSRs)	Total Delay (ms)	Next Hop Node Name	Link Name	Hop Delay (ms)
1	SEA-AR1 - MIA-AR1	SEA-AR1	MIA-AR1	MIA-AR1	8	20.000	8	19.347	SEA-R1	L_SEA_1	1.709
2									DEN-R1	L_SEA_DEN_1	3.816
3									CHI-R1	L_DEN_CHI_1	4.630
4									ATL-R1	L_CHI_ATL_1	3.349
5									ATL-R2	L_ATL_3	1.100
6									MIA-R1	L_ATL_MIA	3.346
7									MIA-AR1	L_MIA_4	1.398

Table 8-6 Primary Route Results

Result	Description
LSP Name Source Node Destination Node	The name, source node, and destination node of the LSP. Click on a name to jump to the object in the Project Editor.
ER Name	The name of the explicit route that is used by the LSP for this route. The ER Name matches the value referenced in the LSP ➘ Configured Paths attribute and defined in the source node's MPLS Parameters ➘ Explicit Routes attribute.
Hop Constraint (LSRs)	The constraint on the maximum number of hops that can be used by the LSP (number of LSRs, including the source LSR).
Hop Length (LSRs)	The number of hops in the actual route (number of LSRs, including the source LSR).
Delay Constraint (ms)	The constraint on the maximum delay in a path used by the LSP, in milliseconds.
Total Delay (ms)	The total delay (in milliseconds) for the path specified in the report. This is the sum of the Hop Delay column over all of the hops.
Next Hop Node Name	The name of the next hop node. This column contains one row for each hop in the explicit route. Click on a node name to jump to that object in the Project Editor.
Link Name	The name of the next hop link. This column contains one row for each hop in the explicit route. Click on a link name to jump to that object in the Project Editor.
Hop Delay (ms)	The delay across the individual hop, in milliseconds. This column contains one row for each hop in the explicit route.
End of Table 8-6	

Secondary Route

This report contains information about a secondary route. It is a drilldown accessed by clicking in the Secondary Hop Length column of the Explicit Routes Summary. The results are the same as those for the Primary Route report, as described in Table 8-6.

Contained LSPs

For each direction of the link, this report gives the set of LSPs routed over the link. It is a drilldown accessed by clicking in the Total Subscription column of the MPLS Link Subscription or MPLS Link Subscription Bidirectional report. The results given in the Contained LSPs report are described in Table 8-7.

Figure 8-8 Contained LSPs Report

	Link Name	Source Node	Destination Node	Link TE BW (Mbps)	Total Subscription (%)	Primary Subscription (%)	Primary LSPs	TE BW Request (Mbps)	Secondary Subscription (%)	Secondary LSPs	TE BW Request (Mbps)
1	_DEN_5	DEN-AR2	DEN-R1	8,845.1482	0.307	0.243	DEN-AR2 - NYC-AR2	8.4240	0.063	DEN-AR2 - MIA-AR2	5.6160
2							DEN-AR2 - LAX-AR2	6.5520			
3							DEN-AR2 - SEA-AR2	5.6160			
4							DEN-AR2 - ATL-AR2	0.9360			
5							DEN-AR2 - CHI-AR2	0.0000			
6	_DEN_5	DEN-R1	DEN-AR2	8,845.148	0.519	0.074	ATL-AR2 - DEN-AR2	3.744	0.444	LAX-AR1 - SEA-AR1	39.312
7							NYC-AR2 - DEN-AR2	2.808		CHI-AR1 - DEN-AR1	6.552
8										CHI-AR2 - DEN-AR2	5.616
9										SEA-AR2 - DEN-AR2	2.808
10										LAX-AR2 - DEN-AR2	1.872
11										MIA-AR2 - DEN-AR2	0.936

Table 8-7 Contained LSPs Results (Part 1 of 2)

Result	Description
Link Name Source Node Destination Node	The name, source node, and destination node of the link. Click on a name to jump to the object in the Project Editor.
Link TE BW (Mbps)	Total Bandwidth on the link in this direction available for traffic engineering. This is specified by the RSVP TE Parameters ➘ Interface Information ➘ Maximum Reservable Bandwidth attribute.
Total Subscription (%)	The total subscription (percentage with respect to the Link TE BW) on this link in this direction. This is the sum of the Primary Subscription + Secondary Subscription.
Primary Subscription (%)	The total subscription based on the primary LSP routing over the link in this direction. This is the sum of the bandwidth requests (where each request is 100 * LSP BW Request / Link TE BW).
Primary LSPs	List of LSPs whose primary route uses this link in this direction. This column contains one row for each LSP using this link in its primary route. Click on an LSP name to jump to that object in the Project Editor.
TE BW Request (Mbps)	The corresponding bandwidth request for each LSP. This column contains one row for each LSP using this link in its primary route. Click on an LSP name to jump to that object in the Project Editor.

Table 8-7 Contained LSPs Results (Part 2 of 2)

Result	Description
Secondary Subscription (%)	The amount of bandwidth used by LSPs routed over the link with their secondary routes for the worst-case failure scenario (link, node, SRG, and so on, based on the Primary<->Secondary ER Relationship setting). Because the LSPs might share capacity with other LSP secondary routes, it is possible that this value is not the same as the sum of the BW request of the secondary LSPs.
Secondary LSPs	List of LSPs whose secondary route uses this link in this direction. This column contains one row for each LSP using this link in its secondary route. Click on an LSP name to jump to that object in the Project Editor.
TE BW Request (Mbps)	The corresponding bandwidth request for each LSP. This column contains one row for each LSP using this link in its secondary route. Click on an LSP name to jump to that object in the Project Editor.
End of Table 8-7	

9 MPLS Differentiated-Service-Aware Traffic Engineering

The mpls_ds_te design action is an enhanced version of mpls_te. In mpls_ds_te, the LSP bandwidth request is specified with a bandwidth class type. Each interface defines the amount of bandwidth available for each class type. The node defines a bandwidth model that controls the rules for allocating bandwidth to each class type. The mpls_ds_te design action determines primary and (optionally) secondary explicit routes that conform to the constraints specified by the relevant class type.

For more information about this design action, open the Configure/Run Design Action dialog box (choose Design > Configure/Run Design Action in the Project Editor).

- For a complete description, select the design action and click View Comments.
- For information about individual attributes, click Edit Attributes. Then click the help (“question-mark”) button next to the attribute of interest.

10 MPLS Explicit Route Configlet Generation

You can use the mpls_er_configlet_generation design action to generate "configlet" files that contain MPLS TE explicit route information. Configlet files contain commands for configuring explicit routes; these commands are identical to those an administrator can enter manually on a Cisco or Juniper router. This design action creates one configlet file for every LSR that meets each of the following conditions:

- The LSR vendor is either Cisco Systems or Juniper Networks (as specified by the Vendor field in the self-description of the underlying node model).
- The LSR is a source for least one exported LSP.
- The exported LSPs sourced at the LSR have at least one valid explicit route.

