



Cisco WAE Design 7.2.2 User Guide

First Published: 2020-07-30

Americas 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: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <https://www.cisco.com/c/en/us/about/legal/trademarks.html>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2020 Cisco Systems, Inc. All rights reserved.



CONTENTS

CHAPTER 1

Overview 1

- Use Cases 1
- Plan Files 2
- Patch Files 3
- WAE Design Tools 3
 - Extensible Tools 4

CHAPTER 2

User Interface 5

- Starting WAE Design 5
- Main Window 5
 - Visualization Toolbar 6
- Viewing Plots 8
 - Viewing Network Plots 8
 - Viewing Site Plots 9
 - Viewing Interface Plots 9
- Objects 10
 - Object Selections 11
 - Filtering to Selected Objects 13
 - Filtering to Nested or Parent Objects 13
 - Refining Viewable Objects 13
- Configuring Preferences 15
- Tables 16
 - Common Tables 16
 - Showing or Hiding Tables or Columns 17
 - Searching and Filtering in Tables 18
 - Saving Table Settings or Restoring Defaults 19

Other Table Actions	20
Context Menus	20
Log and Report Windows	21

CHAPTER 3**Plan Objects 23**

Editing Objects Quick Reference	23
Understanding Nodes and Sites	24
Parent Sites and Contained Objects	24
Deleting Sites	25
PSN Nodes	26
Creating Nodes and Sites	27
Creating Nodes	27
Creating Sites	27
Duplicating Nodes	28
Duplicating Sites	28
Merging Nodes	28
Trimming Nodes and Moving Their Traffic	30
Circuits and Interfaces	31
Layer 1 Relationships	31
Creating Circuits and Interfaces	31
Duplicating Circuits	32
Creating a Circuit Mesh Between Nodes	33
Merging Circuits	34
SRLGs	35
Creating SRLGs	35
Creating SRLGs for Circuits Only	35
Ports, Port Circuits, and LAGs	36
Creating Ports	37
Layer 1 Relationships	38
Creating Port Circuits	38
Creating LAGs	39
Setting LAG Simulation Properties	39
Editing Object Properties	39
Tags	40

Adding Tags	40
Enabling and Disabling Tags	40

CHAPTER 4**Traffic Demand Modeling 43**

Demands	44
Demand Sources and Destinations	45
Demand Meshes	46
Demand Latency Bounds	46
Visualizing Demands	47
Viewing Demand Plots	48
Viewing Shortest Paths and Shortest Latency Paths	49
Demand Traffic	50
Creating and Editing Traffic Levels	50
Demand Deduction	51
Example Demand Deduction	51
Differences in Measured and Simulated Traffic	54
Minimizing Differences Between Measured and Simulated Traffic	55
Flow Measurements in Demand Deduction	56
Creating Demands and Demand Meshes	57
Creating Demands	57
Duplicating Demands	58
Creating Demand Meshes	59
Creating Demand Meshes for LSPs	61
Creating Demand Latency Bounds	61
Adding and Modifying Demand Traffic	63
Estimating Demand Traffic Using Demand Deduction	63
Generating Demand Traffic Using Node Traffic Estimates	64
Modifying Demand Traffic	66

CHAPTER 5**Simulation 69**

Traffic Simulation	69
State	70
Failed State	70
Failing and Recovering Objects	72

Active State	73
Setting Active or Inactive States	74
Operational State	74
Simulated Capacity	75
Simulated Delay and Distance	76
Latency and Distance Initializer	76

CHAPTER 6**Simulation Analysis 79**

Worst-Case Traffic Utilization	80
Worst-Case QoS Violations	81
Failing Circuits to Worst-Case Utilization	83
Worst-Case Demand Latency	84
Failing Demands to Worst-Case Latency	84
Failure Impact	85
Simulation Analysis Reports	86
Protecting Objects	87
Running Simulation Analysis	88
Viewing Simulation Analysis on a Subset of Failure Scenarios or Service Classes	89

CHAPTER 7**Traffic Forecasting 91**

Demand Groupings	91
Demand Groupings and Their Filters	91
Creating a Single Demand Grouping	94
Creating a Demand Grouping Mesh	95
Demand Groupings Reports	98
Example of Capacity Bound and Capacity Violation	98
Growth Plans	100
Demand Grouping Traffic Growth	100
Single-Period Demand Grouping Traffic	101
Multi-Period Demand Grouping Traffic	102
Demand Traffic Growth	102
Interface Traffic Growth	102
Creating Growth Plans	103

CHAPTER 8**IGP Simulation 105**

- OSPF and IS-IS 105
 - OSPF Area Simulation 105
 - IS-IS Multi-Level Simulation 106
 - Setting IS-IS Multi-Level Simulation 106
- EIGRP 107
- IGP Multipath Simulation 107
- Multi-Topology Routing 108
 - Editing Topologies 108
 - Adding Topologies to Objects 108
- IGP Metric Initializer 109
- Viewing Shortest IGP, Latency, and TE Paths 110
 - Plotting Shortest Paths 111
- Setting IGP Simulation 112

CHAPTER 9**MPLS Simulation 113**

- LSP Types 113
- Creating and Visualizing LSPs and LSP Paths 114
 - LSP Paths 115
 - Path Options and Active Path 115
 - Creating LSP Paths 115
 - Visualizing LSP Shortest Paths 116
- Path Latency Calculations 117
- Explicit LSP Paths Initializer 117
- Routing Demands 118
 - Intra-Area LSPs 118
 - Inter-Area LSPs 119
- Loadsharing 119
- Class-Based Forwarding 120
 - Class-Based Forwarding Simulations 120
 - Example Simulations 122
 - Workflow for Creating Service Class and LSP Class Mappings 124
 - Creating LSP Classes 125

Assigning LSP Classes to LSPs	125
Creating Mappings Between Service Classes and LSP Classes	126
Private LSPs	126
Creating Private Demands for Existing LSPs	127
Creating Private LSPs for Demands	127
Deleting Demands when Private LSPs are Deleted	128
Configuring LSPs to Transport Specific Traffic Types	128
Routing Inter-Area LSPs	129
Order of Routing Through Areas	129
Explicit Versus Dynamic Inter-Area LSP Routing	130
Simulation Modes	132
LSP Establishment Order	132
Reports and Diagnostics	133
LSP Path Reports	133
Simulation Diagnostics	133
Using Simulation Diagnostics to Troubleshoot	133
Running LSP Simulation Diagnostics	134

CHAPTER 10

RSVP-TE Simulation	135
Dynamic LSP Routing and CSPF	135
Interface MPLS Properties	136
Viewing LSP Reservations	137
LSP Setup BW Initializer	138
RSVP LSP Paths	138
Hot Standby	138
Example Response to Failures	139
Named Paths and Explicit LSP Routing	140
Named Path Hops Example	140
Creating Named Paths and Their Hops	141
Editing Named Paths and Named Path Hops	142
Actual Paths	143
Deactivating Actual Paths for Simulations	144
Affinities	144
Creating and Editing Affinities	145

Assigning Affinities to Interfaces	145
Associating LSPs with Affinities	146
Inclusion and Exclusion Rules	146
Example	146
Assigning LSPs to Affinities	147
Assigning LSP Paths to Affinities	147
Assigning Affinities When Creating LSP Meshes	148
Simulation Modes	148
Fast Reroute Simulations	149
FRR Fundamentals	149
FRR LSP Routing	150
Link and Node Protection	151
SRLG Protection	152
Routing Around Failures	153
Setting Up Fast Reroute Simulations	154
Running Fast Reroute Simulation	158
Autobandwidth Simulations	159
Example Autobandwidth Convergence Without and with Failures	160
Advanced RSVP-TE LSP Simulations	163
Ignoring Reservable Bandwidths for Capacity Planning	163
Example	163
Enabling Capacity Planning Mode for LSPs	166
P2MP LSPs	167
Plotting P2MP LSPs	168
P2MP LSP Bandwidth	168
P2MP LSP Demands	170
Explicit P2MP LSP Path Initializer	171
Creating P2MP LSPs and Sub-LSPs	173
Creating Demands for P2MP LSPs	175
Deleting P2MP LSPs and Sub-LSPs	175
Unresolved LSP Destinations and Hops	176
Resolving Plan File to Objects	177

SR LSP Segment Types	179
Node Segment List Hops	180
Interface Segment List Hops	180
LSP Segment List Hops	181
Anycast Group Segment List Hops	182
SR LSP Paths	184
SR LSP Routing	185
Inter-AS SR LSP Routing	185
Creating SR LSPs and Their LSP Paths	187
Creating Segment Lists	188
Creating Anycast Groups	189
SR-TE Protection	189
SR-TE Tunnel Simulation Before Convergence	189
SR-TE Tunnel Simulation After Convergence	189
Constraints	190

CHAPTER 12

Layer 1 Simulation	191
L1 and L3 Views	191
Alternate Views	192
Multi-Layer Views	192
L1 Topology	193
L1 Objects	194
L1 Waypoints	195
Sites	195
L1 Simulations	195
Active L1 Path Selection	196
Standby Paths and Path Options	196
L1 Circuit Path Routing Order	196
Viewing L1 Circuit Paths	197
L1 Circuit Path Hops	198
Explicit L1 Circuit Paths Initializer	199
L1 Circuit Route Reports	199
L1 Link Metrics	200
L1 Link Metric Initializer	200

L1 Circuit Path Setup Priority	201
L1 Circuit Wavelengths	201
Transmission Feasibility	202
Spectrum Modeling	203
Central Frequency ID Assignments	204
Central Frequency ID Blocked List	207
L1 Diversity Constraints	208
Relationships Between L1 and L3	208
Example Relations Between L1 Link and L3 Circuit	209
Delay and Distance	212
L1 Failures	214
L1 Failure Effects on L3	215
Resiliency Under L1 Failures	216
Minimum Routed Paths	218
Inactive State	219
L1 Simulation Analysis	220
Worst-Case Simulations	220
Failure Impact	220
Worst-Case Failures	222
Creating L1 Topology	222
Creating L1 Topology from L3 Topology	222
Creating L1 Nodes	223
Creating L1 Links	224
Creating L1 Circuits	224
Plotting L1 Circuits	225
Creating L1 Circuit Paths	225
Plotting L1 Circuit Paths	226
Editing L1 Circuit Paths	226
Creating Individual L1 Circuit Path Hops	227
Creating L1 Waypoints	228
L1 Link Properties	228
L1 Link Context Menu	228
L1 Waypoint Context Menu	229
Duplicating L1 Objects	229

Creating Central Frequency ID Blocked List	229
Inserting L1 Diversity Constraint	229
Creating L1 and L3 Mappings	230
Creating L3-L1 Links Between Nodes	230
Mapping Multiple L1 Circuits to L3 Circuits	231
Examples	232
Mapping an L1 Circuit to an L3 Circuit	234
Mapping an L1 Circuit to an L3 Port Circuit	235
Mapping L1 Ports to L3 Ports	235
Visualizing the L1 Network	235
Editing L1 Plot Options	236
L1 Link Appearance	236
Multi-Layer Visualization	236

CHAPTER 13
Quality of Service Simulation 239

QoS Parameters	239
QoS Bound and QoS Violations	241
Policies and QoS Bound Calculations	243
Interface Queue Properties and QoS Bound Calculations	244
Priority	244
Weight	245
Police Limits	245
Interface QoS Bound Calculations Using Multiple QoS Parameters	246
Service Class QoS Bound Calculations Using Multiple QoS Parameters	246
Viewing Queue and Service Class Information	247
Viewing QoS Bounds and QoS Violations	247
Creating Service Classes	248
Creating Queues	249
Assigning Queues to Interfaces	250
Mapping Service Classes to Queues	250
Creating or Editing Policy Groups for Interfaces	250
Creating or Editing Service Class Policies	251
Editing Interface Queues Properties (QoS Requirements)	252

CHAPTER 14	BGP Simulation	253
	Internal and External AS Types	253
	Inter-AS Routing	254
	Route Selection Between Internal ASes	254
	Routes Between External and Internal ASes	255
	External Meshes	257
	BGP Simulations	259
	BGP Routing Details	259
	BGP Multihop	259
	BGP Load Balancing	259
	BGP Next Hop	260
	Trimming ASes	260
	Creating ASes	261
	Associating Nodes with an AS	262
	Editing AS Routing Policies	262
	Assigning Names to ASNs	263

CHAPTER 15	Advanced Routing with External Endpoints	265
	Routing with External Endpoints	265
	External Endpoint Members	266
	Routing Simulations	267
	Traffic Distribution	267
	States	269
	In-Network Routing Examples	270
	Content from Multiple Data Centers	270
	External Source	272
	Intra-AS Routing Examples	273
	Failover to Nodes in External ASes	273
	Failover from an External AS to Another	275
	Creating External Endpoints and Their Members	276

CHAPTER 16	VPN Simulation	279
	VPN Model	280

VPN Objects	280
VPN Topology and Connectivity	280
VPNs	281
VPNs Table	281
Identifying Interfaces Used by VPNs	282
VPN Nodes	283
VPN Nodes Table	283
Layer 3 VPN Example	284
VPN Simulation Analysis	287
Creating VPN Nodes	288
Creating VPNs	288
Creating VPNs from Existing VPN Nodes	288
Creating New VPNs	290
Adding VPN Nodes to VPNs	290

CHAPTER 17

Multicast Simulation	291
SSM Parameters	291
Discovered Versus Simulated Multicast Flows	291
Global Multicast Simulations	293
Flow Hops	293
Cisco Next Hop	294
Multicast Demands	294
Initialized Multicast Demands	295
Simulated Multicast Demands	295
Viewing Multicast Flows	296
Creating Multicast Flows	297
Creating Demands for Multicast Flows	299
Toggling Multicast Flow Destinations to On and Off	299

CHAPTER 18

Metric Optimization	301
Core Versus Edge	301
Metric Optimization	302
Tactical Metric Optimization	304
Reports	305

Metric Optimization Report 305

Warnings 305

Routing Summary 307

Metrics 307

CHAPTER 19**LSP Optimization 309**

LSP Disjoint Path Optimization 309

Disjoint Routing Selection 309

Disjoint Path Requirements 310

Example 311

Constraints 313

Example 313

Disjoint Report 314

Prerequisites 315

Creating Disjoint Groups 315

Optimizing Disjoint Paths 315

LSP Loadshare Optimization 316

Running the LSP Loadshare Optimization Tool 316

Optimization Options 317

Minimization Example 317

Bin Example 320

LSP Setup Bandwidth Optimization 320

Workflow 320

Summary Report 321

LSP Groups Report 321

CHAPTER 20**RSVP-TE Optimization 323**

Optimization Input 324

Selecting LSP Groups 325

Setting Optimization Parameters 326

Setting Disjoint Groups 327

Setting Avoidance Constraints 328

Setting Post-Optimization Parameters 328

Optimization Output 328

- Properties Created 328
- Optimized LSPs Reconfigured 329
- Reports 329
- Examples 329
 - Optimizing New LSPs Example 329
 - Tactical Mitigation Example 331
- Workflow 333

CHAPTER 21 **Explicit and Tactical RSVP-TE LSP Optimization 335**

- Running the RSVP-TE LSP Optimization Tool 335
 - Setting Optimization Details 338
- Tactical Explicit RSVP-TE LSP Optimization 340

CHAPTER 22 **Segment Routing Optimization 341**

- SR-TE Optimization 341
 - Optimizing SR-TE LSPs 342
 - Optimizing the Path Metric 343
 - Bound and Margin 344
 - Bound: Maximum Acceptable Path Length 344
 - Margin: Maximum Acceptable Metric Above Shortest Path 344
 - Constraints 345
 - Using SR Tags and Creating New Plan Files 346
 - Optimization Report 347
- SR-TE Bandwidth Optimization 347
 - Operating Modes 347
 - Constraints 347
 - Bandwidth Optimization Report 348
 - Examples 348
 - Optimize Bandwidth 351
 - Analyzing Congestion Under Different Failure Sets 352
- Compressing Segment Lists 352

CHAPTER 23 **Capacity Planning Optimization 355**

- Workflow 356

Reports	358
Summary Report Example	358
Capacity Upgrades Report Example	359
Layer 3 Optimization Options	359
Parallel Circuits Design Example	361
Layer 1 Optimization Options	361
Creating Layer 1 Circuits	362
Advanced Optimization Options	362

CHAPTER 24	L1 Circuit Path Optimization	365
	Running the L1 Circuit Path Optimization Tool	365
	Reports	366
	Example	366

CHAPTER 25	Changeover	369
	Running the Changeover Tool	369
	Reports	370
	Summary	370
	Steps	371
	Utilizations	372

CHAPTER 26	Patch Files	373
	Opening a Remote Plan File	373
	Creating Patch Files	375
	Applying Patch Files	376
	Viewing Patch Files	376
	Sending Patch Files	377
	Saving and Clearing Settings	378

CHAPTER 27	Reports	379
	Plan Comparison Reports	380
	Report Columns	380
	Report Sections	385
	Creating Plan File Comparison Reports	386

Traffic Comparison Reports	387
Creating Traffic Comparison Reports	387
Viewing or Exporting Reports	388

CHAPTER 28**Cost Modeling 389**

Infrastructure Costs	389
Cost Basis	390
Setting Cost Properties	390
Cost Circuit Initializer	391
Infrastructure Cost Report	392
Example Infrastructure Cost Workflow	393
Demand and Demand Grouping Costs	396
Example	396

CHAPTER 29**Plot Legend for Design Layouts 399**

Interfaces and Circuits	400
Circuit Colors	400
Circuit Active, Failed, and Operational States	402
L3 Nodes	404
Sites	405
Demands	407
Demand Segment Lists	407
External Endpoint Members	408
LSPs	409
LSP Paths	409
LSP Named Paths	410
LSP Actual Paths	410
Segment-Routed LSP Paths	411
L1 Objects	412
L1 Links	412
L1 Circuit Paths	413
L1 Circuit Path Hops	414
L1 Nodes, L1 Ports, and L1 Waypoints	414



CHAPTER 1

Overview

WAE Design is a standalone application that you can run on multiple platforms, or on standard laptops or desktops.