Workflow Description

You can use this design action and other features to configure explicit routes in your production network. The following steps outline this workflow:

1) Generate device configuration files that include LSP and explicit-route information in your production network.
2) Generate a network model by importing the device configuration data (either import the files directly or import from VNE Server).
3) Validate the network model using Flow Analysis and NetDoctor to confirm that the network configurations are correct and behaving as expected.
4) Run the MPLS Traffic Engineering design action to compute near-optimal and survivable primary and secondary explicit routes.
5) Verify and validate the explicit routes using Flow Analysis before you export them.
6) Run the MPLS Explicit Route Configlet Export design action to generate configlet files that define the generated explicit routes.
7) Transfer the configlet files to routers in your production network and have the router process the commands in the configlet file. (For information about how to load configuration files on individual routers, see the documentation supplied by the device vendor.)

Prerequisites

This section describes what you need to know before you run the MPLS Explicit Route Configlet Generation design action.

Background Knowledge

To use this design action most effectively, you must be familiar with the following topics:

- OPNET node, link, and LSP path models
- MPLS and OPNET's MPLS model
- MPLS Traffic Engineering

Design Action and Attribute Descriptions

In addition to the information in this chapter, you should also view the action description and individual attribute descriptions available in the Configure/Run Design Action dialog box. For more information, see Viewing Information about a Design Action on page DA-1-4.

For general information about configuring and running a design action, see Running a Design Action on page DA-1-8. For detailed information about the attributes of this design action, see the online description for each attribute. In the Configure/Run Design Action dialog box, click Edit Attributes; then click on the question mark next to the attribute of interest.

Input Network Scenario: Requirements

For this design action to generate a configlet file, a label switch router (LSR) nodes must have its MPLS Parameters ➘ Explicit Routes attribute set. Each explicit route has a name and a complex attribute (Path Details) that specifies the hop-by-hop LSP route. An LSP refers to these named routes on its source router, and has a Configured Paths attribute with a list of named explicit routes. The lowest-priority configured path is the primary explicit route.

Generated Configlet Files

The MPLS Explicit Route Configuration Export design action can generate files based on Cisco and Juniper configuration file formats. You can also choose to choose to export MPLS commands related to bandwidth requests and LSP destination addresses as well. To specify the information you want to export, set the Commands attribute.

The design action places these files in the directory specified by its Configlet Directory attribute. By default, the design action names each explicit-route file based on the LSR node name and vendor using the convention <*LSR_node_name*>_er.<*vendor_name*> (for example, router_A_er.cisco). The default configlet directory is <*opnet_user_dir*>/op_admin/tmp/Configlets.

Cisco Configlet Files

Cisco configlet files are based on the Cisco IOS 12.3 configuration file format, as shown in Figure 10-1.

Figure 10-1   Cisco Configlet File: Example

```
! MPLS Explicit Route Configlet Generated by SP Guru
! Device: DCI Network.sink
! ID: 10.10.10.5
!

! LSP:  sink - source
interface Tunnel0
 tunnel mpls traffic-eng path-option 1 explicit name sink-B-source
!
!
ip explicit-path name sink-B-source enable
 next-address 10.1.1.17
 next-address 10.1.1.5
```

Juniper Configlet Files

Juniper configlet files are based on the JUNOS 6 (6.0 - 6.4) file format, as shown in Figure 10-2.

Figure 10-2 Juniper Configlet File: Example

```
/*
MPLS Explicit Route Configlet Generated by SP Guru
Device: DCI Network.juniper_lsr
ID: 10.10.10.5
*/

groups {
  mpls-lsp-group {
    protocols {
      mpls {
# LSP:  juniper_lsr - cisco_lsr
        label-switched-path "juniper_lsr - cisco_lsr" {
          primary juniper_lsr-B-cisco_lsr {
        }
      }
    }
  }
}
mpls-er-group {
  protocols {
    mpls {
      path juniper_lsr-B-cisco_lsr {
          10.1.1.17 strict;
          10.1.1.5 strict;
        }
      }
    }
  }
}
apply-groups [ mpls-er-group mpls-lsp-group ];
```

Network Changes

This design action makes no changes to the input scenario.

Log Messages

To open the Design Action log, choose Design > Results > View Latest Log. This log is the first place to determine whether the design action ran successfully. It is good practice to view the Summary Statistics log message immediately after you run a design action. This message shows information about the resulting design, such as the number of configlets that were successfully generated.

Table 10-1 Categories of MPLS Explicit Route Configlet Export Log Messages

Category	Description
Design Manager	High-level information about the last run
Warning	Warning messages generated during the run
Detail	Information about each configlet generated
Summary	A list of statistics that were generated before and after the run
End of Table 10-1	

Table 10-2 Classes of MPLS Explicit Route Configlet Export Log Messages

Class	Description
Scenario	Input scenario for the design action
Model Errors	Messages generated because of errors in the network model
Write – Success	Information about configlets that were successfully generated by the design action
Summary	A list of statistics that were generated before and after the run
Success	Status of the last design-action run
End of Table 10-2	

Available Reports

This design action does not generate any reports or output tables.

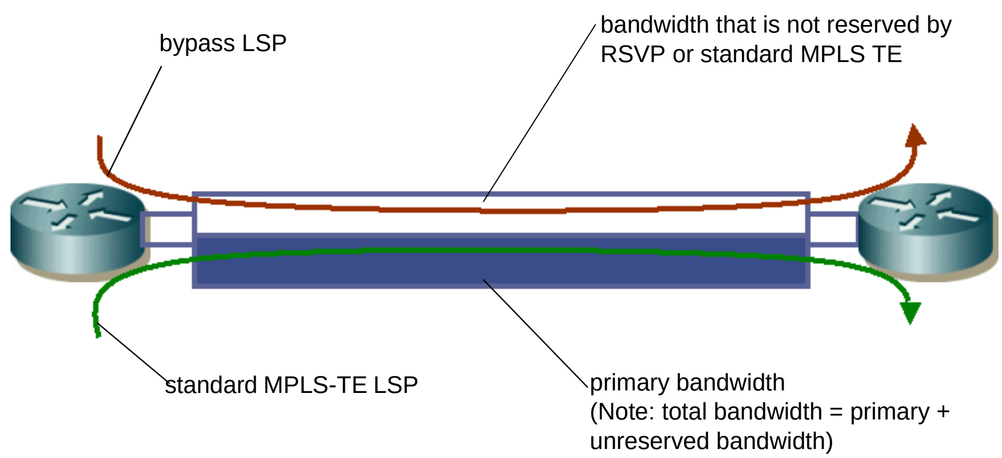
11 MPLS Minimum Fast Reroute Design

You can use the mpls_frr_bypass_lsp_routing design action to optimize the set of routes for existing bypass tunnels, where optimality is evaluated with respect to a specified measure of bandwidth usage. This design action assigns routes to bypass tunnels to ensure that, in any single-failure scenario, every set of bypass tunnels consumes no more than a specified fraction of the *unreserved* bandwidth on each link.

Primary and Bypass LSPs

If no network failures interrupt the path of a standard MPLS-TE LSP, the LSP routes traffic using primary bandwidth. *Primary bandwidth* is reserved by RSVP and does not exceed the maximum bandwidth for standard MPLS-TE. (For this reason, an LSP that uses primary bandwidth is also called a *primary LSP*.) If a network element fails and makes a node or link in the primary LSP path unusable, a *bypass LSP* routes traffic across a backup path around the failed element. The backup path uses “unreserved” bandwidth that exceeds the primary bandwidth.

Figure 11-1 Primary and Bypass LSPs



SP Guru also includes the MPLS Traffic Engineering design actions that enable you to assign routes to MPLS Label Switch Paths (LSPs) in an MPLS network:

- The mpls_te design action computes near-optimal primary and secondary routes.
- The mpls_lsp_bypass_routing design action computes Fast-Reroute backup paths for Bypass LSPs that are set up to provide protection for link, node, and facility failures. A bypass LSP provides a "bypass" route used in place of a hop or a two-hop subroute in an end-to-end primary or secondary route.

Note—The mpls_frr_bypass_lsp_routing design action does not create bypass tunnels automatically. The network scenario must have FRR Bypass Tunnels specified, and link capacity reserved for traffic engineering, before you run the design action. For more information, see Input Network Scenario: Requirements on page DA-11-3.

Prerequisites

This section describes what you need to know before you run the mpls_frr_bypass_lsp_routing design action.

Background Knowledge

To use this design action most effectively, you should be familiar with the following topics:

- OPNET node, link, and LSP path models
- MPLS and OPNET's MPLS model, including Cisco-style Bypass Tunnels
- MPLS Traffic Engineering

Design Action and Attribute Descriptions

In addition to the information in this chapter, you should also view the action description and individual attribute descriptions available in the Configure/Run Design Action dialog box. For more information, see Viewing Information about a Design Action on page DA-1-4.

For general information about configuring and running a design action, see Running a Design Action on page DA-1-8. For detailed information about the attributes of this design action, see the online description for each attribute. In the Configure/Run Design Action dialog box, click Edit Attributes; then click on the question mark next to the attribute of interest.

Input Network Scenario: Requirements

This design action requires that the following be configured in the input network scenario:

1) For each protected interface on a MPLS label-switched router (LSR) or label-edge router (LER) node, the following attribute must be set for each Bypass LSP whose source is at that interface:

   MPLS Parameters ➤ Interface Information [<*interface_number*>] ➤ Bypass Tunnel Configuration.[<*tunnel_number*>] ➤ LSP Name

   If the interface is protected under a next-hop (NHOP) protection scheme, there must be one or more Bypass LSPs that terminate at the destination node of the link attached to that interface.

   If the interface is protected under a next-next-hop (NNHOP) protection scheme, there must be one or more Bypass LSPs for each interface reachable in *exactly* two hops from the protected interface. This requirement ensures protection against failure of the next-hop router (that is, the router at the other end of the protected interface).

2) The following attribute must be set on every node:

   RSVP Protocol Parameters ➘ Interface Information[<*interface_number*>] ➘ Maximum Reservable BW

   Guru uses this attribute in two ways. It determines the link bandwidth available for bypass routing based on the following formula:

   <*bandwidth_available_for_bypass_routing*> =
   <*max_subscription_attribute_value*> *
   (<*data_rate_of_outgoing_link*> – <*maximum_reservable_bandwidth*>)

   The <*max_subscription_attribute_value*> is specified by the ER Computation ➘ Max Link Subscription % attribute of the mpls_frr_bypass_lsp_routing design action. This attribute specifies the maximum percentage of a link's total bypass bandwidth that may be reserved for bypass LSPs. As shown in the previous equation, the link's total bypass bandwidth equals the data rate minus the maximum reservable bandwidth.

   Guru also uses the Maximum Reservable BW attribute in two ways:

   a) To determine "Backup Bandwidth Protection" for bypass LSPs that protect an interface

   b) To ensure that this bandwidth is completely protected by its Bypass LSPs

      The maximum reservable bandwidth of a link interface must be less than or equal to the total backup bandwidth of the bypass LSPs that protect that interface. If this requirement is not met, and the protected interface fails, the Bypass LSPs might be unable to reroute all the impacted traffic.

      The maximum reservable bandwidth is specified by the Maximum Reservable BW attribute on the link. The LSP backup bandwidth is specified by the "Backup Bandwidth Protection ➘ Backup Bandwidth (bps)" attribute on the LSPs.

3) The network must contain a set of bypass LSPs (named in the interface attributes described previously); on each LSP, the Backup Bandwidth Protection attribute must be configured to one of the following:

   — The amount of TE-reservable bandwidth on the source interface, or

   — A user-specified alternate value.

   Keep in mind that this attribute is different from the (non-Bypass) TE bandwidth requirement of the LSP. Any LSP whose "Design Constraint" attribute is set to "Locked" will NOT be modified by the design action.

4) The following attribute must be set to Enabled for all connected interfaces on LSRs:

   MPLS Parameters ➘ Interface Information[<*interface_number*>] ➘ MPLS Status

Network Changes

This design action assigns bypass routes to LSPs and configures the bypass routes using the Configured Paths attribute (on LSPs) and the MPLS Parameters.Explicit Routes attribute (on LSRs). As a result, this design action replaces existing Configured Paths and appends new Explicit Routes to existing Explicit Routes.

Log Messages

To open the Design Action log, choose Design > Results > View Latest Log. This log is the first place to determine whether the design action ran successfully. It is good practice to view the Summary Statistics log message immediately after you run a design action. This message shows information about the resulting design, such as the link subscription and the number of routed LSPs.

Table 11-1 Categories of MPLS Fast Reroute Design Log Messages

Category	Description
Design Manager	High-level information about the action run
Detail	Information about each LSP processed
Network Changes	A list of changes to nodes and LSPs
Error	Error messages generated during the run.
Warning	Warning messages generated during the run.
Summary	A list of statistics that were generated before and after the run
Success	Status of the last run (Success or Failure)
End of Table 11-1	

Table 11-2 Classes of MPLS Fast Reroute Design Log Messages (Part 1 of 2)

Class	Description
Scenario	Input scenario for the design action
Direct Routes – Filtered Objects	Objects that were excluded from the design. These are usually configuration objects or nodes that do not have the necessary MPLS configurations (as specified in Input Network Scenario: Requirements on page DA-11-3).

Table 11-2 Classes of MPLS Fast Reroute Design Log Messages (Part 2 of 2)

Class	Description
Bypass Route – Summary	Summary statistics about the bypass routes (number of LSPs routed/unrouted, Total LSP bandwidth, and so on)
Bypass Route – Options Summary	Summary of options used to create the bypass routes, based on the attribute settings of the last run
Success	Status of the most recent run (Success or Failure)
End of Table 11-2	

Available Reports

You can view detailed reports using the Results Viewer, as described in Viewing Results on page DA-1-10. Table 11-3 lists the available reports generated by the MPLS Fast Reroute design action.

Table 11-3 Reports Generated by MPLS Fast Reroute Design Action

Report	Reference
Global reports:	
• MPLS Link Subscription	MPLS Link Subscription Report
• MPLS Link Subscription Bidirectional	MPLS Link Subscription Bidirectional Report on page DA-11-8
• MPLS LSP Explicit Routes Summary	LSP Explicit Routes Summary Report on page DA-11-9
Link Object Report:	
• Contained LSPs	Contained LSPs Report on page DA-11-9
LSP Object Reports:	
• Bypass Route	Bypass Route/Direct Route Reports on page DA-11-10
• Direct Route	Bypass Route/Direct Route Reports on page DA-11-10
End of Table 11-3	

MPLS Link Subscription Report

The MPLS Link Subscription report shows information about the links used in the topology design. The links are sorted in descending order of bandwidth.