Note This guide references `$CARIDEN_HOME`, which is the directory in which the Cisco WAE executables and binaries are installed. On Linux, the default is `/opt/cariden/software/mate/current`, where `/opt/cariden` is the default installation directory.

This section contains the following topics:

- [Use Cases, on page 1](#)
- [Plan Files, on page 2](#)
- [Patch Files, on page 3](#)
- [WAE Design Tools, on page 3](#)

Use Cases

WAE Design lets you model, simulate, and analyze failures, design changes, and impact of traffic growth, as well as optimize your network for maximum efficiency. For example, with WAE Design models and tools, you can answer a wide variety of questions about traffic management.

Planning:

- Where is traffic going in the network?
- When and where will my network run out of capacity?
- How do I convert my growth forecasts into upgrade plans?

Engineering:

- How can I better distribute traffic throughout the network?
- How can I design for differentiated quality of service?
- Where is my network most vulnerable to failure, and how can I mitigate it?
- How will a Layer 1 design affect the resilience of Layer 3?

Operations:

- How can I visualize the current health of the network?
- What caused the outage experienced earlier today?
- How would a current failure affect the network at peak time tomorrow?
- Which VPN customers will be affected by a planned outage or by an unplanned failure?

Plan Files

A *plan file* (Figure 1: Opened Plan File, on page 2) is comprised of a series of tables that store information about a network, including topology, configuration information, traffic, failure state, and visual layout. Additionally, WAE Design uses plan file information to perform simulations.

Figure 1: Opened Plan File

Plan File Opened in Excel

<Nodes>			
Name	Site	Function	Protected
cr1.ams	AMS	core	F
cr1.fra	FRA	core	F
cr1.lon	LON	core	F
cr1.par	PAR	core	F
er1.ams	AMS	edge	T

← **Table Name**
← **Column Names**
← **Values**

Plan File Opened in the GUI

Nodes			
Name	Site	Function	Protected
cr1.ams	AMS	core	F
cr1.fra	FRA	core	F
cr1.lon	LON	core	F
cr1.par	PAR	core	F
er1.ams	AMS	edge	T

<Sites>			
Name	LocationCode	Longitude	Latitude
AMS	AMS	4.78	52.32
BRU	BRU - Brussels (Belgium)	4.48	50.9
FRA	FRA	8.57	50.03
LON	LON	-0.45	51.47
PAR	PAR	2.55	49.02

Sites			
Name	Location Code	Longitude	Latitude
AMS	AMS	4.78	52.32
BRU	BRU - Brussels (Belgium)	4.48	50.90
FRA	FRA	8.57	50.03
LON	LON	-0.45	51.47
PAR	PAR	2.55	49.02

Initially, network topology and routing information is captured by WAE discovery tools and stored in plan files. These plan files are the basis of information displayed in and used by WAE Design. Plan files contain:

- Network configurations
- Visual layouts
- IP/MPLS routes, including multicast and LSPs
- Measured traffic
- Estimated end-to-end traffic matrix
- Operational state of network objects
- Results of analyses, such as worst-case failure analysis results

Plan files have two formats, both of which are editable by SQL commands:

- The .pln format is compact and can be quickly loaded to and saved in the WAE Design GUI.

- The .txt format contains tab-delimited columns. You can create and edit these directly in a text editor or spreadsheet, such as Excel, and quickly apply bulk edits. In these plan files, each table is labeled with angle brackets, such as <Nodes> and <Sites>.

Patch Files

A *patch file* is a compact way to represent differences between plan files. These differences or “patches” can be applied to other plan files or deployed to the network. Patch files have a .plp format.

For information on creating and sending patches, see [Patch Files, on page 373](#).

WAE Design Tools

WAE Design contains numerous tools with user-friendly features, such as easy navigation of object dependencies, bulk editing, and tight interaction between plots and tables. Tools fall under these categories:

- Visualization—Provides graphical layouts of the network, showing views of different regions and layers, as well as utilizations and routings.
- Simulation—Lets you perform interactive what-if simulations by, for example, failing objects, modifying network topology, creating and changing the traffic demands (which simulate traffic flows), and modifying routing configurations.
 - Demand deduction—Estimates an end-to-end traffic flow matrix based on available measured traffic, such as interface utilizations and LSP traffic.
 - Failures and simulation analysis—Allows evaluation of the effect of failures of network objects, including circuits, nodes, and Layer 1 equipment. You can evaluate overall risk using worst-case analyses over these failures.
 - Forecasting—Lets you create plans for future expected traffic levels.
- Offline traffic engineering—Lets you optimize the network’s reliability and performance.
 - Optimization—Suggests changes that let you optimize the network, for example, by reducing congestion under circuit failures.
 - Changeover—Recommends a step-by-step series of configuration changes to maneuver a network safely from one configuration state to another.
- Other tools:
 - Reporting—Lets you generate reports that compare IP/MPLS topology and configuration information, demand routings, demand groupings, and LSP routings. You can also generate reports that compare traffic, compare plan files, and estimate network infrastructure costs. Other reports, such as Simulation Analysis and Demand Deduction, are automatically generated.
 - Importing, exporting, and getting plans—Multiple tools are available for importing traffic and objects, exporting routes, tables, explicit LSP settings, and spectrum utilization, and for getting data from sources such as config files and IGP databases.

Extensible Tools

The plan file format is open and documented. Add-on capabilities are available for extending GUI functionality. Additionally, command-line tool equivalents of the GUI tools are provided for scripting and automation. For information, see the *Cisco WAE Design Integration and Development Guide* .



CHAPTER 2

User Interface

This chapter explains how to use the WAE Design GUI.

This section contains the following topics:

- [Starting WAE Design, on page 5](#)
- [Main Window, on page 5](#)
- [Viewing Plots, on page 8](#)
- [Objects, on page 10](#)
- [Configuring Preferences, on page 15](#)
- [Tables, on page 16](#)
- [Context Menus, on page 20](#)
- [Log and Report Windows , on page 21](#)

Starting WAE Design

To start WAE Design from the directory in which the WAE Design software is installed, do one of the following:

- On Windows or Linux, double-click the **wae_design** executable.
- On a Mac, navigate to the directory where you unpacked the distribution and double-click the **WAE_Design** application. Alternately, from the **Terminal**, enter:

```
cd  
<directory-where-you-unpacked-the-distribution>  
open ./WAE_Design.app
```

- From the CLI, enter **./wae_design**.

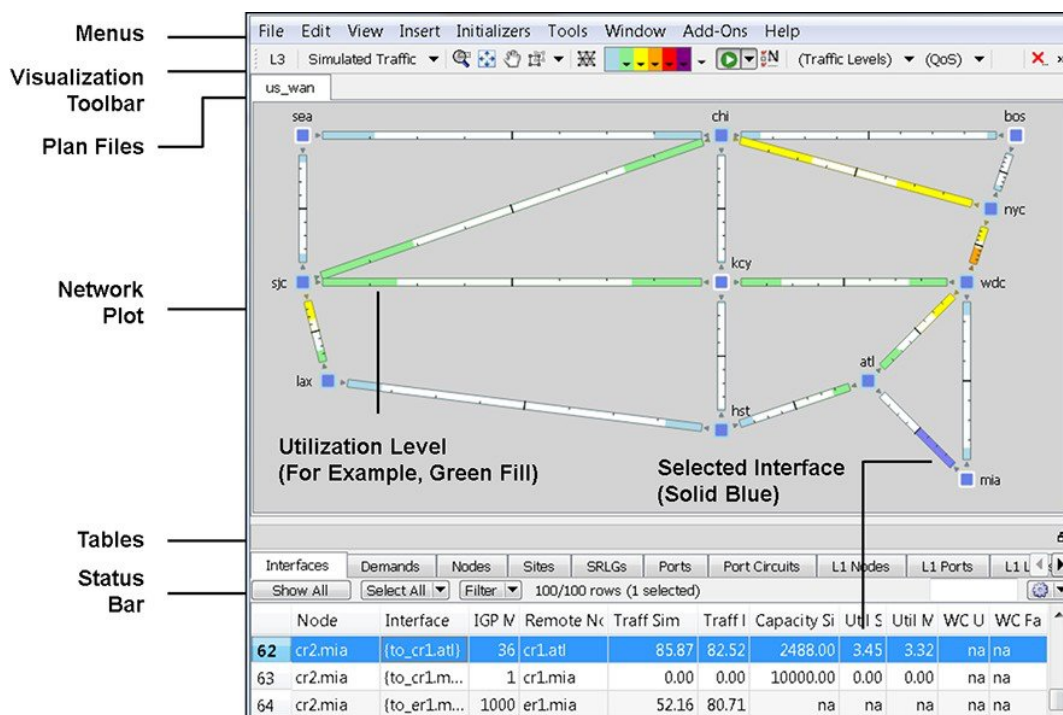
(Optional) On Windows, you can associate the plan file using the .pln format with the wae_design executable. Double-clicking a .pln file opens the plan in a new instance of the GUI.

Main Window

The WAE Design GUI is divided into the following primary sections. [Figure 2: WAE Design User Interface, on page 6](#) shows the GUI with a sample us_wan.txt plan in the network plot area.

- Menus—Menu selections for network modeling, simulation, and optimization.
- Visualization toolbar—Drop-down lists and icons that control how the network is displayed.
- Plan file tabs—Identify which plan is currently opened. There can be multiple plans open; the tab with a white background is the active plan. To bring a plan to the forefront and make it the active plan, click its associated tab.
- Network plot—Displays a visual representation of the network topology. The color on each interface represents the percentage of traffic utilization on that interface. Rows that are selected in the tables, such as demands or interfaces, are highlighted in the plot with additional graphical attributes.
- Tables—Display network properties, including both measured and simulated information, for the network plan.
- Status bar—Identifies the network simulation mode.
- Context menus—Let you right-click anywhere to view a list of actions that you can take.

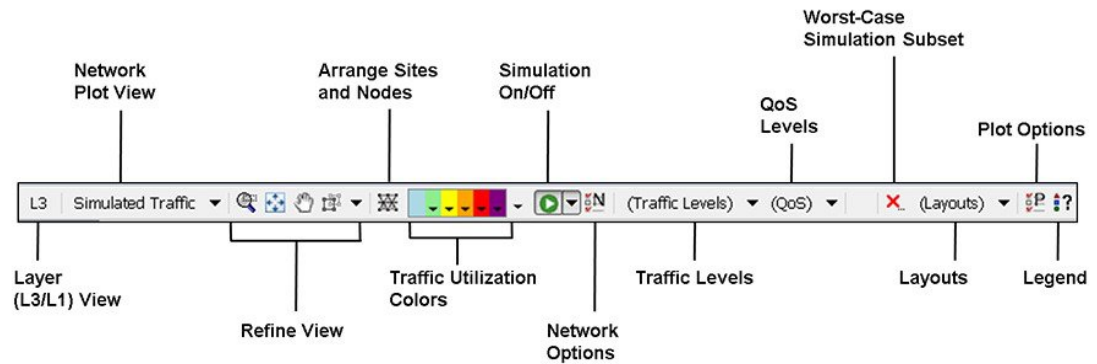
Figure 2: WAE Design User Interface



Visualization Toolbar

The Visualization toolbar (Figure 3: Visualization Toolbar, on page 7) provides a convenient way to change the visual aspects of the network plot.

Figure 3: Visualization Toolbar



- **L3/L1 View**—A toggle button to change the view between Layer 3 (L3) and Layer 1 (L1). You can also simultaneously view L3 and L1 networks. This is set in the Layer 1 tab of the Plot Options. For more information, see [Layer 1 Simulation, on page 191](#).
- **Network Plot**—A drop-down menu from which you choose the type of traffic displayed in the plot.
- **Refine View icons**—These icons let you refine and enhance your view in the network plot and in the site plots.
- **Arrange icon**—Opens an Arrange palette that lets you move selected nodes and sites into a selected schematic alignment.
- **Utilization Color**—Traffic utilizations are displayed by colors as a percent of available capacity used on that interface. This enhances the other visual cues, which is the length of the fill. You can change the colors and the utilizations they represent.



Note All traffic is displayed in Mbps. In the network plot, the traffic utilization colors represent outbound traffic.

- **Simulation On/Off** (white arrow in green circle) icon—By default, any change that voids the current simulation automatically triggers a resimulation. For example, changes in topology trigger a resimulation. To turn off this resimulation, use the drop-down arrow next to the icon to uncheck **Auto resimulate**.
- **Network Options icon**—Lets you set global simulation modes, such as Full Convergence or Fast Reroute for MPLS, and whether to include multicast hops. You can also set IGP, BGP, and Multicast protocol options.
- **Traffic Levels**—Drop-down list of user-defined traffic levels in the plan. For more information, see [Traffic Demand Modeling, on page 43](#).
- **QoS Level**—Drop-down list of service classes, interface queues, and undifferentiated traffic. From the Manage QoS option, you can add or edit service classes, assign service classes to queues, and add or edit service policies. For more information, see [Quality of Service Simulation, on page 239](#).
- **Worst-Case Simulation Subset** (red X) icon—After a simulation analysis is performed over a specified set of failure scenarios (for example, circuits, nodes, and ports), you can click this icon to select which subset of these failure scenarios (for example, circuits only), as well as service classes you want included in the worst-case analysis. This avoids having to run a second simulation analysis.

- **Layouts**—Drop-down list of user-defined plots. For example, you might have schematic layouts, geographic layouts, separate layouts for different continents, simplified layouts with hidden nodes or sites, or other representations of the network.
- **Plot Options icon**—Lets you customize circuit width, text on interfaces, layouts (such as background images and geographic views), and other plot options.
- **Legend (question mark) icon**—Displays a legend of icons and line types in the plot and their meanings. For more information, see [Plot Legend for Design Layouts, on page 399](#).

Viewing Plots

WAE Design opens to the main network plot. Although WAE Design shows only the sites in the network plot and tables, you can view those sites in more depth. You can also view more depth of interfaces and demands in their respective plots. In addition to these plot views, you can create and view layouts that show only those portions of the network plot that you choose. For more information, see the *Cisco WAE Network Visualization Guide*.

Viewing Network Plots

The network plot, also referred to simply as the *plot*, is the main area showing the network topology. It can contain both sites and nodes.

There are multiple network plot views available (top left of [Figure 3: Visualization Toolbar, on page 7](#)). The default view is Simulated Traffic. To change the view, select it from the Network Plot View drop-down list in the Visualization toolbar.

- **Simulated Traffic**—View of simulated traffic that can be used for capacity planning, what-if analysis, and failure planning. Circuits are sized proportional to their capacity. Interfaces are colored according to the percentage of simulated traffic (in comparison to available capacity as defined by the size of the interface or the QoS bound).
- **Measured Traffic**—View of traffic as measured from live views of the network. Circuits are sized proportional to their capacity. Interfaces are colored according to the percentage of measured traffic (in comparison to available capacity as defined by the size of the interface or the QoS bound).
- **Worst-Case Traffic**—Interfaces are colored to show the maximum utilization of the interface over all failure scenarios as defined by the most recently run simulation analysis. For more information, see [Simulation Analysis, on page 79](#).
- **Failure Impact**—Interfaces and nodes are colored to indicate how a failure of that interface or node would impact other interfaces and nodes. For more information, see [Simulation Analysis, on page 79](#).
- **LSP Reservations**—Circuits are sized proportional to their LSP reservable bandwidth. Interfaces are colored according to the percentage of bandwidth (in comparison to what is reserved for the LSP). For information, see [RSVP-TE Simulation, on page 135](#).
- By editing the plan file directly, you can add a user-defined static view to display interfaces with customized color fills and fill percentages. The view then appears as an option in the Network Plot drop-down list. For information, see the *Cisco WAE Design Integration and Development Guide*.

Although WAE Design focuses on IP/MPLS Layer 3 (L3) network views and traffic, you can also create and view Layer 1 topologies. To toggle between views of these two layers, choose **View > Layer 1 View** or **View > Layer 3 View**, or use the shortcuts Ctrl-1 for Layer 1 and Ctrl-3 for Layer 3. For information, see [Layer 1 Simulation, on page 191](#).