Table 11-4 MPLS Link Subscription Report (Part 1 of 2)

Field	Description
Link Name	Name of the link
Source Node	Name of the source node of the link
Destination Node	Name of the destination node of the link
FRR BW (Mbps)	Bandwidth available for routing Bypass demands

Table 11-4 MPLS Link Subscription Report (Part 2 of 2)

Field	Description
Link Delay (ms)	Transmission delay across the link
FRR Subscription (%)	Total subscription of the link in the direction from Source to Destination. This field is a drilldown to the Contained LSPs report that shows the LSPs routed over the link.
FRR Residual (Mbps)	FRR total bandwidth minus FRR subscribed bandwidth
End of Table 11-4	

MPLS Link Subscription Bidirectional Report

This report shows information about the links used in the topology design. The links are sorted in descending order of bandwidth.

Table 11-5 MPLS Link Subscription Bidirectional Report

Field	Description
Link Name	Name of the link
Node 1	Name of the first end node of the link
Node 2	Name of the second end node of the link
FRR BW (Mbps)	Bandwidth available for routing Bypass demands
Min Link Delay (ms)	Transmission delay across the link, which is the minimum of the two directional values
Max FRR Subscription (%)	The total link subscription from Source to Destination, which is the maximum of the two directional values. This field contains a drilldown to the Contained LSPs report that shows the LSPs routed over the link.
Min FRR Residual (Mbps)	The difference between FRR bandwidth and FRR subscribed bandwidth, which is the minimum of the two directional values.
End of Table 11-5	

LSP Explicit Routes Summary Report

This report is a link object table that is accessed as a drill-down from the Link Summary report. This report lists each LSP that is routed over the link in each direction.

Table 11-6 LSP Explicit Routes Summary Report

Field	Description
LSP Name	Name of the LSP whose route information is being reported
Source Node	Source node of the LSP
Destination Node	Destination node of the LSP
Bandwidth Request (Mbps)	Bandwidth reserved for the LSP along its path (based on the Backup Bandwidth Protection ➘ Backup Bandwidth attribute of the LSP)
Design Constraint	Whether the bypass was Locked (fixed before the design action ran) or Unconstrained (updated by the design action)
ER Name	Name of the explicit route used by the LSP. The explicit route is defined in the MPLS Parameters ➘ Explicit Routes attribute of the source LSP
Protected Hop Length (LSRs)	Number of LSRs in the protected hop or sub-route (two for NHOP or 3 for NNHOP). This field is a drilldown to the Primary Route detail table.
Bypass Hop Length (LSRs)	Number of LSRs in the bypass sub-route (two for NHOP or 3 for NNHOP). This field is a drilldown to the Bypass Route detail table.
Protected Delay (ms)	Propagation delay of the protected hop or sub-route
Bypass Delay (ms)	Propagation delay of the bypass route
Delay Increase (ms)	Difference between the bypass delay and protected delay
End of Table 11-6	

Contained LSPs Report

This report is a link object table that is accessed as a drill-down from the Link Summary report. This report lists each LSP that is routed over the link in each direction.

Table 11-7 Contained LSPs Report (Part 1 of 2)

Field	Description
Link Name	Name of the link whose contained LSPs are being listed
Source Node	Source node of the link

Table 11-7 Contained LSPs Report (Part 2 of 2)

Field	Description
Destination Node	Destination node of the link
Bypass LSPs	List of the LSPs routed over the link in the specified direction
Backup BW (Mbps)	Amount of bandwidth protected by this Bypass LSP in Mbps. This value is based on the Backup Bandwidth Configuration attribute setting of the link
End of Table 11-7	

Bypass Route/Direct Route Reports

Each of the Bypass Route and Direct Route tables is an object table available for each routed LSP. You can access these reports either directly through the Object Tables section of the Results Viewer or as drilldowns from the Contained LSPs or LSP Summary reports. The table includes several rows for each LSP; each row corresponds to a link traversed by the route.

Table 11-8 Bypass Route/Direct Route Report

Field	Description
LSP Name	Name of the LSP whose route information is being reported
Source Node	Source node of the LSP
Destination Node	Destination node of the LSP
Hop Constraint	(*Bypass Route only*) User-specified limit on the total number of hops (LSRs) in the bypass route
Delay Constraint	(*Bypass Route only*) User-specified limit on the total end-to-end propagation delay of the Bypass Route
Hop Length (LSRs)	Number of LSRs in the explicit route, including the source and the destination
Total Delay (ms)	End-to-end propagation delay along the route
Next Hop Node Name	Each hop in the path (except the source LSR) appears in the Next Hop Node Name and Link Name column. The Next Hop Node Name is the next LSR in the path. The last entry in this column is the destination node.
Link Name	The name of the link used to get from the previous node to the current node (specified in the Next Hop Node Name column for the same row)
Hop Delay	Propagation delay across the link
End of Table 11-8	

12 MPLS Minimum Cost Topology Design

The design actions for topology design help you determine where to place links in a network. Different design actions are available, depending on the type of network you are building.

The min_cost_mpls_net_design design action is used to determine the near-minimum cost network that supports a specified set of MPLS Label Switched Paths (LSPs). For MPLS networks, you can use this design action to place the near-minimum cost set of links in the network needed to support the specified LSPs.

This design action is intended for Greenfield designs; it simultaneously determines link placement, link sizing and LSP routing (in contrast to the other topology design actions, which focus primarily on link placement). This design action minimizes the cost of building a network to support the primary routing of the specified LSPs. Resiliency and connectivity are not directly addressed.

Prerequisites

This section describes what you need to know before you run the min_cost_mpls_net_design action.

Background Knowledge

To use this design action most effectively, you should be familiar with the following topics:

- Greenfield topology design
- Routers, ports and their supported links
- OPNET node and link models
- MPLS and OPNET's MPLS model

Design Action and Attribute Descriptions

In addition to the information in this chapter, you should also view the action description and individual attribute descriptions available in the Configure/Run Design Action dialog box. For more information, see Viewing Information about a Design Action on page DA-1-4.

For general information about running or configuring a design action, see Running a Design Action on page DA-1-8. For detailed information about the attributes of this design action, see the online description for each attribute (in the Configure/Run Design Action dialog box, click Edit Attributes; then click on the question mark next to the attribute of interest).

Configuring and Running the Minimum-Cost MPLS Network Design Action

The min_cost_mpls_net_design action constructs a topology design by routing LSPs one-by-one over the working network:

- Initially, the network does not contain any links and the LSPs are all unrouted.
- To route the first LSP, the design action finds the minimum-cost set of links that can be added to the network to route the LSP from its source to its destination.
- To route the next LSP, the design action finds the minimum-cost augmentation of the network that supports the LSP. It considers links to be free if they have been added already and have available capacity. It can upgrade existing links to larger links as more LSPs are routed between the same node pair.
- This process continues until all LSPs are routed.
- After all LSPs are routed, the design action can run subsequent iterations (depending on the attribute settings) to refine and improve the initial solution.

The quality of the design is sensitive to the order in which the LSPs are routed. Several action attributes influence the order in which the LSPs are routed and the number of orders that are examined.

Log Messages

To open the Design Action log, choose Design > Results > View Latest Log. This log is the first place to determine whether the design action ran successfully. It is good practice to view the Summary Statistics log message immediately after you run a design action. This message shows information about the resulting design, such as the total cost and the number of routed LSPs.

The min_cost_mpls_net_design design action generates several types of log messages to show the summary and network changes. Table 12-1 and Table 12-2 list these log messages.

Table 12-1 Categories of min_cost_mpls_net_design Log Messages

Category	Description
Detail	Candidate link models considered when adding or replacing individual links
Design Manager	A list of subactions that ran successfully
Network Changes	A list of the links that were created
Error	Error messages generated during the run
Warning	Warning messages generated during the run
Summary	A list of statistics that were generated before and after the run. Statistics include: average cost per link, cost of new links, and the number of links added and reused.
End of Table 12-1	

Table 12-2 Classes of min_cost_mpls_net_design Log Messages

Use This Class...	To View...
Link Pool	Candidate link models considered when adding links
Filtered	Nodes that were excluded from the design. These are usually configuration objects or nodes that do not have the necessary MPLS configurations.
Created	Links that were created by the design action
Summary	Summary statistics about the design action (for more information, see Summary Report on page DA-12-4)
Success	Whether or not the design action ran successfully
End of Table 12-2	

Available Reports

You can view detailed reports using the Results Viewer, as described in Viewing Results on page DA-1-10. Table 12-3 lists the available reports generated by the min_cost_mpls_net_design action.

Table 12-3 Reports Generated by min_cost_mpls_net_design Action

Report	Reference
Global reports:	
• Summary	Summary Report on page DA-12-4
• LSP Summary	LSP Summary Report on page DA-12-6
• Link Summary	Link Summary Report on page DA-12-6
Object Reports	
• Contained LSPs	Contained LSPs Report on page DA-12-7
• Explicit Routes	Explicit Routes Report on page DA-12-8
End of Table 12-3	

Summary Report

You can read the Summary report statistics to evaluate the overall quality of the design and to determine whether the design achieved your design goals. These statistics are also shown in the Summary Statistics log message.

Table 12-4 Summary Report (Part 1 of 2)

Field	Description
Vertices in Graph	The number of vertices in the design graph. Each vertex corresponds to an LSR in the specified input node set.
Links Added	The number of links added by the design action
Cost of New Links	The total cost of the new links
Average Cost Per Link	The average cost of the links
Min Link Subscription (%)	The maximum link subscription over all links. **NOTE—**For this and the next two results, the subscription of a link is considered to be the maximum subscription in both directions.
Average Link Subscription (%)	The average link subscription over all links in the design graph. For this average, each link is weighted equally regardless of its bandwidth.
Total Link Bandwidth (Mbps)	The sum of the bandwidth capacity of the links in the routing graph. A link's bandwidth is added for each direction.

Table 12-4 Summary Report (Part 2 of 2)

Field	Description
Link Bandwidth Efficiency	The total subscribed bandwidth on all links (both directions) divided by the total link bandwidth (both directions). This is the bandwidth-weighted average link subscription value.
Graph Diameter	The maximum number of hops in the shortest-hop path between any node pair. The number of hops in the path is the number of nodes in the path (excluding the source node).
Graph Average Distance	The average number of hops in the shortest hop path over all of the node pairs (excluding source to source pairs).
Traffic Averaged Distance	The traffic-averaged number of hops between any node pair in the graph. For each pair, the current length of the minimum-hop-distance route is weighted by the total LSP bandwidth between the two nodes.
Minimum Graph Connectivity	The connectivity between two nodes is the number of node-disjoint paths between them in the design graph. This measure is the minimum connectivity over all node pairs.
Average Graph Connectivity	The average connectivity over all node pairs in the graph.
Traffic Averaged Connectivity	The traffic-weighted average connectivity over all node pairs in the design graph. For every two nodes, the connectivity is weighted by the total amount of LSP-required bandwidth between the nodes.
Number of Routed LSPs	The number of LSPs whose explicit route was successfully computed and assigned by the design action.
Number of Unrouted LSPs	The number of LSPs whose explicit route could not be computed or assigned by the design action.
Bandwidth of Routed LSPs (bps)	The total bandwidth request of all of the routed LSPs (specified by the TE Parameters.Min Bandwidth attribute on each LSP).
Bandwidth of Unrouted LSPs (bps)	The total bandwidth request of all of the unrouted LSPs (specified by the TE Parameters.Min Bandwidth attribute on each LSP).
Best Random Seed	The random seed used at the start of the random case that provided the best solution. To re-create the best solution, enter this value in the "Initial Random Seed" design action attribute and run the design action for one random case.
Number of Iterations	The number of iterations required in the best random case to achieve the best solution.
End of Table 12-4	

LSP Summary Report

The LSP Summary report provides top-level route information for all LSPs. Unrouted LSPs are listed first. The TE Bandwidth Request is the secondary sorting key (largest to smallest).

Table 12-5 LSP Summary Report

Field	Description
LSP Name	The name of the LSP whose route information is being reported
Source Node	The source node of the LSP
Destination Node	The destination node of the LSP
TE Bandwidth Request (Mbps)	The bandwidth reserved for the LSP along its path (based on the configuration of its TE Parameters.Min Bandwidth attribute value)
ER Name	The name of the explicit route used by the LSP. The explicit route is defined in the MPLS Parameters.Explicit Routes attribute table of the source LSR.
ER Length (LSRs)	The number of LSRs in the explicit route, including the source and destination. This value is a drilldown to the Explicit Route report.
End of Table 12-5	

Link Summary Report

The link summary shows information about the links used in the topology design. The links are sorted in descending order of bandwidth.

Table 12-6 Link Summary Report Statistics (Part 1 of 2)

Result	Description
Link Name	The name of the link that was added, reused, or upgraded
Node A, Node B	The endpoints of the link
Operation	The operation performed taken on the link by the design action
Model	The link model for the link
Bandwidth	Total capacity of the link, in bits per second (bps)
Cost	The total cost of the link, based on the "financial cost" attribute of the link set by the Link Pricer subaction.
Subscription (A–>B%)	The total subscription of the link in the direction from Node A to Node B

Table 12-6 Link Summary Report Statistics (Part 2 of 2)

Result	Description
Subscription (B–>A%)	The total subscription of the link in the direction from Node B to Node A
Max Subscription (%)	The maximum of the A–>B and B–>A subscription
Min Cost per Mbps	The minimum cost per Mbps of subscribed bandwidth over the A–>B and B–>A directions of the link
Contained LSPs	A drilldown to the Contained LSPs report that shows the LSPs routed over this link
End of Table 12-6	

Contained LSPs Report

The Contained LSPs report is a link object table that you can access as a drill-down from the Link Summary report. This report lists each LSP that is routed over a link in each direction.

Table 12-7 Contained LSPs Report

Result	Description
Link Name	The name of the link whose contained LSPs are being listed
Link Source Node	The source node of this link for this direction
Link Destination Node	The destination node of this link for this direction
LSP Name	The name of each LSP routed over the link in this direction
TE Bandwidth Request	The amount of bandwidth reserved by this LSP, in Mbps (based on the TE Parameters.Min Bandwidth attribute)
LSP ER Name	The name of the explicit route used by the LSP. The name corresponds to a defined route configured in the MPLS Parameters.Explicit Routes attribute table of the source LSR.
ER Length (LSRs)	The number of LSRs in the explicit route (including the source and destination LSR). This value is a drilldown to the Explicit Route report.
End of Table 12-7	

Explicit Routes Report

The Explicit Routes table is an object table available for each routed LSP. It may be accessed directly through the Object Tables section of the Results Viewer or as a drilldown from the Contained LSPs or LSP Summary reports.