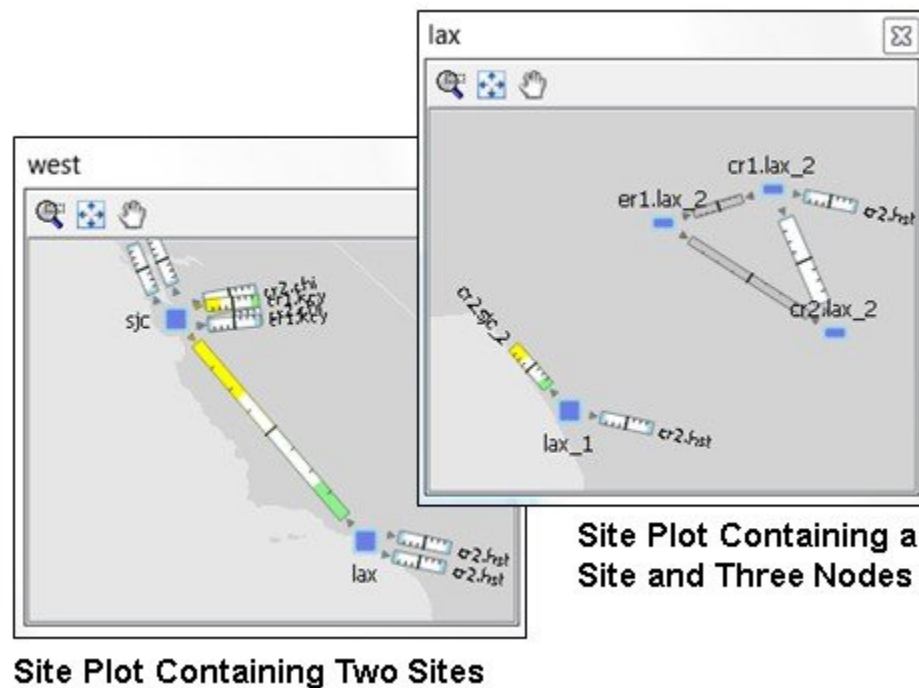
Viewing Site Plots

The site plot can contain other sites or nodes and their connecting interfaces, including external interfaces that are labeled with their destination node ([Figure 4: Site Plots, on page 9](#)).

To open a site plot, do one of the following:

- Double-click a single site from the plot.
- Choose multiple sites from the plot and then double-click one of those sites.
- Right-click one or more sites from the plot or the Sites table and choose **Open Site View**.

Figure 4: Site Plots



Viewing Interface Plots

The interface plot is a window that displays selected interfaces in a stacked format, making it easier to compare one with another ([Figure 5: Interface Plot, on page 10](#)).

-
- Step 1** Select one or more interfaces from either the Interfaces table or the plot.
- Step 2** Open the interface plot in one of two ways:

- Choose **View > Interface Plot**.
- Right-click and choose **Plot Interfaces**.

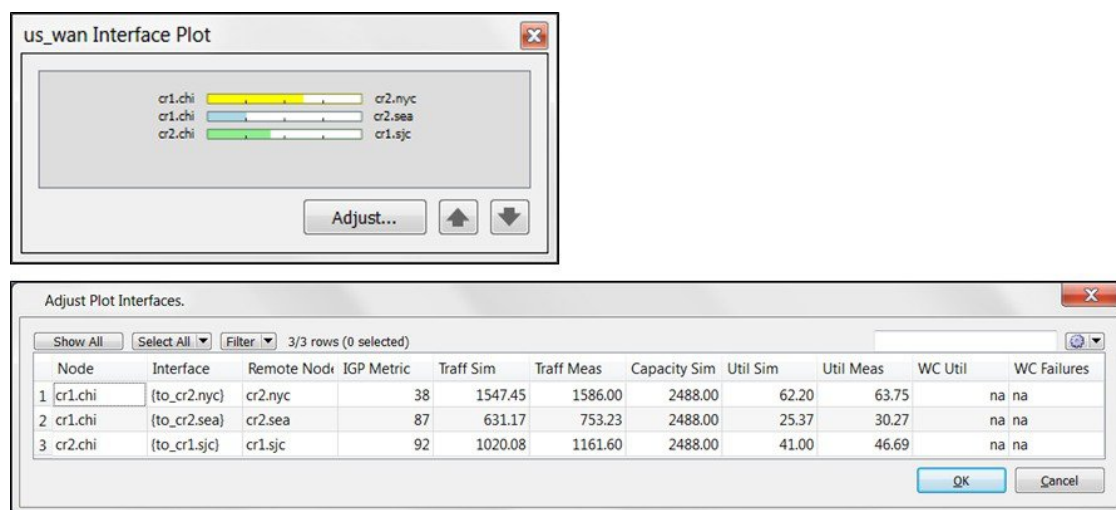
Step 3 Work with interface plots in these ways:

- To vertically move the interface list within the plot, click the up and down arrows.
- To view a table of the selected interfaces, click the **Adjust** button. A new Interfaces table appears.
- To sort the interfaces by ascending or descending order, click the **Adjust** button. Click a column heading to toggle its sorting between ascending or descending order, then click **OK**.

The interface plot displays the interfaces according to the sort.

What to do next

Figure 5: Interface Plot



Objects

Other than a site, an *object* is a representation of elements found in networks, such as nodes (which represent routers), circuits, interfaces, LSPs, and more. A site is also an object, and is a WAE Design construct for simplifying the visualization of a network by grouping nodes within a site, or even by grouping sites within a site.

Objects have *properties* that identify and define them, many of which are discovered, though they can also be manually added and changed. For example, all circuits have a discovered Capacity property that can be edited. Other properties are derived. For example, Capacity Sim is derived from the Capacity property. Another example is that interfaces have a Util Sim property that identifies the percentage of Capacity Sim the simulated traffic is using. Properties are viewable and editable through an object's Properties dialog box (right-click the object and choose Properties). They are also viewable as table columns.

Object Selections

Objects are selectable through tables and plots. However, not all objects are selectable from both, though most are. Some object types are only selectable through their tables, such as LSPs.

All selections made in a plot appear in the associated table and vice versa ([Table 1: Select Objects](#), on page 11). Likewise, changes made in the plot are reflected in the tables, and vice versa. Hidden objects are excluded from the selection.

[Table 1: Select Objects](#), on page 11 assumes that all objects are in the foreground. For information on selecting objects that are in the foreground versus background, see [Refining Viewable Objects](#), on page 13.

Table 1: Select Objects

To...	Do This...
<p>Site Selections</p> <ul style="list-style-type: none"> • If the object you select is a site, none of the child objects (child sites, nodes, and circuits) are selected. • To select a site and all objects contained within it, press and hold the Alt key during the selections identified in this table. <p>Examples</p> <ul style="list-style-type: none"> • To select only a site, click it. To select a site and all of its nested child objects, press and hold Alt while clicking the site. • To select only two sites, click one object and then Ctrl-click each site that you want to select. To select two sites and all the objects they contain, press and hold both Alt and Ctrl while selecting both sites. • To select all sites in the plan file, press Ctrl-A in either the network plot or the Sites table. To select all sites and all objects they contain, press and hold Alt and Ctrl-A together in either the network plot or the Sites table. 	
Select an object. If other objects were selected prior to this new selection, those objects become deselected.	Click one or more objects in any plot or table.

To...	Do This...
Select multiple objects from the plot.	<ul style="list-style-type: none"> • Do one of these steps from the plot: <ul style="list-style-type: none"> • Click and hold while you drag the cursor across any objects you want included in the selection. • Click one object and then Ctrl-click each object that you want to select. • Select all objects showing in a plot: <ul style="list-style-type: none"> • Press Ctrl-A (Cmd-A on a Mac). • Click and hold while you drag the cursor over an entire plot.
Select multiple objects from tables.	<ul style="list-style-type: none"> • Do one of these steps in a table: <ul style="list-style-type: none"> • Click and hold while you drag the cursor across any objects you want included in the selection. • Click one object and then Ctrl-click each object that you want to select. • Select all objects in a table: <ul style="list-style-type: none"> • Press Ctrl-A (Cmd-A on a Mac). • Click and hold while you drag the cursor over an entire table. • If you are selecting objects from multiple tables, make your selection in one table, and then press and hold the Alt key (Option key on a Mac) while making selections in another table.
Clear the entire selection.	<p>Do one of these steps:</p> <ul style="list-style-type: none"> • Choose Select All > Clear Selection. This deselects all objects within that table. If objects in other tables are selected, they remain selected. • Click anywhere in any plot. This deselects all objects in that plot and in the associated table.
Invert the current selection (everything selected becomes deselected and vice versa).	<ol style="list-style-type: none"> 1. Show the table whose selections you want to invert. 2. Choose Select All > Invert Selection.

Filtering to Selected Objects

To filter only to the selected objects, choose one or more objects in any plot or table, then choose **Filter > Filter to Selection**. Alternatively, you can press and hold Ctrl-Shift (Cmd-Shift on a Mac), and then double-click.

As with other object selections, to include the nested objects for site selections, add Alt to the selection process. For example, press and hold Alt while clicking one or more sites, and then choose **Filter > Filter to Selection**.

Filtering to Nested or Parent Objects

If a site contains multiple levels of child sites, you can filter to those nested objects. Right-click a site, choose **Filter to Contained <object>**, and then choose one of the following, where <object> equals sites, nodes, or L1 nodes:

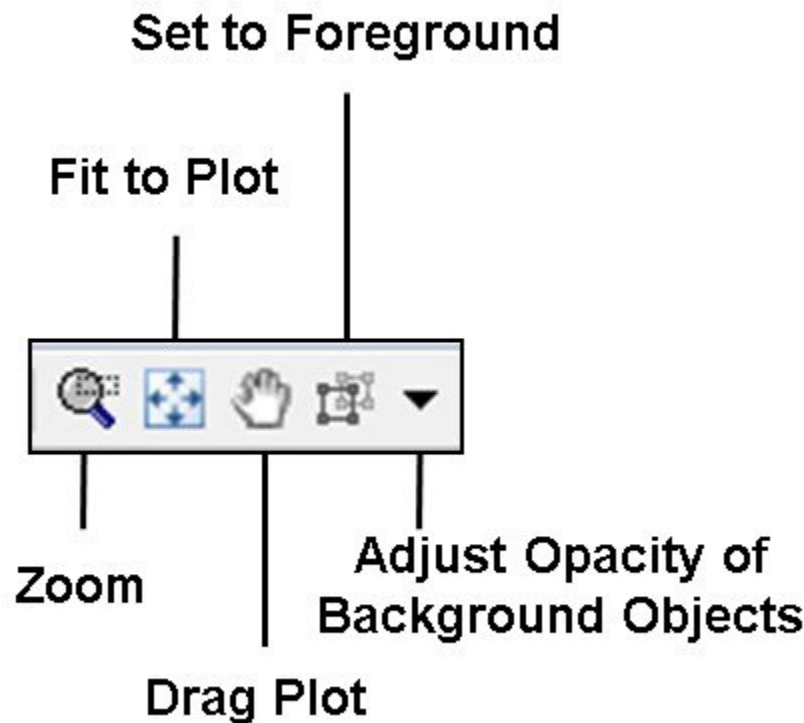
- Direct—To see only the objects immediately contained within the site.
- All—To see all contained objects, no matter how deeply nested they are within child sites.

If a site contains only one level, the context menu changes to reflect as such.

Conversely, you can find a parent site for any given node. Choose it, and then choose **Filter to Sites**.

Refining Viewable Objects

View icons are available both in the network plot and in the site plots. Each of these icons makes it easier to find what you need.



Except for Set to Foreground, when accessed through the Visualization toolbar, the view changes apply only to the network plot. When accessed through a site plot, the Zoom, Fit-to-Plot, and Drag actions apply only to that site plot.

Both the Zoom and Set to Foreground icons let you better focus and work in specific plot areas.

- Zoom—Lets you drag in the area that you want enlarged to fit the entire plot.

Shortcut—Press the letter “z,” which toggles this functionality on and off.

To escape from the Zoomed view, press Esc.

- Fit-to-Plot—Resizes the diagram to fit within and fill the plot area.

Shortcut—Press the letter “f,” which toggles this functionality on and off.

- Drag Plot—Lets you move the cursor over any portion of the plot to drag it to a different location.

Shortcut—Press the letter “d” or press the icon again to toggle this functionality on and off. The feature remains enabled until you turn it off.

To escape from the Drag option, either press Esc or click the icon to toggle it off.

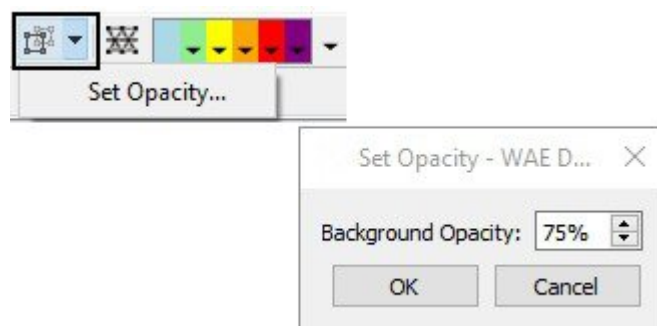
- Set to Foreground—Brings selected objects to the foreground for improved viewing while sending all other objects to the background where they appear as if they are transparent. Selected objects must be L3 nodes, L3 circuits, sites, L1 nodes, or L1 links. This feature does not apply to other objects, such as demands, LSPs, L1 circuits, and L1 circuit paths.

To escape the Set to Foreground option, click the icon to toggle it off while objects are selected.

To increase or decrease the transparency (opacity) of how well the background objects appear, click the drop-down arrow next to the Set to Foreground icon. Enter the desired value, and click **OK**. The higher the value, the denser (darker) the background objects appear.



Note If you are viewing both L3 and L1 objects, this opacity value for background objects takes precedence over the layout opacity value that is set for viewing different layers.



Configuring Preferences

You can configure scaling, foreground and background object preferences, and the background color for layouts.

Step 1 Choose **View > Preferences**.

Step 2 Click the **General** tab and configure the following options:

- **Uniform Scaling**—Enter a scale factor from 5% to 1000%. Objects and fonts are scaled uniformly based on the factor you enter.
- **Individual Scaling**—Enter an individual scale factor for L3/L1 nodes and sites, circuits and L1 links, and fonts.
WAE Design considers the settings on circuit and link width in the layouts and scales them accordingly based on the given scale factors in the network plot.
- **Enable Selection of Background Objects**—If checked, you can select objects in the background.
- **Fit to Screen Zooms to Foreground Objects**—If checked, clicking the Fit to Screen icon takes into account only the foreground objects. If unchecked, clicking Fit to Screen takes all objects into account.
- **Show Tooltips**—If checked, tooltips are shown.

One method of seeing related information about an object is to show the tooltips and then hover the mouse over a site, node, L1 node, interface, or L1 link. For example, hovering over a site you can see the number of contained L3 nodes, L1 nodes, and sites, as well as the total number of interfaces connected to the contained L3 nodes.

- **Scroll Table to Selected Object**—If checked, when you click a single plot object (site, node, L1 node, interface, or L1 link), the table automatically scrolls to the row for that object.

For example, if in the plot you click a node, and the Nodes table is open, the table scrolls automatically to the row for that node.

Step 3 Click the **Layouts** tab and configure the following options:

- **Schematic Layout**—Choose a gray or white background for schematic plots. (If sites or nodes have a white border and the background is also white, a black line appears around them.)
- **Geographic Layout**—Choose a gray or multi-color palette for geographic plots.
- **Use Site, Node, and L1 Node Icons**—If checked, icons are displayed for sites, nodes, and L1 node names.

Step 4 (Optional) To save all preferences for future sessions, check **Save Preferences**.

Step 5 (Optional) To reset the settings back to their default, click **Reset**. The dialog box refreshes automatically with the default values.

Step 6 Click **OK**.

Tables

The tables that are viewable from the WAE Design GUI display a series of rows and columns, where the rows are objects and the columns are properties (Figure 6: Common Table Functions, on page 17). Plan files also contain other tables that are not viewable from the GUI. These tables contain more complex information, such as showing complex relationships between objects. If you open a plan file using a .txt editor, each table is labeled with angle brackets, such as <Nodes> and <Sites>.



Note For detailed information about plan files and tables, as well as information on creating user-defined columns and tables, see the *Cisco WAE Design Integration and Development Guide*.

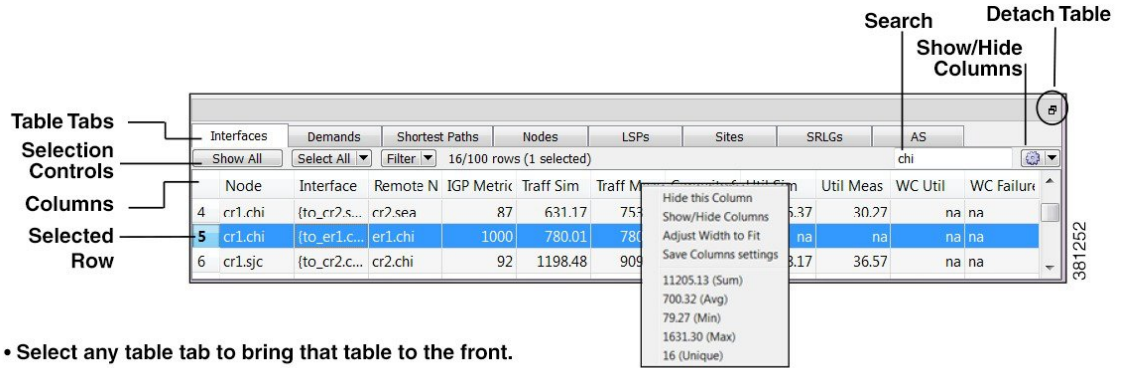
Common Tables

There are numerous tables available for specialized purposes, such as Multicast, P2MP LSPs, and Ports. The tables listed below are the most commonly used and are the defaults.