Table 12-8 Explicit Routes Report Statistics

Result	Description
LSP Name	The name of the LSP whose explicit route is being reported
Source Node	The source node of the LSP
Destination Node	The destination node of the LSP
TE Bandwidth Request (Mbps)	The bandwidth reserved for the LSP along its path (based on the configuration of its TE Parameters.Min Bandwidth attribute value)
ER Name	The name of the explicit route used by the LSP. The explicit route is defined in the MPLS Parameters.Explicit Routes attribute table of the source LSR.
ER Length (LSRs)	The number of LSRs in the explicit route including the source and destination.
Next Hop Node Name	The Next Hop Node Name and Link Name columns list every hop in the path (except the source LSR). The Next Hop Node Name is the next LSR in the path. The last entry in this column is the destination node.
Link Name	The name of the link used to get from the previous node to the node specified in the Next Hop Node Name column for the same row.
End of Table 12-8	

13 Project Pricing Reporter

You can use the project_pricing_reporter to generate detailed cost reports for proposed upgrades, build-outs, and other project plans. You can create highly detailed cost models, then apply a cost model to the links and nodes in a scenario. The project_pricing_reporter design action then generates a detailed report based on the cost model and the current network.

Workflow Description

The following steps outline the basic workflow for creating a cost report:

1) From the "Pricing" palette, place the "pricing preferences" utility into the project space. For simplicity, this node will be referred to as the “pricing preference object.”

2) Configure the attributes of the pricing preference object (see Configuring the Pricing Preference Object on page DA-13-2).

3) Create nodes and links from the Pricing object palette. These models contain the attributes needed to apply the cost parameters specified in the pricing preference object.

4) Configure the cost attributes of the relevant nodes and links (see Specifying Pricing Information on Nodes and Links on page DA-13-6).

   NOTE—You must configure the pricing preference object (step 2) before you can perform this step, because the node/link cost attributes depend on settings in the pricing preference object.

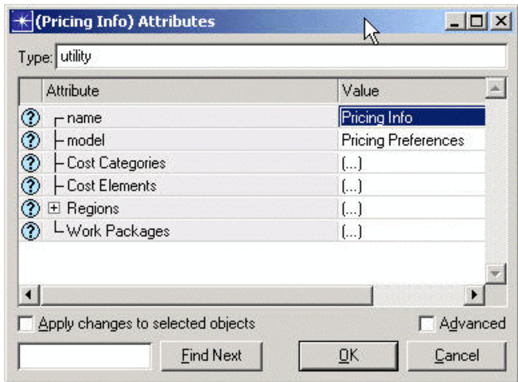
5) Configure and run the project_pricing_reporter design action (see Running the project_pricing_reporter Design Action on page DA-13-8).

6) View the cost reports generated by the design action (see Viewing Results on page DA-13-9).

Configuring the Pricing Preference Object

Figure 13-1 shows the attributes in the pricing preference object.

Figure 13-1 Attribute table in the Pricing Preference Object



It is good practice to configure the attributes in this order, as described in the following sections:

1) Regions on page DA-13-3

2) Cost Elements on page DA-13-3

3) Cost Categories on page DA-13-4

4) Work Package on page DA-13-5

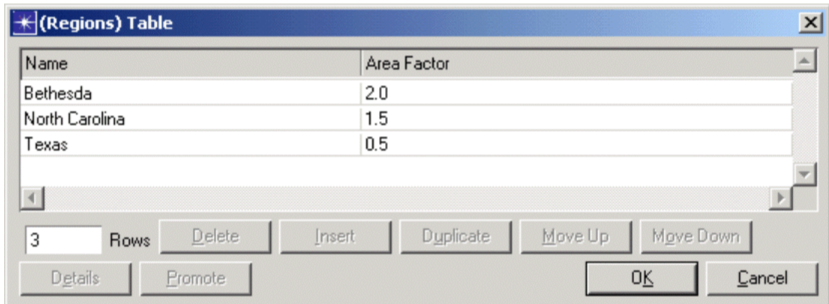
Note—If you do not set the Regions attribute (which is permissible), you cannot assign a region to an object, and the area factor defaults to 1.0. If you do not set the Work Package attribute (which is also permissible), you can set the work package name to Default only.

Note—Make sure that the associations defined in the pricing preference object remain valid when its attributes are changed. For example, if you remove an entry in the Cost Elements table, then make sure that no entries in the Cost Categories attribute refer to that deleted entry.

Regions

A “region” is a user-defined geographic area with a name and an area cost factor. The area factor defines the relative cost of a region: the more costly the region, the higher the area factor.

Figure 13-2 “Regions” Attribute Table



Cost Elements

Cost elements are user-provided descriptions of an activity (such as a site survey) that are assigned a default unit, a unit price, and the object types associated with this cost element. The Applicable To attribute defines the object types with which a cost element can be associated (such as nodes only, links only, or nodes and links).

Figure 13-3 “Cost Elements” Compound Attribute

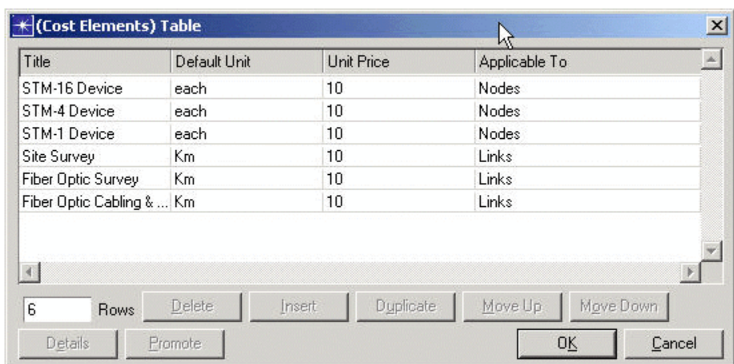
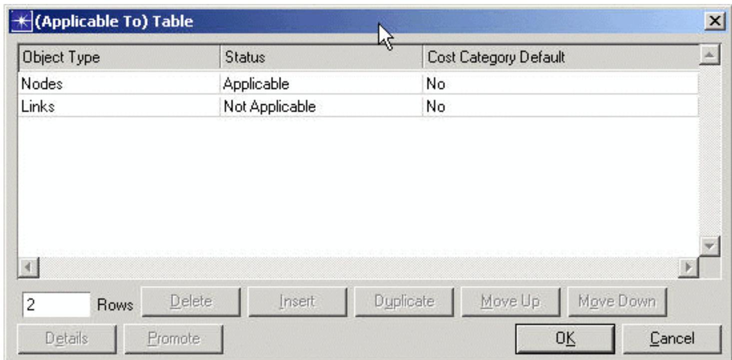


Figure 13-4 shows the Applicable To attribute, which is part of the Cost Elements compound attribute. (The significance of the Cost Category Default field will become clear in the following section.)