To bring a table to the forefront and make it the active table, click its associated tab. For example, to open the LSPs table, click the **LSPs** tab.

- Interfaces—A list of the interfaces in the network.
- Demands—A list of the demands in the network. Each demand specifies how much traffic is routed from a source (node, external AS, or external endpoint) to a destination (node, external AS, external endpoint, or multicast flow destination).
- Shortest Paths—A list of the routes between selected sites.
- Nodes—A list of network routers, which typically have names that suggest their location and function. For example, node cr1.atl is core router 1 in the Atlanta site.
- LSPs—A list of MPLS LSPs in the network; each LSP contains both a source and destination. Note that sub-LSPs within point-to-multipoint (P2MP) LSPs are listed in the LSP table.
- Sites—A list of network sites.
- SRLGs—A list of the shared risk link groups (SRLGs); an SRLG is a group of objects that might all fail due to a common cause.
- AS—A list of both internal and external autonomous systems (ASes). An AS is a collection of connected IP routing prefixes that are controlled by one operator.

Figure 6: Common Table Functions



- Select any table tab to bring that table to the front.
- Right-click on any table tab to hide that table or add new tables.
- Right-click on any column (or click wheel icon) to hide that column or add new columns for that table, and to view other variables related to the column, such as average values.
- Click on any column to toggle its sort between ascending and descending order.
- Enter text or numbers in the Search field to find a matching string for that table.

Showing or Hiding Tables or Columns

Table 2: Show or Hide Tables or Columns

To...	Do This...
Show or hide one or more tables	<ol style="list-style-type: none"> Do one of these two steps: Choose View > Tables > Show/Hide Tables. Right-click any table tab and choose Show/Hide Tables. Toggle the tables you want to show to T (true) and the ones you want to hide to F (false). Click OK.
Hide a single table	Right-click any table tab and choose Hide this Table .
Show or hide one or more columns	<ol style="list-style-type: none"> Do one of these steps: Choose View > Tables > Show/Hide Columns. Right-click any column heading and choose Show/Hide Columns. Click the wheel icon in the table area. Toggle the columns you want to show to T (true) and the ones you want to hide to F (false). Click OK.
Hide a single column	Right-click any column heading and choose Hide this Column .

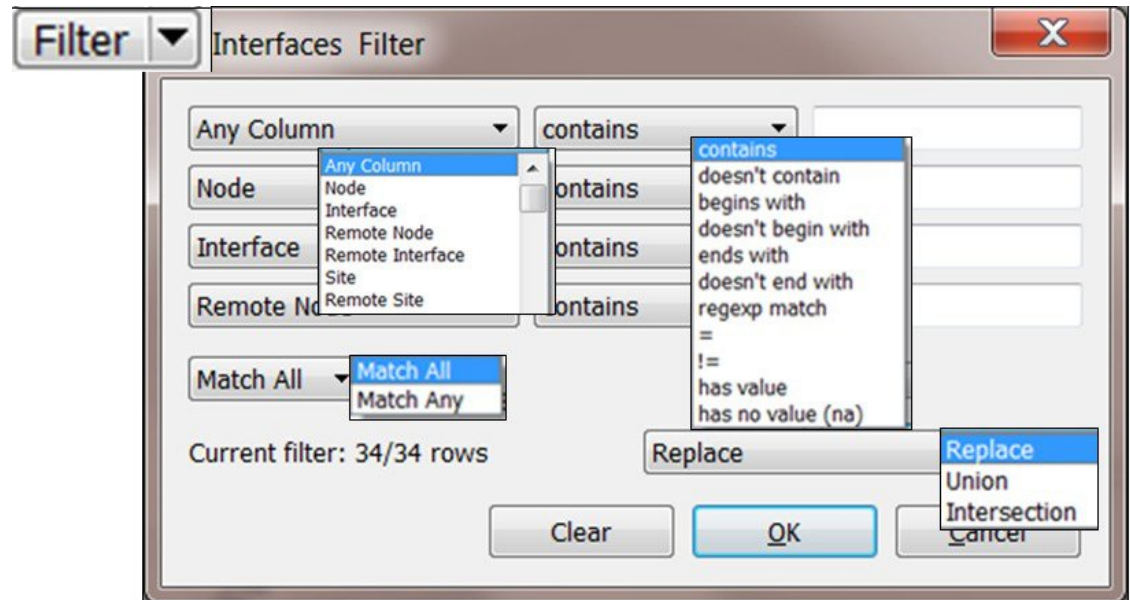
Searching and Filtering in Tables

Most WAE Design tables, whether below the network plot or with dialog boxes, have an empty Search field in the top right as shown in [Figure 6: Common Table Functions, on page 17](#). Likewise, they usually have a Filter control menu that opens a Filter dialog box specific to that table. [Figure 7: Interfaces Filter, on page 19](#) is an example of an Interfaces Filter dialog box.

Table 3: Table Search and Filter Features

To...	Do This...
Search for objects that match a specific string.	<ol style="list-style-type: none"> 1. If you are not in a dialog box, show the table whose objects you want to search. 2. In the Search field, enter a search string and press Enter.
Filter the table to a list of specific search values, and either replace, add to, or narrow the list of objects currently showing in a table.	<ol style="list-style-type: none"> 1. If you are not in a dialog box, show the table whose objects you want to filter. 2. Click the Filter control menu button (Figure 7: Interfaces Filter, on page 19). 3. Choose from the drop-down lists and enter search criteria. 4. If you are filtering by more than one criterion, you can optionally use these controls: <ul style="list-style-type: none"> Match All—(Default) Filters to only those rows that match all the specified criteria (default). Match Any—Filters to rows that match any one of the criteria. 5. Select whether to replace, add to, or narrow the list of objects currently showing in a table. <ul style="list-style-type: none"> Replace—Replaces objects showing in the table with those found in the search (default). Union—Adds the objects found to those showing in the table. Intersection—Filters to (search for) objects only within those objects that are currently showing in the table.
Filter the selection to associated objects. For example, you might want to filter a circuit to associated interfaces.	Choose Filter > Filter to X , where X is a name of a viable object.
Toggle whether to change the filter selection with or without first asking.	Choose Filter > Interaction with current filter > X , where X is either Always Ask or Replace .

Figure 7: Interfaces Filter



Saving Table Settings or Restoring Defaults

Table 4: Table Settings

To...	Do This...
Save table and column settings	Choose View > Tables > Save All Settings .
Save table settings	Do one of these steps: <ul style="list-style-type: none"> • Choose View > Tables > Show/Hide Tables. • Right-click any table tab and choose Save Table Settings.
Save column settings	<ol style="list-style-type: none"> 1. Show the table whose column settings you want to save. 2. Do one of these steps: <ul style="list-style-type: none"> Choose View > Tables > Show/Hide Columns. Right-click any table tab and choose Save Column Settings.
Restore all table and column settings to system defaults	Choose View > Tables > Reset All .
Resize column widths	<ol style="list-style-type: none"> 1. Show the table whose column settings you want to save. 2. To adjust the column width to that of the longest variable, right-click the column heading, and choose Adjust Width to Fit. To adjust the column width to any desired size, click and drag the column heading edge to any size.

To...	Do This...
Restore column widths to their original size	<ol style="list-style-type: none"> 1. Show the table whose column widths you want to restore. 2. Choose View > Tables > Reset Column Widths.

Other Table Actions

Table 5: Miscellaneous Table Tools

To...	Do This...
Show all objects for a table within the plan	<ol style="list-style-type: none"> 1. Show the table whose objects you want to see. 2. Click Show All.
Toggle the columns to sort in ascending or descending order	Click a column heading.
View the sum of the values in a column, the average, the minimum, and the maximum	Right-click a column heading.
View how many entries in a column are unique	Right-click a column heading.
Resize column widths	<ul style="list-style-type: none"> • To adjust the column width to that of the longest variable, right-click the column heading and choose Adjust Width to Fit. • To adjust the column width to any desired size, click and drag the column heading edge to any size.
Restore column widths to their original size	<ol style="list-style-type: none"> 1. Show the table whose column widths you want to restore. 2. Choose View > Tables > Reset Column Widths.
Detach the table from the plot	<p>Do one of these steps:</p> <ul style="list-style-type: none"> • Click and hold the empty gray area above the table while you drag it to a different location. • Click the double-page icon in the right of that same gray area.
Reattach the table to the plot	Click and hold the empty gray area above the table while you drag it to the bottom of the plot. When an empty blue area appears, release the mouse button.

Context Menus

WAE Design has a hidden menu called a *context menu*. Anywhere that you right-click, whether it is in an empty area in the plot, on an object in the plot, on a table tab, or on a column or row in a table, a context menu

appears with available options based on where you clicked. The context menu does not apply only in the main menus and in the Visualization toolbar.

The most common use case for context menus is for filtering to related objects. For example, to find the L1 circuits mapped to an L3 circuit, right-click the L3 circuit and choose **Filter to L1 Circuits**. You can select multiple objects and right-click just one to find the related information for all selected objects. If you want to see only selected objects in a table, right-click one of the selected objects and choose **Filter to Selections**.

If in a site, an interface, or a demand plot, the right-click actions taken affect the network plot.

Log and Report Windows

The following additional windows are displayed using menu items or after running tools:

- Log window—This window appears automatically when warnings or errors are logged. You can view these by choosing **Window > Log window**.
- Reports—Reports are generated through the Tools > Reports menu, as well as automatically when running certain tools such as Simulation Analysis. The report window appears at that time, and is accessible later by choosing **Window > Reports**. For more information, see [Reports, on page 379](#).



CHAPTER 3

Plan Objects

WAE Design networks consist of objects, such as nodes, sites, interfaces, circuits, SRLGs, ports, and port circuits. This chapter describes these basic objects and their relationships, as well as how to create, edit, and delete them.

Most objects are represented in the network plot, and all of them are represented in tables. Objects in the plot and in the associated tables have a set of properties that you can manage using a Properties dialog box. These object properties are represented by columns in the object's table, or by entries in tables of related objects. For example, all circuits have a Capacity property.

WAE Design has both a Layer 3 (L3) view and a Layer 1 (L1) view, each containing objects.

- Throughout this guide, the terms *node* and *circuit* refer to objects in the L3 view. When describing or contrasting both layers, the qualifier *L3* is used.
- The terms *L1 node* and *L1 circuit* refer to objects in the Layer 1 view.

This section contains the following topics:

- [Editing Objects Quick Reference, on page 23](#)
- [Understanding Nodes and Sites, on page 24](#)
- [Circuits and Interfaces, on page 31](#)
- [SRLGs, on page 35](#)
- [Ports, Port Circuits, and LAGs, on page 36](#)
- [Editing Object Properties, on page 39](#)
- [Tags, on page 40](#)

Editing Objects Quick Reference

This table is a simplification of basic object editing capabilities. Not all tasks apply to all objects.

Task	Select Object First?	Menu	Context Menu
Add	—	Insert	New
Duplicate	One object	Edit > Duplicate	Duplicate
Delete	One or more objects	Edit > Delete	Delete

Task	Select Object First?	Menu	Context Menu
Merge	One or more objects	Edit > Merge	Merge
Edit properties	One or more objects		Properties

Understanding Nodes and Sites

Both *node* and *site* are WAE Design terminology.

- **Node**—A device in the network, which can be one of three types: physical, PSN (pseudonode), or virtual. The Type property distinguishes whether the node represents a real device or an abstraction, such as a single node representing a number of edge nodes connected in the same way to the network. A physical node is a Layer 3 device, or router. Physical and virtual nodes behave in the same way within WAE Design. A pseudo-node (PSN) is typically used to represent a Layer 2 device or a LAN.

Nodes can reside both inside and outside of sites. The external arrangement could be useful for small networks where routers are not geographically dispersed.

- **Site**—A collection of nodes and/or other sites that potentially form a hierarchy of sites. Any site that contains other sites is called the *parent* site.

Both nodes and sites have simulated traffic, while nodes also have measured traffic.

- Nodes are shown as rectangles. The border color is an indication of traffic sourced from and destined to the node.
- Sites are shown as squares. The border color indicates the traffic utilization of all the nodes and circuits inside the site, including all nested nodes and circuits.

Note that sites can contain both L3 nodes and L1 nodes. Empty sites appear in both views, as do sites containing both L3 and L1 nodes. Only sites containing L3 nodes appear in the L3 view, and only sites containing L1 nodes appear in the L1 view. To change these defaults, see the *Cisco WAE Network Visualization Guide*.

Parent Sites and Contained Objects

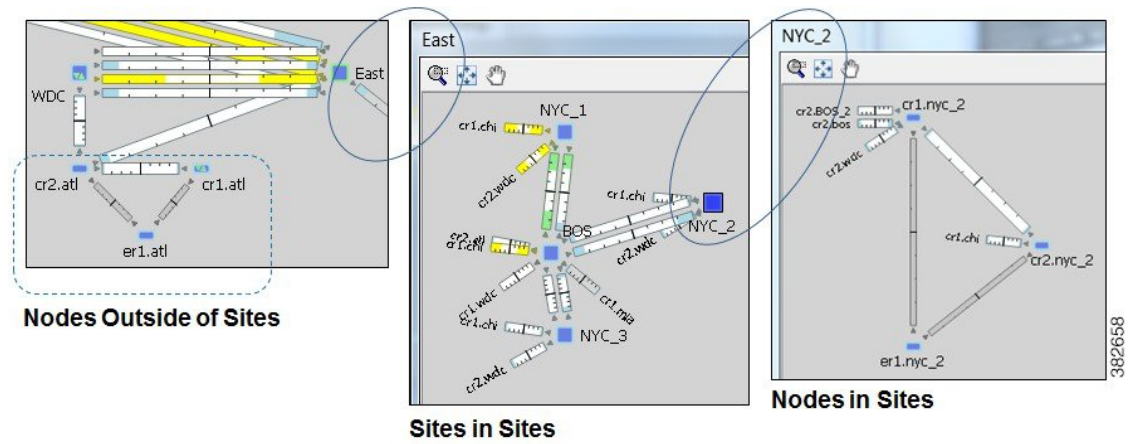
Unless a site is empty, there is a hierarchy of the sites and nodes that a site contains. A site can be both a parent and a child site. If a site contains another site, it is a *parent* site. Any nodes, sites, or circuits within a parent site are *contained* (*nested*) objects. The contained nodes and sites are also called *children*. Often the child nodes and sites are geographically co-located. For example, a site might be a PoP where the routers reside.

- The site's Parent Site property defines whether it is nested within another site. If it is empty, the site is not nested.
- In the network plot, the parent site shows all egress inter-site interfaces of all nodes contained within it, no matter how deeply the nodes are nested. Similarly, this is true for each child site plot within it.
- Selecting a site from the network plot does not select the sites or nodes under it.

- To filter to the sites or nodes contained within a site, right-click to access its “Filter to contained” menu. If you select “Direct,” WAE Design filters to only those objects contained immediately within it. If you select “All,” WAE Design filters to all objects contained within it, no matter how deeply they are nested.

Figure 8: Example of Node and Site Relationships, on page 25 shows an example of nodes outside of sites (atl nodes), sites containing other sites (the East site contains three NYC sites and one BOS site), and nodes within a site (NYC_2 site contains three nyc nodes). While this figure only shows one of the child site plots, if you were to expand all nested sites with the East site, you would find more nested child sites and nodes. As you can see in the network plot, the East site has 10 egress interfaces extending from it, some of which are within nested sites.

Figure 8: Example of Node and Site Relationships

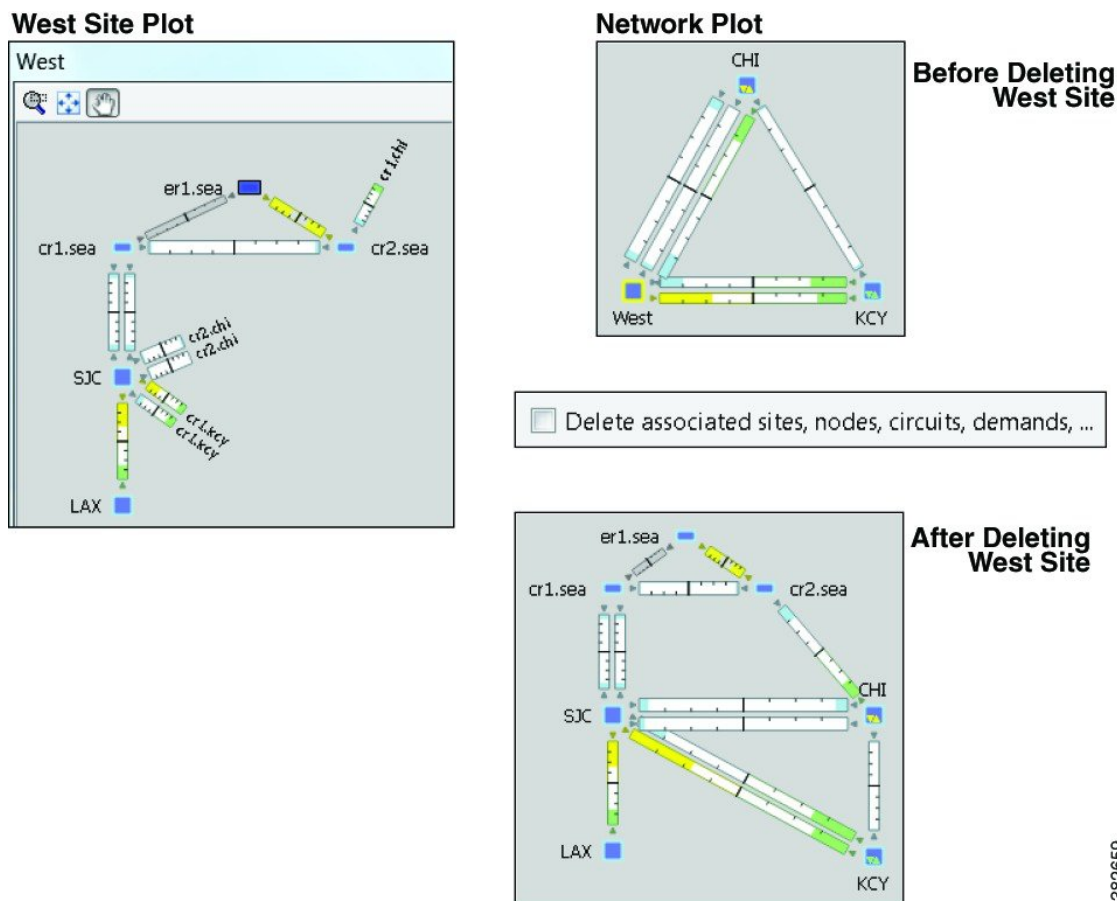


Deleting Sites

Selecting a site from the network plot does not select the sites or nodes under it except when deleting the site. In this case, all objects within a site are selected for deletion. However, in the confirmation that appears, you have the option to keep the contained sites and nodes. If you do, then the objects that are contained directly within it are moved to be on the same level as the site that is removed. The other, more deeply nested objects maintain their parent relationships.