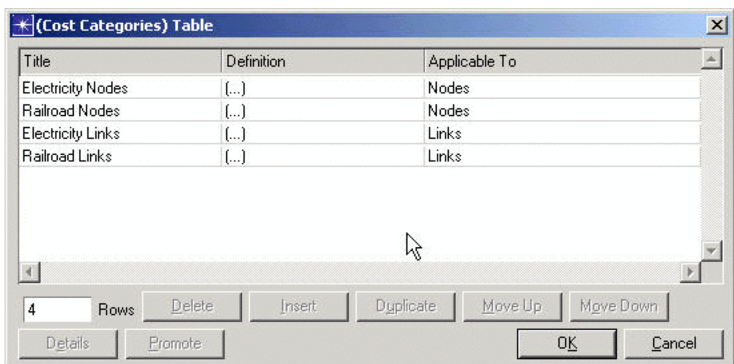
Figure 13-4 "Applicable To" Attribute of "Cost Elements" Compound Attribute



Cost Categories

Cost categories are user-specified categories that group together cost elements defined in the Cost Elements attribute.

Figure 13-5 "Cost Categories" Attribute Table



The Applicable To attribute specifies the object types to which the cost category applies.

The compound attribute Definition defines the list of cost elements that are associated with a specific cost category. The value Add All Elements (in the Definition drop-down list) adds all the defined cost elements that

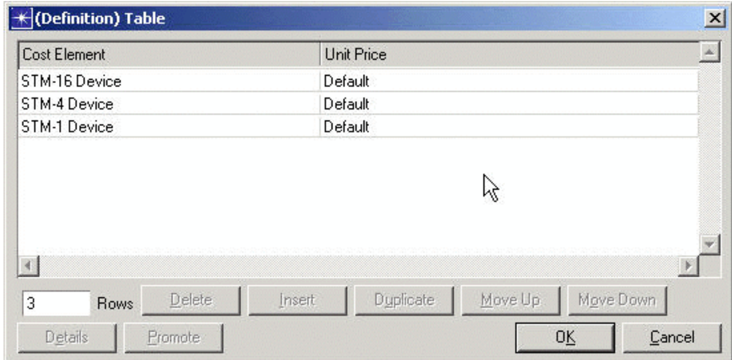
1) have the same object type association as the cost category, AND

2) have the Cost Category Default attribute set to Yes (see Figure 13-4).

This is simply a mechanism to populate the cost category with its cost elements.

Figure 13-6 shows the Definition compound attribute. If the Unit Price is set to Default, the corresponding cost-element unit price is inherited from the “Cost Elements” Compound Attribute. (You can override the default price if desired.)

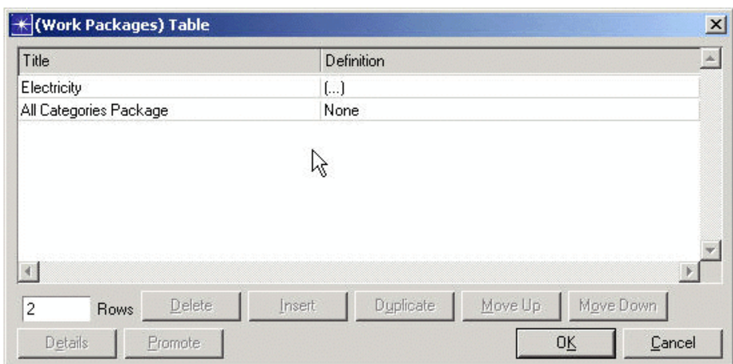
Figure 13-6 “Definition” Attribute of “Cost Categories” Compound Attribute



Work Package

A work package groups together one more cost categories defined in the Cost Categories attribute. When the Definition attribute is set to None, all defined cost categories are associated with the work package.

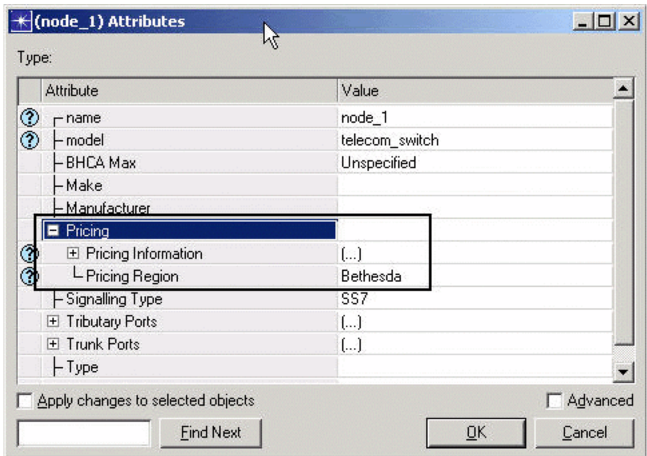
Figure 13-7 “Work Packages” Attribute Table



Specifying Pricing Information on Nodes and Links

After you specify the cost settings in the pricing preference object, the next step is to assign cost settings to the links and nodes in your scenario. Some node and link models contain a Pricing compound attribute, which is used to map the cost models defined in the pricing preference object. These models are available on the Pricing object palette.

Figure 13-8 Pricing Attributes on Supported Models



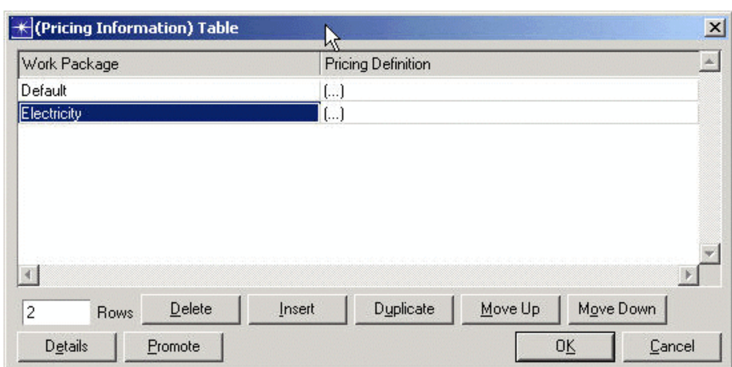
The Pricing attribute contains the following attributes:

1) Pricing Region—Selected from the list of regions that were defined in the Regions attribute of the pricing preference object. The pricing region defines the object's area factor.

2) Pricing Information—Ties together the work package, the cost categories that belong to the work package, and the cost elements associated with each cost category.

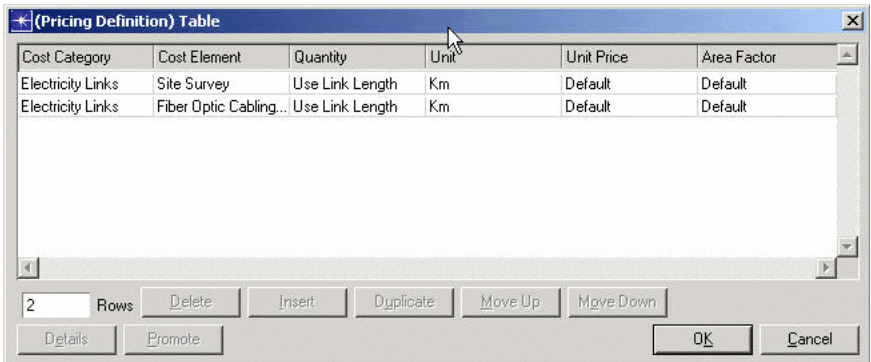
Pricing Information Compound Attribute

Figure 13-9 "Pricing Information" Attribute Table



By default, each Pricing Information entry has its work package set to Default. This special value enables you to select all the defined entries for the Cost Categories compound attribute in the pricing preference object.