Figure 9: Example Site Deletion, on page 26 shows a West site plot directly containing two sites (SJC and LAX) and three nodes. This parent West site resides in the network plot. Both SJC and LAX contain two sites each. By deleting the West site plot, but retaining its children, both SJC and LAX move to the network plot level, as do the three sea nodes. The sites within SJC and LAX remain as children sites to their respective parent sites.

Figure 9: Example Site Deletion



382659

PSN Nodes

WAE Design plan files can contain nodes of with a Type property of “psn.” These nodes represent pseudonodes (PSNs), which are used to model LANs or switches that connect together more than two routers. They are used in two situations: for IGP modeling and BGP peer modeling.

In an IGP network, a LAN interconnecting multiple routers is represented by a PSN node, with circuits connected to each of the nodes representing the interconnected routers. Both OSPF and IS-IS have a built-in system whereby one of the routers on that LAN is the designated router (DR) for OSPF or the designated intermediate system (DIS) for IS-IS. The PSN node is named after this designated router. WAE creates nodes with a property Type of psn automatically during IGP discovery.

When BGP peers are discovered, WAE might find that a router is connected to multiple peers using a single interface. This is typical at switched Internet Exchange Points (IXPs). WAE then creates a node with a property Type of psn, and connects all the peers to it, each on a different interface.

Following are a few points to consider when working with nodes that have psn as a Type property.

- Two PSNs cannot be connected by a circuit.
- If a PSN node is created by WAE, “psn” is prepended to the designated router’s node name.

- When creating demand meshes, WAE Design does not create demands with nodes of Type psn as sources or destinations. This is possible in manual demand creation, but not recommended. WAE Design sets the IGP metric for all egress interfaces from a node of Type psn to zero. This ensures that the presence of a PSN in a route does not add to the IGP length of the path.

Creating Nodes and Sites

Creating Nodes

To create a node, either choose **Insert > Node**, or right-click in an empty plot area and choose **New > Node**. Following are some of the frequently used fields:

- Name—Required unique name for the node.
- IP Address—Often the loopback address used for the router ID.
- Site—Name of the site in which the node exists. If left empty, the node resides in the network plot. This offers a convenient way to create a site while creating the node, move nodes from one site to another, or remove nodes from a site so it stands alone in the network plot.
- AS—Name of the AS in which this node resides, which identifies its routing policy. This can be left empty if no BGP is being simulated.
- BGP ID—IP address that is used for BGP.
- Function—Identifies whether this is a core or edge node.
- Type—The node type, which is physical, PSN, or virtual. Because a PSN node represents a Layer 2 device or a LAN, interfaces on a PSN must all have their IGP metrics set to zero, and two PSN nodes cannot be directly connected to one another. If you change a node type to PSN, WAE Design automatically changes the IGP metrics on its associated interfaces to zero.
- Longitude and Latitude—Geographic location of the node within the network plot. These values are relevant when using geographic backgrounds.
- X and Y—Location of the node within the network plot. These values are relevant when using schematic backgrounds.

Creating Sites

To create a site, either choose **Insert > Site**, or right-click in an empty plot area and choose **New > Site**. Following are some of the frequently used fields:

- Name—Required unique name for the site.
- Display Name—Site name that appears in the plot. If this field is empty, the Name entry is used.
- Parent Site—The site that immediately contains this site. If empty, the site is not contained within another one.
- Location—To automatically place a site in its correct geographic location and update the Longitude and Latitude fields, enter the airport code and press Enter. To select from a list of cities worldwide, press Alt-Enter.

- Longitude and Latitude—Geographic location of the site within the network plot. These values are relevant when using geographic backgrounds.
- Screen X and Y—Location of the site within the network plot. These values are relevant when using schematic backgrounds.

Duplicating Nodes

Duplicating a node creates an identical node and duplicates the connectivity of that node by creating new, identical circuits containing the same amount of traffic. By default, each demand and each LSP that uses the node as a source or destination is also duplicated. The newly created node is sequentially named the same. For example, if you duplicate a node named “chi,” WAE Design creates a chi[1] node. If you duplicate it again, a chi[2] node is created.

Step 1 Use one of these methods to open the Duplicate Node dialog box:

- Right-click a node and choose **Duplicate**.
- Select a node and choose **Edit > Duplicate**.

Step 2 Select whether to duplicate demands and LSPs, and then click **OK**.

Duplicating Sites

Duplicating a site creates an identical site, including the nodes within it and its connectivity. The default site name and display name are the same as the copied site plus a sequential number. For example, duplicating OKC creates OKC[1]. However, you have the option to change both the site name and the display name. The nodes that are created are likewise sequentially renamed. To change a duplicated node name, open the node’s Properties dialog box.

You cannot duplicate a site that contains other sites.

Step 1 Use one of these methods to open the Duplicate Site dialog box:

- Right-click a site and choose **Duplicate**.
- Select a site and choose **Edit > Duplicate**.

Step 2 If desired, rename the new site or display name, and click **OK**.

Merging Nodes

Real network topologies often have a number of nodes, typically edge nodes, connected to the network in the same way. For example, they might all be connected to the same core node or pair of core nodes. For planning and design of the network core, it is often desirable to merge these physical nodes into a single virtual node,

which simplifies the plan and accelerates the calculations and simulations performed. Note that merging nodes changes the plan itself, not just the visual representation.

The name of a newly merged node can be based on the site name, selected node name (base node), or have a new user-specified name. The node merge effects are as follows:

- Reattach circuits from other nodes to the base node.
- Move demands to or from other nodes to the base node.
- Move LSPs to or from other nodes to the base node.
- Set base node traffic measurements to the sum of measurements of the selected nodes.
- Delete other nodes.

Step 1 Select multiple nodes. If you do not select nodes, WAE Design merges all nodes.

Step 2 Choose **Initializers > Merge Nodes**, or right-click one of the nodes and choose **Merge**.

Step 3 Select whether to merge the nodes per site or merge them into one node.

- Separate merge per site—Merges nodes on a per-site basis. For example, if you selected all nodes in the plan, the result would be one merged node per site. If you do not specify a new suffix, the default name is the same as the site.
- Merge all nodes together into base node—Merges all nodes selected into one node. For example, if you selected two nodes in one site and three nodes in another, the result would be a single node in the site and node combination selected as the base node.

If you do not specify a new name, the default is to use the name of the base node.

Step 4 Click **OK**.

Trimming Nodes and Moving Their Traffic

A plan file created through WAE discovery often has more edge nodes in the plot than you are interested in viewing or simulating. These might be nodes that forward very little traffic or are singly connected to the core. However, you can reduce such a network model in WAE Design for simplicity and performance improvements by trimming (removing) these nodes. This process also removes all associated circuits, demands, and external endpoints.

Note that you can either select nodes to trim or you can choose to trim all nodes except those that are selected.

Typically, plan files are trimmed before demands are created. However, you can optionally trim demands in a plan file. This moves a demand that is sourced on a trimmed node to the first hop on the demand path in the remaining network. Likewise, a destination to a trimmed node is moved to the last hop of the demand path in the remaining network. If two or more demands in the same service class are trimmed so that their resulting source and destination nodes are equal, these demands are aggregated into one demand with their traffic summed. Demands split by ECMPs are converted into multiple demands, each with traffic divided proportionally to the ECMP split.

Trimming a node is not the same as deleting a node. When you trim a node, the measured traffic carried through that node is transferred to the nodes that remain so that the overall traffic in the network remains unchanged. If the node is deleted, this traffic is discarded. Only measured traffic is preserved in this way.

Note that multicast demands are removed, and cannot be trimmed.

Example: Edge node E is singly connected to node C in the core. When E is trimmed, the measured traffic on the interface from E to C is added to the total source traffic on node C. The measured traffic on the interface from C to E is added to the total destination traffic on node C. By default, demands to and from node E are removed.

Example: A demand source is node A, and destination is node E, and is routed through nodes A-B-C-D-E. If nodes A and E are trimmed and if you select the option to trim demands, then the new route for the demand is B-C-D.

-
- Step 1** (Optional) If you want to trim the traffic of nodes under a specific utilization, sort one of the Traffic columns to find the lowest to highest utilization. Alternatively, use the Nodes table filter to search for traffic of a certain level using, for example, the less than operator (<).
- Step 2** Select one or more nodes that you either want to trim or that you want to preserve while trimming nodes.
- Step 3** Right-click and choose **Trim**.
- Step 4** Select whether to trim the nodes or to keep the nodes and trim those that are unselected.
- Step 5** (Optional) Select the option to trim demands.

By default, the moved demands are named “Trimmed,” though you can enter a different name.

Step 6 Click **OK**.

Circuits and Interfaces

In WAE Design, an interface is either an individual logical interface or a LAG logical interface. If there is a one-to-one mapping between a logical and physical interface, then the interface contains both Layer 3 properties (for example, Metric) and physical properties (for example, Capacity). If there is a one-to-many mapping between logical and physical interfaces, then the interface is the logical LAG and the ports are included in the plan file as the physical ports in the LAG. For more information on ports and port circuits, see [Ports, Port Circuits, and LAGs, on page 36](#).

Each circuit connects a pair of interfaces on two different nodes. Therefore, an interface always has an associated circuit. Both the interface Properties dialog box and the circuit Properties dialog box let you simultaneously edit properties for the pair of interfaces and the circuit.

Interfaces have both measured and simulated traffic. The traffic that appears in the Circuits table is the higher of the traffic in the two interfaces.

Layer 1 Relationships

Each circuit can be connected directly to an L1 circuit. To make this connection, select the circuit, and either select or create the L1 circuit from the Advanced tab. An L3 circuit can be associated to multiple port circuits, and each port circuit can be associated with a different L1 circuit. Thus, L3 circuits can be associated indirectly with multiple L1 circuits. For more information, see [Ports, Port Circuits, and LAGs, on page 36](#) and [Layer 1 Simulation, on page 191](#).

The Latency and Distance Initializer uses mapped L1 circuit delays to calculate L3 circuit delays. If L1 circuits are not mapped to the L3 circuits, the node Longitude and Latitude properties are used in the calculation. For information, see [Simulation, on page 69](#).

Creating Circuits and Interfaces

To create a circuit, choose **Insert > Circuit**, or right-click in an empty plot area and choose **New > Circuit**.

Circuit Name:	cr1.chi-cr2.nyc		
Capacity:	10 GigE (10,000 Mbps)		
Delay:	8.37	Distance:	1630.77
SRLGs:	1		
Last Template Update:	yyyy/MM/dd hh:mm:ss UTC		
	<input type="checkbox"/> Protected	<input checked="" type="checkbox"/> Active	<input type="checkbox"/> Failed
Parallel Group Name:			

381163

As a result of creating a circuit, two interfaces are also created.

- Capacity—The amount of total traffic this circuit can carry. The drop-down list has a selection of the most widely used capacities.
- SRLGs—If you want this circuit to belong to an SRLG, select it from this list or create a new one by clicking **Edit**. See [Creating SRLGs for Circuits Only, on page 35](#).
- Parallel Group Name—To include this circuit in a new or existing parallel grouping, enter its name. To visually combine these parallel groups in the network plot, see the *Cisco WAE Network Visualization Guide*.
- Interface A and B—You must specify two interfaces that are connected by the circuit.

Interface A		Interface B	
Site:	bos	Site:	chi
Node:	cr1.bos	Node:	cr2.chi
Name:	{to_cr2.chi}	Name:	{to_cr1.bos}

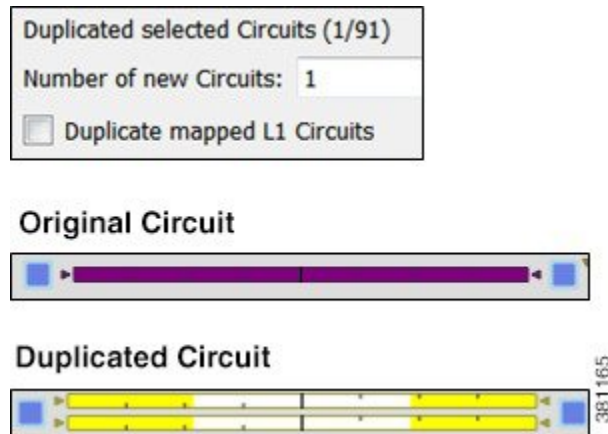
381162

Duplicating Circuits

Duplicating a circuit has the following effect on the new circuit and its interfaces:

- Duplicates the circuit itself and all its properties, including the circuit name.

- As a result, it duplicates both associated interfaces and renames them by adding an incremental numeric suffix to the original names, such as [1]. All other interface properties are copied to the new interfaces.



-
- Step 1** Select one or more circuits to duplicate.
- Step 2** Right-click one of these circuits and choose **Duplicate**.
- Step 3** Enter the number of new circuits you want to create for each selected circuit. For example, if you selected two circuits and you enter **3**, each of the selected circuits has three new parallel circuits, for a total of six new circuits.
- Step 4** To duplicate L1 circuits that are mapped to selected L3 circuits, check **Duplicate mapped L1 Circuits**.
- Step 5** Click **OK**.
-

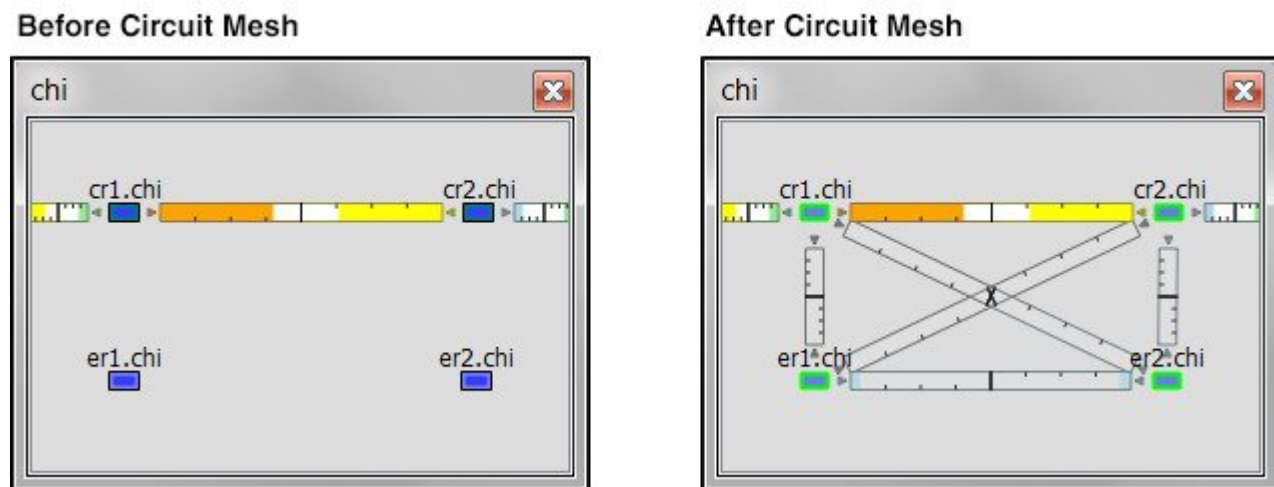
Creating a Circuit Mesh Between Nodes

To create a mesh of circuits between multiple nodes, follow these steps. [Figure 10: Example Circuit Mesh, on page 34](#) shows an example circuit mesh created between all the nodes within a site.

-
- Step 1** Select multiple nodes that you want to interconnect.
- Step 2** Right-click the interface you want to duplicate and choose **Duplicate Between Selected Nodes**.
- Step 3** Click **OK**.
-

What to do next

Figure 10: Example Circuit Mesh



Merging Circuits

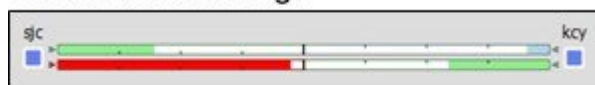
You can simplify the plan by merging circuits that have the same source and destination endpoints (nodes). This capability is useful, for example, in long-term capacity planning where multiple parallel circuits can be ignored and only the site-to-site connections are of interest.

Merging circuits changes the plan itself, not just the visual representation. (To change only the visual representation, see the *Cisco WAE Network Visualization Guide*.)

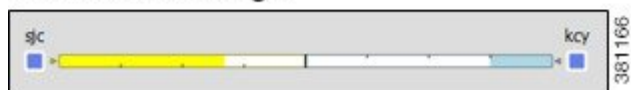
The circuit merge effects are as follows:

- Set base circuit capacity to the sum of capacities.
- Set base circuit metric to minimum of metrics.
- Set base traffic measurements to the sum of measurements.
- Delete other circuits.

Before Circuit Merge



After Circuit Merge



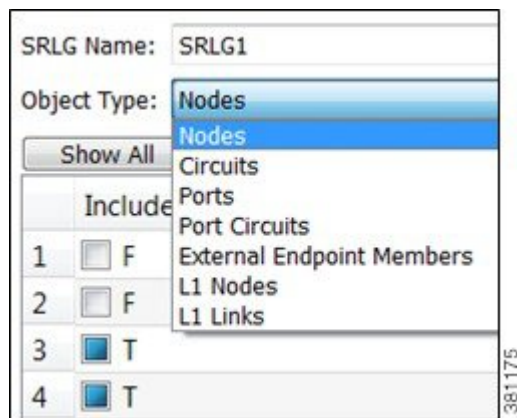
-
- Step 1** Select multiple circuits that have the same source and destination endpoints.
- Step 2** Choose **Initializers > Merge Circuits**, or right-click one of the circuits and choose **Merge**.
- Step 3** The name of a newly merged circuit is based on the selected circuit. If you want to base the name on a different circuit, select it from the Base Circuit list.
- Step 4** Click **OK**.
-

SRLGs

An SRLG is a group of objects that might all fail due to a common cause. For example, an SRLG could contain all the circuits whose interfaces belong to a common line card.

Creating SRLGs

- Step 1** To open the SRLG dialog box, do one of the following:
- Choose **Insert > SRLG**.
 - Right-click in the plot and choose **New > SRLG**.

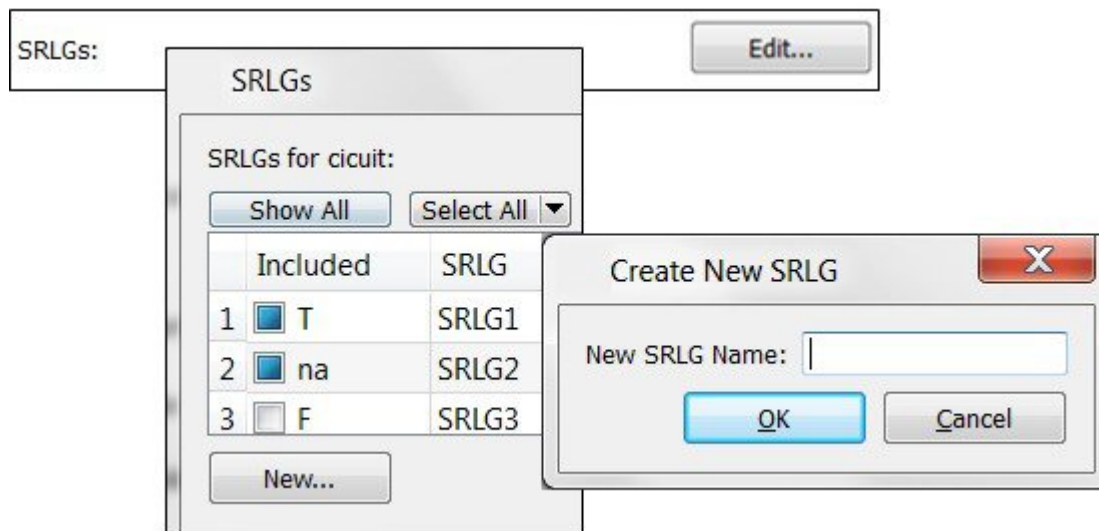


- Step 2** Enter the SRLG name, or keep the sequentially named default names (SRLG1, SRLG2, and so on).
- Step 3** From the Object Type drop-down list, choose the type of object that you want to include in the SRLG.
- Step 4** For each that object you want to include in the SRLG, check the **F** (false) check box to toggle it to **T** (true).
- Step 5** Repeat Step 3 and Step 4 for each object included, and then click **OK**.
-

Creating SRLGs for Circuits Only

- Step 1** Open the Circuits Properties dialog box.

- If you are creating SRLGs for selected circuits, select one or more circuits. Then double-click one of them. This is a very quick method of adding numerous circuits to an SRLG.
- If you are creating a new circuit, right-click in an empty plot area and choose **New > Circuit**. For more information, see [Creating Circuits and Interfaces, on page 31](#).



Step 2 Click the **Edit** button associated with the SRLG field.

Step 3 Associate the circuits with one or more existing SRLGs, or create a new SRLG.

Associate Circuits with Existing SRLGs	Create New SRLG
<p>For each SRLG in which you want to include the selected circuits, check the F (false) or “na” check box to toggle it to T (true).</p> <p>An “na” next to an SRLG means that one or more of the selected circuits are contained in that SRLG, but not all circuits are contained in it.</p>	<p>a. Click the New button.</p> <p>b. Enter the new SRLG name, and click OK.</p>

Step 4 Click **OK** in the SRLG dialog box to save the changes, and click **OK** again in the Properties dialog box.

Ports, Port Circuits, and LAGs

In WAE Design, a port is a physical interface. You can model link aggregation groups (LAGs) and port channels using WAE Design port and port circuits ([Figure 11: Ports, Port Circuits, and LAGs, on page 37](#)).

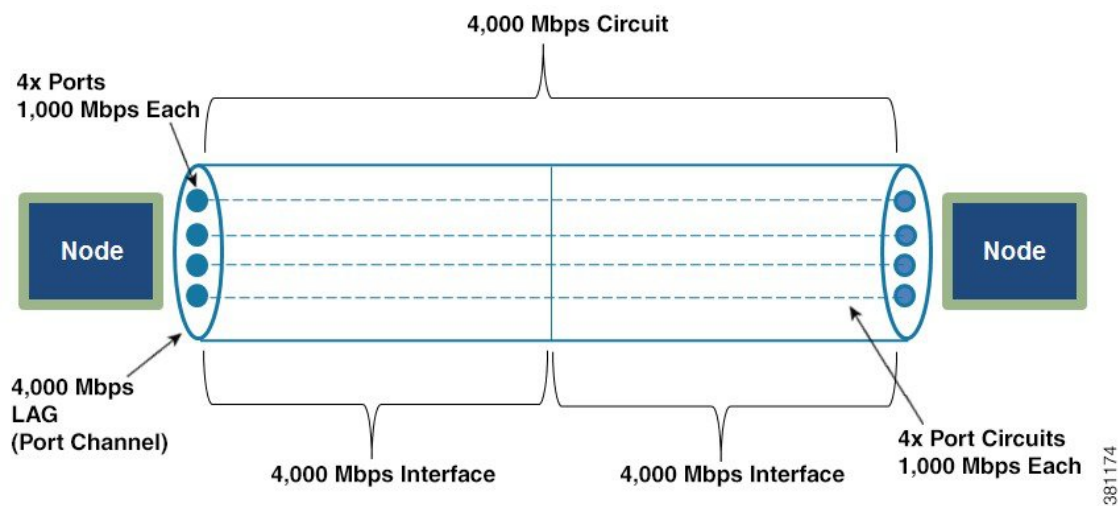
A LAG is a group of physical ports that are bundled into a single logical interface. A LAG is also known as *bundling* or *trunking*.

By default, each logical interface listed in the Interfaces table corresponds to a single physical port, and these ports need not be explicitly modeled. The exception is when the logical interface is a LAG, which bundles more than one physical port. In this case, the physical ports are listed in the Ports table.

A port circuit is a connection between two ports. However, ports are not required to be connected to other ports by port circuits.

Port circuits can be associated with L1 circuits, and ports can be connected to L1 nodes. The port circuit's Delay Sim value is copied from the associated L1 circuit's Delay Sim property. Otherwise, the Delay Sim value is set to 0. For information on L3 Delay and Delay Sim calculations, see [Simulation, on page 69](#). For information on L1 Delay and Delay Sim calculations, see [Layer 1 Simulation, on page 191](#).

Figure 11: Ports, Port Circuits, and LAGs



381174

Creating Ports

To create a port, either choose **Insert > Port**, or right-click in an empty plot area and choose **New > Port**. Following are some of the frequently used fields:

- Site and Node—Site and node on which this port exists.
- Interface—Logical interface to which this port is mapped. This must be defined to create port circuits using this port.
- Capacity—The amount of total traffic this port can carry. The drop-down list has a selection of the most widely used capacities.

Name:	port_er1.chi
Site:	chi
Node:	er1.chi
Interface:	{to_cr2.chi}
Description:	
Capacity:	GigE (1,000 Mbps)
MAC Address:	

381170

Layer 1 Relationships

Each port can be connected to an L1 node or to both an L1 node and an L1 port. Creating these connections populates the L1 Node and L1 port columns in the Ports table. If this port belongs to a port circuit, the Remote L1 Node and Remote L1 Port columns are updated, as applicable.

To connect the port to an L1 node or L1 port, select the Layer 1 tab. Then select the site (if applicable), the L1 node, and the L1 port to which the port is connected. For more information on Layer 1, see [Layer 1 Simulation, on page 191](#).

Creating Port Circuits

A port circuit specifies a pair of connected ports.

- Both of the ports must exist and be mapped to interfaces that are connected by a circuit.
- When selecting the two ports for the port circuit, note that if one is assigned to an interface, the other must be assigned to the remote interface on the same circuit.

To create ports and map them to their interfaces, see [Creating Ports, on page 37](#).

To create a port circuit, either choose **Insert > Port Circuit**, or right-click in an empty plot area and choose **New > Port Circuit**. The site, node, and port are all required fields.

PortA	
Site:	chi
Node:	er1.chi
Port:	port_er1.chi

PortB	
Site:	chi
Node:	cr2.chi
Port:	port_cr2.chi

381172

- Site and Node—Site and node on which the port exists.
- Port—Name of the port.

- **Capacity**—The amount of total traffic this port circuit can carry. The drop-down list has a selection of the most widely used capacities.

Each port circuit can be indirectly associated with an L1 circuit via direct mappings of L1 ports to L3 ports. For more information on Layer 1, see [Layer 1 Simulation, on page 191](#).

Creating LAGs

To make one of the existing interfaces into a LAG by assigning ports to it, follow these steps. If the interface does not contain ports, you must first create them (see [Creating Ports, on page 37](#)).

-
- Step 1** From the Ports table, select all ports that belong to the LAG. You can quickly do this by filtering to their common interface.
- Step 2** Double-click one of the ports to open the Properties dialog box.
- a) Select the interface that contains these ports.
 - b) Click **OK**.
-

Setting LAG Simulation Properties

Each interface is considered to be a LAG (port channel). You can configure LAG properties so that if it loses too much capacity due to non-operational ports, the entire LAG is taken down.

-
- Step 1** Double-click an interface or circuit; the Properties dialog box opens.
- Step 2** Click the **Advanced** tab.
- Step 3** In the Port Channel area, set one or both of the following parameters for one or both interfaces on the circuit:
- a) In the Min Number of Ports field, enter the minimum number of ports that must be active and operating for the LAG circuit to be up.
 - b) In the Min Capacity field, enter the minimum capacity that must be available for the LAG circuit to be up.
- Step 4** Click **OK**.
-

Editing Object Properties

Most objects in the plot and in the associated tables have a set of properties that you can manage using a Properties dialog box. These are the properties that WAE Design uses to define and simulate an object.

To change the properties of individual objects, do one of the following to open the Properties dialog box:

- Right-click one or more objects from the network plot or a table and choose **Properties**.
- Double-click one or more objects from the network plot or table. (The exception is that double-clicking a site from the plot opens a site plot.)

The Properties dialog box always lists how many objects were selected before opening the dialog box. For example, if the plan file includes 500 interfaces, and you have 342 of them selected, the notation is 342/500.

Tags

Many objects can be *tagged* as a way to group them for subsequent processing. Once created, tags appear as options in dialog boxes, enabling you to select and act upon all objects with a given tag name. Thus, tags provide a very flexible way to group interfaces, demands, nodes, LSPs, sites, and ASes that are not easily selected using standard attributes. For example, when using LSP paths you might want to tag selected paths to make it easier to assign or change their path affinities.

Adding tags to a parent site does not add tags to its children sites and nodes.



Adding Tags

-
- Step 1** Right-click one or more objects and choose **Properties**.
 - Step 2** Click the **Tags** tab and then **New Tag**.
 - Step 3** Enter a name that identifies the objects you are tagging (for example, Midwest Region), then click **OK**.
 - Step 4** Click **OK**.
-

Enabling and Disabling Tags

-
- Step 1** Right-click one or more objects and choose **Properties**.
 - Step 2** Click the **Tags** tab; a list of available tags opens.

Included		Tags
1	<input checked="" type="checkbox"/> T	Eastern Region
2	<input checked="" type="checkbox"/> T	Midwest Region
3	<input type="checkbox"/> F	Northern Region

Step 3 Toggle the tags to enable (**T** is true) or disable (**F** is false).

Step 4 Click **OK**.

What to do next



CHAPTER 4

Traffic Demand Modeling

WAE Design uses *demands* to describe the source and destination of a potential traffic flow across a network. A route simulation determines the routes that this traffic takes from the source to the destination, which are determined by the topology, the routing protocols, and the failure state of the network. To model IGP routing, these sources and destinations are nodes or interfaces within the topology. To model basic inter-AS routing, the sources and destinations are neighboring external ASes, peering nodes in these ASes, or interfaces to these peering nodes.

Each demand has a specified amount of traffic. There are several methods for putting this traffic into the demands, including the Demand Deduction tool, which calculates a realistic amount of per-demand traffic based on measured traffic.

The demand traffic is the basis for many of the WAE Design simulation and traffic engineering tools. An accurate set of demands and demand traffic is essential for effective planning, designing, engineering, and operating of a network. Accurate knowledge of the demands is essential for accurate traffic trending and traffic growth predictions.

This chapter describes demands, including how they are created and how their traffic can be estimated.

Use Demands To...	Suggested Steps to Take...
Model discovered networks	<ol style="list-style-type: none"> 1. Create a demand mesh based on where the traffic originates. For example, if all traffic is between edge routers, create a demand mesh between those edge routers. 2. Set the demand traffic by importing it or using Demand Deduction.
Model future usage in the network	<ol style="list-style-type: none"> 1. Create a demand mesh. 2. After setting the traffic, use WAE Design tools for growing the traffic and then analyze the effects on the network. You can import demand growth, you can modify selected demand traffic to emulate growth, or you can use demand groupings and other forecasting tools (described in Traffic Forecasting, on page 91).
Design networks	<ol style="list-style-type: none"> 1. Create a demand mesh. 2. Set the demand traffic using methods described in Traffic Forecasting, on page 91.

Use Demands To...	Suggested Steps to Take...
Analyze existing plans	<p>Use a variety of WAE Design tools that rely on demand traffic.</p> <ul style="list-style-type: none"> • Simulation Analysis lets you run what-if analysis. • Manually edit the network topology and configuration to determine the effects on traffic utilization. • Traffic Engineering tools let you control routings and optimize utilizations.

This section contains the following topics:

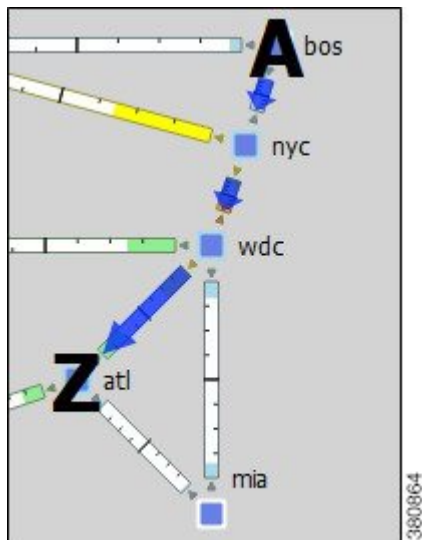
- [Demands, on page 44](#)
- [Demand Traffic, on page 50](#)
- [Creating Demands and Demand Meshes, on page 57](#)
- [Adding and Modifying Demand Traffic, on page 63](#)

Demands

Because demands determine how traffic is routed through the simulated WAE Design model, creating realistic demands and demand meshes is imperative to the accuracy of other information that can be derived from WAE Design. As such, all defaults are set to create demands and demand meshes that best suit most network models.

Each demand is comprised of unique properties (keys) that define it, other properties, and traffic. The following list summarizes these, though for a complete list of properties, refer to the available columns in the Demands table.

Selected demand paths are blue. An “A” labels the source, and a “Z” labels the destination.