Figure 13-10 “Pricing Definition” Attribute Table



The Pricing Definition table defines the applicable cost category and its elements. For each Cost Category/Cost Element pair, you can specify the following:

- Quantity—Specifies the number of units for the corresponding cost element entry. Links support a special value called Use Link Length when the unit of the cost element is in kilometers. In this case, the quantity is obtained from the Manual Length attribute of the link model. If the value of the Quantity attribute is set to As Calculated, the value of the Calculated Length attribute is used.
- Unit Price—Specifies the unit price for the corresponding cost category and cost element pair. If the value for Unit Price is Default, then the value for the unit price is obtained from the corresponding entry in the Cost Categories attribute of the pricing preference object. You can override this value here.
- Area Factor—A factor that multiplies the final cost of the corresponding cost category and cost element pair. If the value for Area Factor is Default, then the value for the area factor is obtained from the Area Factor entry in the Regions attribute of the pricing preference object. If the Pricing Region value is unspecified, the area factor is defaulted to 1.0. You can override the default here.
- Unit—Specifies the unit of the Cost Element. The value for the Unit is picked up from the corresponding entry in the Cost Elements attribute of the pricing preference object. You cannot override this value here; you can do this in the pricing preference object only.

Running the project_pricing_reporter Design Action

The "project_pricing_reporter" design action generates cost reports based on the cost data you specified in the pricing preference object and in the links and nodes.

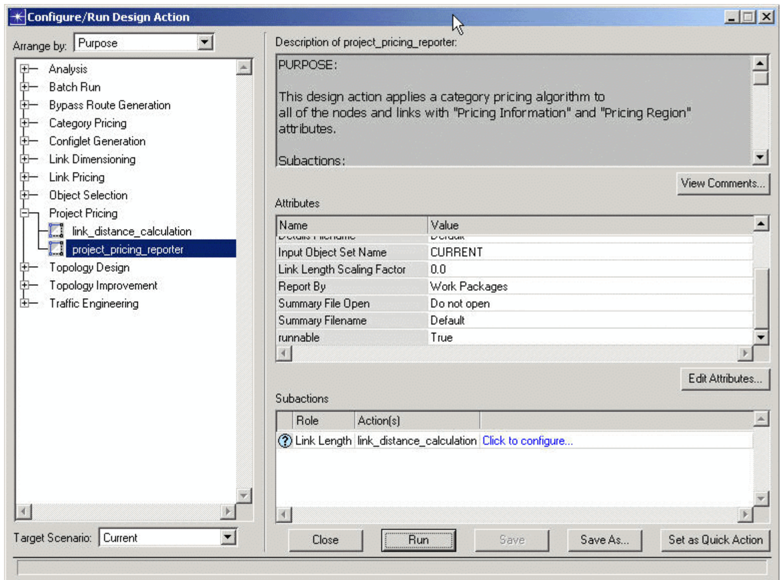
Procedure 13-1 Running the project_pricing_reporter Design Action

1 Choose Design > Configure/Run Design Action… in the Project Editor menu.

➥The Configure/Run Design Action dialog box (Figure 13-11) appears.

2 Select Purpose from the “Arrange by” menu (top left corner).

Figure 13-11Configure/Run Design Action Dialog Box



3 Select Current from the Target Scenario menu (bottom left corner).

4 Select project_pricing_reporter under the Project Pricing entry in the treeview.

5 Configure the project_pricing_reporter design action and click Run to generate the reports (for more information, see Configuring the project_pricing_reporter Design Action on page DA-13-9).

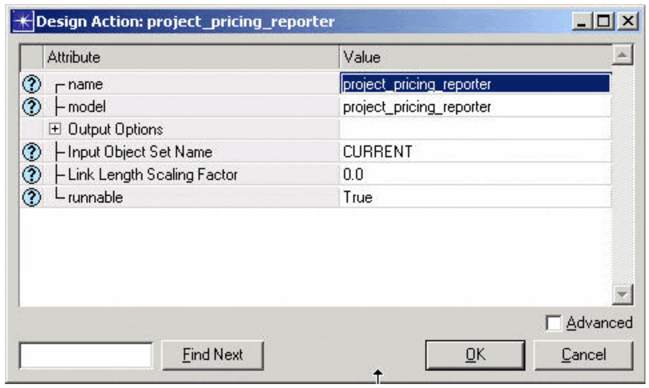
6 The design action generates four kinds of reports. View the resulting reports as described in Viewing Results on page DA-13-9.

End of Procedure 13-1

Configuring the project_pricing_reporter Design Action

Figure 13-12 shows the configurable attributes defined on the project_pricing_reporter design action.

Figure 13-12 project_pricing_report Attribute Table



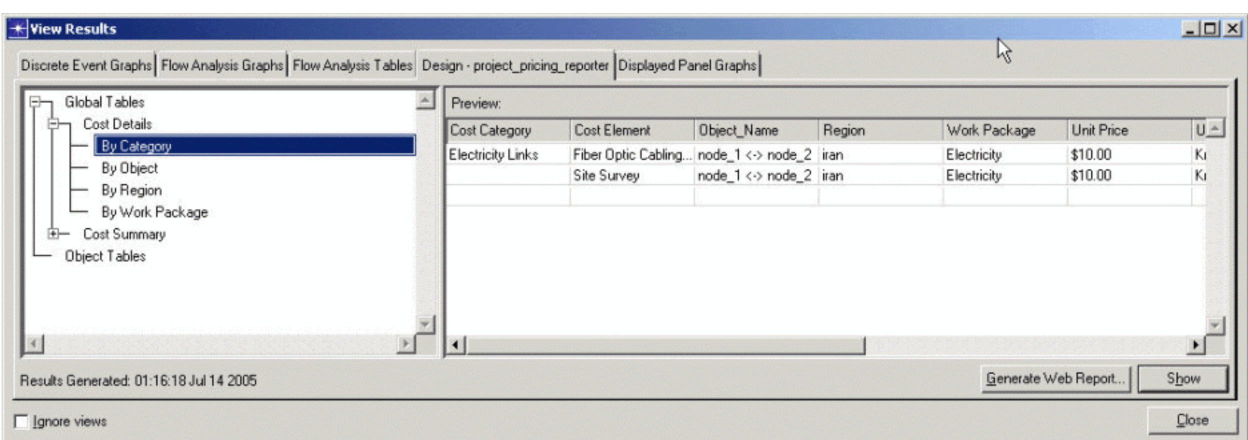
The following attributes are particularly relevant:

- Link Length Scaling Factor—Multiplies the link length when it is computed using the calculated length. This attribute is used to account for physical links that do not follow a straight path.
- Report By (Output Options)—Specifies which generated report will be exported to a CSV (comma-separated value) file. By default, the work package report is exported to CSV. Each report consists of a detailed report and a corresponding summary report.

Viewing Results

To view all the generated reports, select the menu Design > Results > View Output Tables, which launches the dialog box in Figure 13-13. Select the report type and click Show to display the corresponding table. Choose File > Export to export the resulting report to various formats.

Figure 13-13 Results for project_pricing_reporter Design Action



Index

O

P

Q

R

S

T

W