Unique Properties (Keys)	Each demand is defined by a unique combination of these four properties.
	Name—By default, this is blank.
	Source—Nodes, interfaces, external ASes, or external endpoints.
	Destination—Nodes, interfaces, external ASes, external endpoints, or multicast destinations.
	Service class—User-defined classification of traffic, such as for voice or video.
Commonly Used Properties	Latency bound—Policy that sets the maximum permissible latency on a demand under normal operation. This property is used by WAE Design traffic engineering tools.
	Topology—Demands can be assigned to a specific IGP, and only route through interfaces that belong to that IGP.
	Reroutable—Enable/disable the routing of demands around failures. Turning off reroutes around failures might be useful, for example, when simulating Layer 2 traffic.
	Active—Only active demands are routed during simulations.
	Tags—User-defined label that lets you group demands for subsequent processing. Most tools, for example, allow you to choose a set of tags on which to operate.
Traffic	By default, demands have zero traffic, so you must add the simulated traffic to them.
	Demand traffic belongs to the service class of the demand.
	Demand traffic can be set per traffic level.

Demand Sources and Destinations

When creating sources and destinations, follow these recommendations:

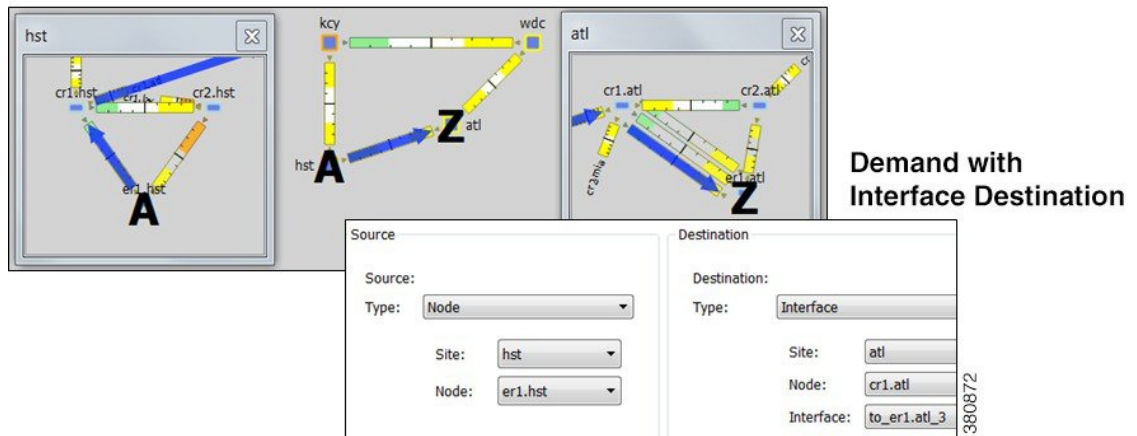
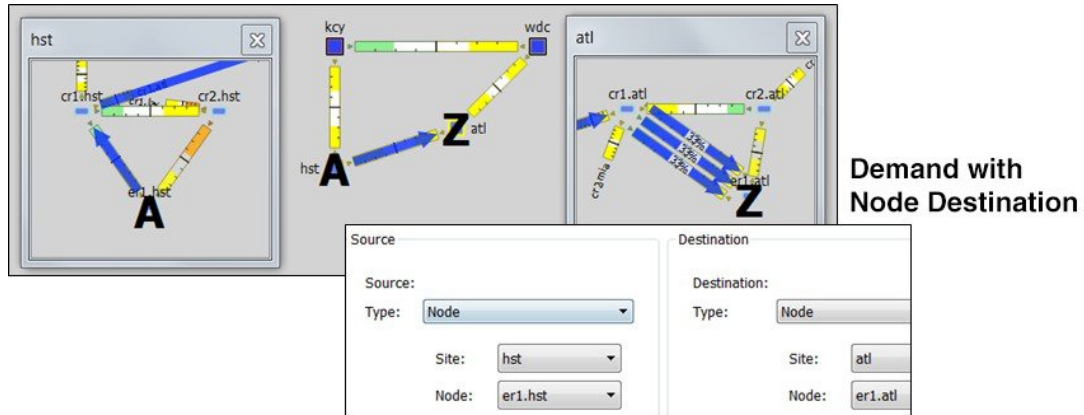
- For internal routing, use nodes.
- For external ASes, use a combination of ASes, nodes, and interfaces. Using interfaces lets you specify the exact interface on which the demand traffic is going into or out of a node.
- For more complex routing where multiple sources or destinations (and multiple failover scenarios) are required, use external endpoints.
- For multicast routing, use multicast destinations.

If multiple interfaces are attached to a node and if a demand is sourced to or destined for that node, the traffic splits across one or more of those interfaces, depending on other properties, such as IGP metrics or BGP policies (on a peering circuit). You can, however, specify just one of those interfaces. [Figure 12: Example of Demands Destined for a Node Versus an Interface, on page 46](#) shows an example of this difference between

using nodes and interfaces. In this example, the interface destination is one of three interfaces going into the er1.atl node.

Note if using an interface as a source of a demand, the source is the inbound interface. If using an interface as the destination of a demand, the destination is the outbound interface.

Figure 12: Example of Demands Destined for a Node Versus an Interface



380872

Demand Meshes

Demand meshes are a time-efficient way of creating numerous demands for all or part of the network. By default, WAE Design creates a source-destination mesh among nodes, interfaces, external ASes, and external endpoints. There are also advanced options, such as the ability to use a different set of destinations to create the demand meshes.

Alternatively, you can import an existing <DemandMesh> table that contains estimated traffic. WAE Design uses these estimations to calculate a full mesh of demands.

Demand Latency Bounds

Each demand can have a latency bound, which is a policy that sets the maximum permissible latency on a demand under normal operation. These can then be used to guide the route selection of the traffic engineering

tools. The Simulation Analysis tool can use these values to determine if latency bounds are violated when worse-case failures occur.

The Demands table has several Latency columns. Key ones are as follows:

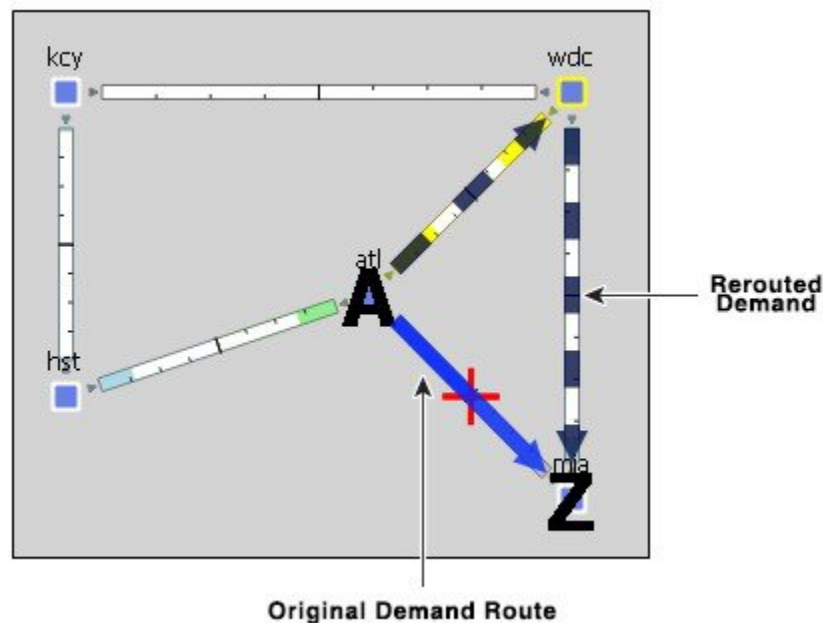
- Average Latency—Average latency over all ECMP subroutes.
- Minimum Latency—Minimum latency over all ECMP subroutes.
- Maximum Latency—Maximum latency over all ECMP subroutes.
- Latency Bound—Maximum permissible latency on a demand.
- Min Possible Latency—Total latency of the shortest path that the demand could take.
- % Diff Min Possible Latency—Maximum latency minus the minimum possible latency expressed as a percentage of Min Possible Latency.

Visualizing Demands

To view demand paths in the network plot, select them in the Demands table. Their path highlight is blue. An “A” labels the source, and a “Z” labels the destination. If sites are nested, these “A” and “Z” labels appear in all relevant child sites.

When you select demands in the Demands table, their associated L1 circuit paths are highlighted in the network plot.

Demands are most commonly used to show how traffic reroutes around failures. A dashed line shows the rerouted demand.



353484

Viewing Demand Plots

Demand routes can be complex because they can go through multiple sites that could hide the details of the path. For example, the network plot might not show the individual nodes traversed by the demand, as in [Figure 13: Network View of Selected Demand, on page 48](#). The demand plot offers an easier way to view end-to-end demands, letting you view the hops that the demands take without having to individually select sites to view them. For example, [Figure 14: Demand Plot View, on page 49](#) shows the demand plot view of the same selected demand as in [Figure 13: Network View of Selected Demand, on page 48](#).

- Step 1** In the Demand table, select one or more demands.
- Step 2** Right-click and choose **Plot Demands**.
- Step 3** To move the plot horizontally or vertically, use the left/right arrows and up/down arrows.
- Step 4** To select all demands within the demand plot view, click **Select Demands**. To deselect all of these demands, click in an empty plot area.
- Step 5** To visualize multilayer L1 paths when plotting demands, use the following drop-down options:

- **L3**—Displays the path in terms of interfaces.
- **L1**—Displays the path in terms of L1 links.

To better align the L1 links, WAE Design checks for L3-L1 links. If they do not exist, WAE Design checks for collocated L3/L1 nodes within sites.

- **L3+L1**—Displays the L3 plot above and the L1 plot below.

WAE Design does not perform alignment between L3 and L1 nodes. When you click an L3 interface, the associated L1 links are highlighted automatically.

What to do next

Figure 13: Network View of Selected Demand

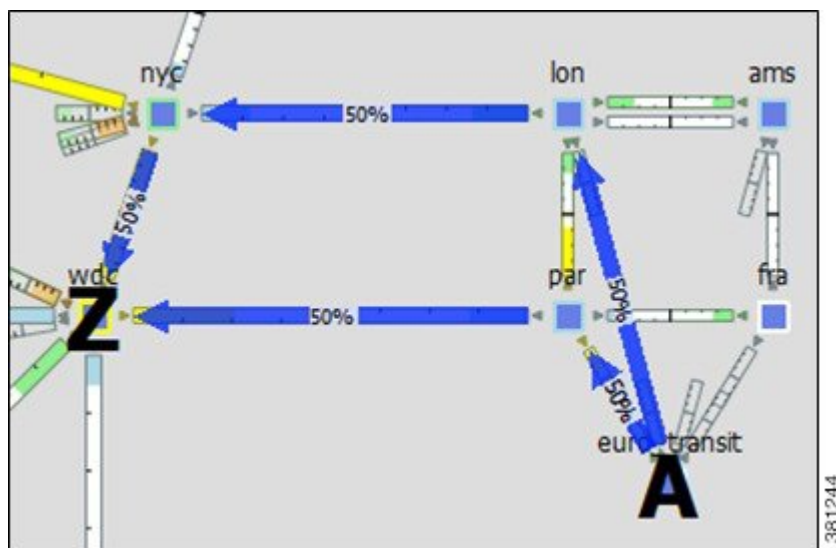


Figure 14: Demand Plot View



Viewing Shortest Paths and Shortest Latency Paths

For a description of shortest IGP paths and shortest latency paths, see [IGP Simulation](#), on page 105.

To View...	Choose...	Example
Shortest IGP path	View > Demands > Shortest IGP Paths	
Shortest latency path	View > Demands > Shortest Latency Paths	

Demand Traffic

Simulated traffic is the amount of traffic a demand is attempting to propagate through the network. For example, demand traffic is used to calculate interface utilizations during simulations. By default, demands have no traffic, and thus there is no simulated traffic. The most complex and powerful method of adding demand traffic is the Demand Deduction tool, which estimates demand traffic from measured traffic values.

Creating and Editing Traffic Levels

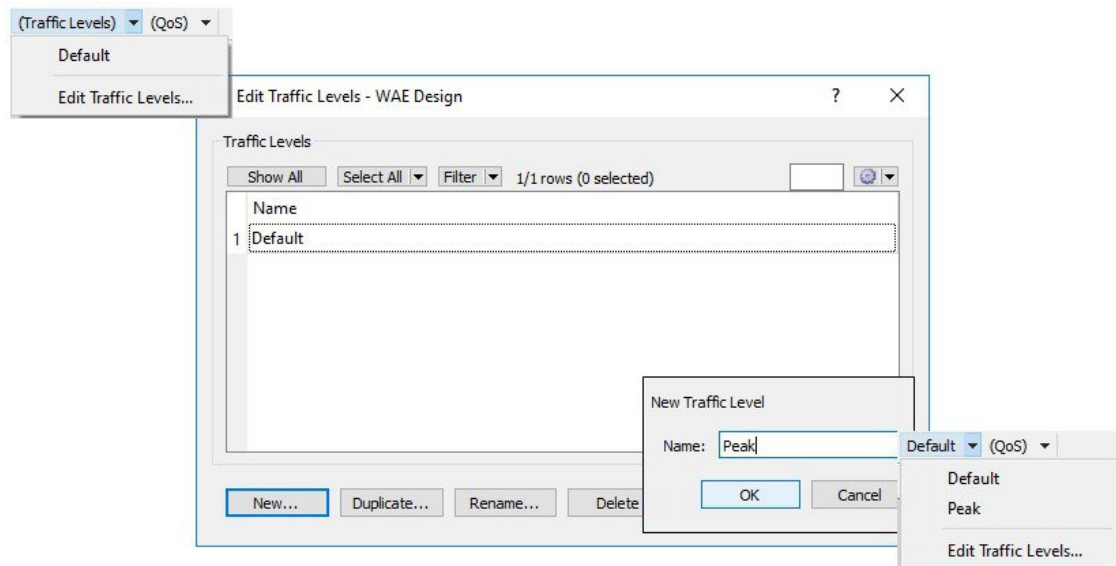
You can associate demands with specific traffic levels, which are identified in the Traffic Levels drop-down list in the Visualization toolbar. The traffic level displayed determines what is displayed in the plot and table calculations. For example, consider a totally static network that has no changes in topology or routing protocols. Throughout a day, the only thing that changes from one plan to the next is the traffic level. Thus, you can store traffic levels for different times of day all in a single plan file. Then, you can view the different states of the network from a single plan file by switching the view between different traffic levels.

You can create and edit traffic levels using the Edit Traffic Levels option ([Figure 15: Create and Edit Traffic Levels, on page 50](#)).

-
- Step 1** From the **Traffic Levels** drop-down list, choose **Edit Traffic Levels**.
 - Step 2** Click **New**, or choose an existing traffic level and click **Duplicate**.
 - Step 3** In the New Traffic Level dialog box, enter the name and click **OK**.
 - Step 4** Click **Done**. The traffic level now appears in the Traffic Level drop-down list.
-

What to do next

Figure 15: Create and Edit Traffic Levels



Demand Deduction

Plan files contain traffic measurements on the discovered network. Traffic can be measured on interfaces, interface queues, and RSVP LSPs, as well as on general traffic flows, such as from LDP LSPs. You can use Demand Deduction to estimate demand traffic based on any of these measurements.

The accuracy and usefulness of the results depend on many factors, including how much measured traffic is available, and of what type. For example, interface measurements are most often available, but LSP measurements might provide more information. The results also depend on the accuracy of the demand mesh and the routing model.

Typically, you only have interface traffic measurements. In this case, the individual demands estimated by Demand Deduction are not necessarily accurate. However, aggregates of demands can be highly accurate. For example, predicting the overall utilization after a failure, a topology change, or a metric change, can be very accurate even if the underlying demands individually are not reliable.

For more accuracy of individual demands, include point-to-point measurements, such as for RSVP LSPs or LDP flows measurements. Also, it is useful to combine different types of measurements together for use in Demand Deduction. Interface measurements are generally the most accurate measurements available, and if included in a Demand Deduction, can correct for missing or inaccurate LSP or flow measurements.

Note that you can also use Demand Deduction to set Traffic Balance (%) values for external endpoint members that are set to a Deduce Traffic type. See [Advanced Routing with External Endpoints, on page 265](#).

Example Demand Deduction

This example demonstrates results when using the Demand Deduction tool on a simple network. [Figure 16: Network Containing Two Demands and No Demand Traffic, on page 52](#) shows the routes of two demands in a network. These demands split between the two parallel core circuits due to an ECMP, and they have a common routing until the last hop. The Traffic column in the Demands table shows 0 because these demands do not yet contain traffic. [Figure 17: Measured Traffic View and Interfaces Associated with the Demands, on page 53](#) shows the Measured Traffic view and the five interfaces associated with the two demands, three of which have measured traffic.

- Edge1 to Core1 has 470 Mbps of measured traffic.
- One Core1 to Core2 interface has 210 Mbps, while the other has 240 Mbps, for a total of 460 Mbps. This unequal split is due to imperfect load balancing of the ECMP.
- There is no traffic from Core2 to Edge2 or from Core2 to Edge3.

Figure 16: Network Containing Two Demands and No Demand Traffic

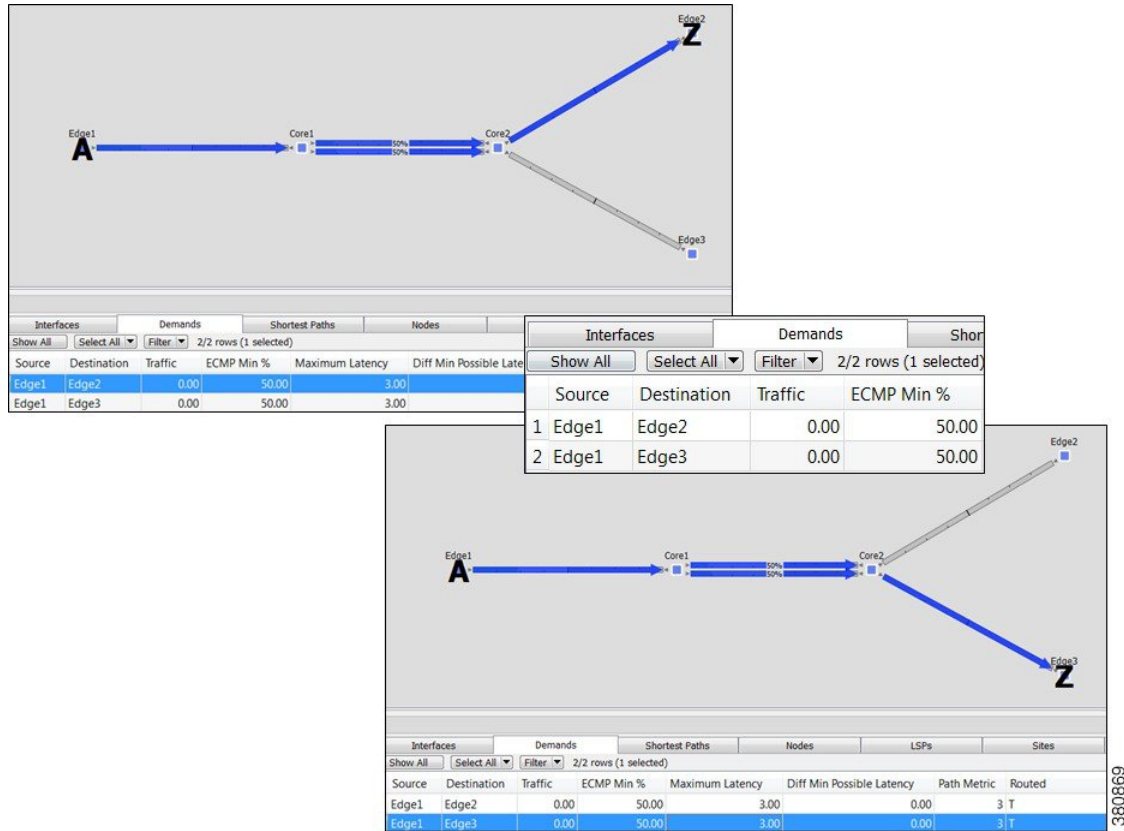
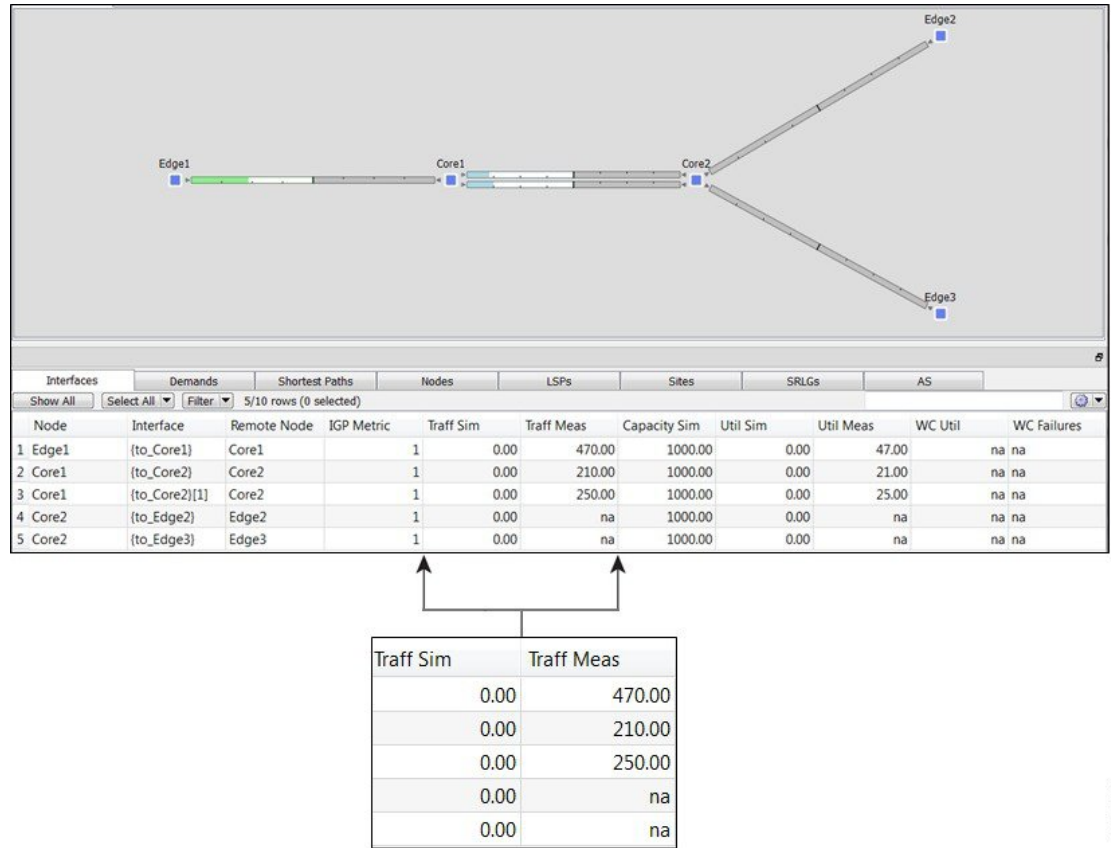


Figure 17: Measured Traffic View and Interfaces Associated with the Demands



Upon running Demand Deduction with its default options, the Simulated Traffic view appears. Other than the measured interface traffic, there is no other information about the demand traffic. So Demand Deduction first splits the difference between the measured 470 Mbps of traffic (Edge1 to Core1) and the measured traffic of 460 Mbps (Core1 to Core2) to get an estimated total demand traffic of 465 Mbps. In the absence of any other information, it divides this 465 equally to give 232.5 Mbps of traffic to each demand (Figure 18: Simulated View Showing Demand Traffic, on page 54). In the Interfaces table, the Traff Sim column now has values and the network plot shows simulated traffic percentages on all five interfaces associated with the demands.

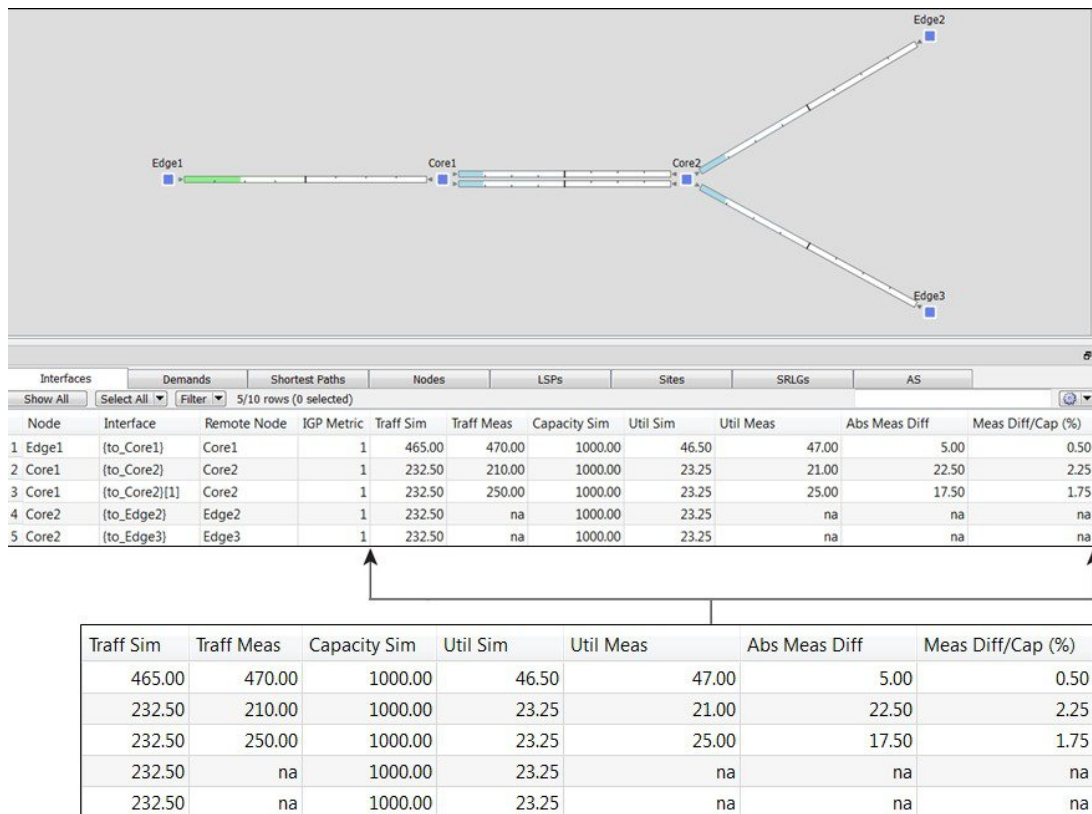
- Edge1 to Core1 has 465 Mbps of simulated traffic.
- Both Core1 to Core2 interfaces have 232.50 Mbps.
- Core2 to Edge2 and Core2 to Edge3 both have 232.50 Mbps.

The Abs Meas Diff and Meas Diff/Cap (%) columns in the Interfaces table show mismatches between measured and simulated values.

- Edge1 to Core1 has a difference of 5 Mbps, or 0.5%.
- One Core1 to Core2 has a difference of 22.5 Mbps, or 2.25%, while the other has a difference of 17.5%, or 1.75%.
- Neither the Core2 to Edge2, nor the Core2 to Edge3 interfaces have values because they had no measured traffic.

380870

Figure 18: Simulated View Showing Demand Traffic



In this same example, if the Core2 to Edge2 interface had 50 Mbps traffic, the results would have been different. Because this interface is used only by the one demand, the measured 50 Mbps of traffic would be used as an estimate only for that one demand. Using the same logic as before, the demands should total to 465 Mbps, so the other demand is set to the difference, which is 415 Mbps.

Differences in Measured and Simulated Traffic

Demand Deduction relies on accurate topologies, demand meshes, and traffic measurements. These can affect the results of the traffic simulated in the demands and cause simulated traffic to differ from the measured traffic, thus affecting the accuracy of WAE Design simulations. You can see how close these values are by showing the Abs Meas Diff and Meas Diff/Cap (%) columns in the Interfaces column.

Abs Meas Diff—The difference between measured traffic (Traff Meas) and simulated traffic (Traff Sim).

Meas Diff/Cap (%)—The absolute measured difference expressed as a percentage of capacity.

If these columns show large values, one of the following situations likely exists:

- Inaccurate measurements—Different measurements, for example of traffic through different interfaces, can be made at slightly different points in time. Fluctuations in traffic levels might take place between the times that measurements are being taken. This means that measurements could be inconsistent with one another. Usually, these inconsistencies are small and do not seriously affect the Demand Deduction results.

- Insufficient measurements—There are typically many more demands in a network than measurements, and many solutions will fit the observed data well. Demand Deduction chooses between possible solutions using knowledge of typical behavior of point-to-point traffic.
- Incorrect network configurations—If the network topology is incorrect in the plan file, the simulated routes would naturally be incorrect and measurements would not be adequately interpreted.
- Unbalanced ECMP—ECMP hashing can result in imperfect load balancing. Demand Deduction, however, distributes traffic evenly across ECMPs.
- Static routes—WAE Design does not model static routes. If these are present, demands routes might be simulated incorrectly, leading to deduction errors.
- Incomplete demand meshes—Demand meshes do not contain nodes even though traffic is routed between those nodes.
- Inappropriate priorities—In the Demand Deduction dialog box, you have the option to set the priority for calculations as 1 or 2. WAE Design first uses the measurements identified as Priority 1 to calculate the demands. Therefore, if the priority settings do not match the consistency of the traffic measurements in the network, the simulated traffic measurements will be less than optimal.

Additionally, Demand Deduction displays warnings for misleading or undesirable results:

- AS “(AS Name)” contains both dynamic LSPs and interface traffic. Interface traffic in AS has been ignored.

Routing of dynamic LSPs is nondeterministic. So it is not possible to make use of both measured interface traffic and measured dynamic LSP traffic for LSPs that may (or may not) traverse these interfaces. If the network contains an AS with both dynamic LSPs and interface traffic, this warning is issued and the interface traffic is not used.

- Some interface measurements exceed capacities by as much as (percent).

This warning is issued if a specified measurement exceeds the corresponding circuit capacity.

Minimizing Differences Between Measured and Simulated Traffic

Demand Deduction estimates demands that predict interface utilizations under incremental changes to the topology, for example failures, metric changes, or design changes, such as adding a new express route. If interface measurements alone are available, you might choose to fine-tune the Demand Deduction calculations to get better results, such as for site-to-site traffic. To enhance the accuracy of Demand Deduction results, consider the following suggestions:

- Include RSVP LSP or LDP measurements in the network discovery process.
- Restrict demand meshes to exclude demands that are known to be zero. For example, if you know that core nodes do not source traffic, then exclude core nodes when creating the demand mesh.
- Check the Nodes table to see if there is a node where the measured traffic going into it (Dest Traff Meas) and out of it (Source Traff Meas) are very different. Ensure these nodes are included in the demand mesh because they are either sources or destinations for traffic.
- In the Demand Deduction dialog box, always set the most consistent measurements to a Priority 1. The most reliable measurements are usually interface measurements. Likewise, LSP measurements are end-to-end, and thus also generally highly reliable. You can set multiple measurements to priority 1.

For example, if the flow measurements are inconsistent and the interface measurements are very consistent, then interfaces should be set to Priority 1 and the flow measurements to Priority 2.

- The Use Selected option in the Demand Deduction dialog box calculates demands for the selected rows in the Demand table. If you know beforehand the traffic for certain demands, you can set the traffic for those demands and then run Demand Deduction on just the unknown demands. Select the Use Selected—Fix Remaining option to fix the traffic of the remaining demands. A typical example is multicast demands, which often have a fixed, known bandwidth. (You can identify multicast demands in the Cast column of the Demands table.)
- If only a few measurements are available or if there are many inaccurate measurements, the tool sometimes estimates more traffic in a circuit than its capacity. To prevent this, in the Demand Deduction dialog box, select the option to keep the interface utilization below 100%. This forces the resulting simulated calculations to be below the given percentage of circuit capacity.

Flow Measurements in Demand Deduction

Besides node, interface, and LSP traffic measurements, Demand Deduction allows more general flow measurements to be used. These flow measurements can be flows from (or through) a specified node, to (or through) another node. Measurements can also be combinations of these node-to-node flows. This measurement format can be used to enter, for example, peer-to-peer flow measurements, or traffic measurements obtained from LDP routing or from NetFlow.

Flow measurements are entered in the plan file in the <Flows> table, and appear in the GUI in the Flows table. [Table 6: Flows Table Columns](#), on page 56 lists some of the more useful columns in the Flows table. Note that exactly which traffic is included is defined in the FromType and ToType columns.

Table 6: Flows Table Columns

Column	Description
From	Source node.
FromType	<ul style="list-style-type: none"> • Source—Traffic originating at the From node is included in the flow. • Interior—Traffic is included that passes through the From node, entering that node from another node in the same AS • Border—Traffic is included that passes through the From node, entering that node from another node in a different AS.
To	Destination node.
ToType	<ul style="list-style-type: none"> • Dest—Traffic that is destined for the To node. • Interior—Traffic that passes through the To node to another node in the same AS. • Border—Traffic that passes through the To node to another node in a different AS.
Traff Meas	Measured traffic used by Demand Deduction in its calculations. If more than one node is included in either the From or To columns, this measurement is the sum of the traffic over all flows between individual pairs of From and To nodes.

Creating Demands and Demand Meshes

Creating Demands

All selections and entries are optional except for identifying the source and destination.

Step 1 Choose **Insert > Demands > Demand**, or right-click in an empty plot area and choose **New > Demands > Demand**.

Step 2 Enter a demand name. The default is to create the demand without a name.

Step 3 Specify the source as a node, interface, external AS, or external endpoint.

Step 4 Specify the destination as a node, interface, external AS, external endpoint, or multicast group.

Step 5 Choose the service class. If there are no service classes, the demand operates on a service class named *Default*.

Service Class: Default

Latency Bound:

Topology: (No change)

Private LSP: (None)

Active Reroutable Require LSP

Step 6 Enter a value for the latency bound.

Step 7 Choose a topology to restrict the demand routes only to interfaces or LSPs belonging to that topology. The default is unrestricted routing.

Step 8 Retain the **Active** default to include the demand in WAE Design simulations, or uncheck **Active** to exclude this demand from simulations.

	Traffic Level	Traffic	Growth %
1	Default	na	
2	Morning	1300.00	
3	Peak	2020.00	

New Traffic Level...

380857

- Step 9** For each traffic level, specify the amount of traffic or leave it empty for Demand Deduction to complete. You can also create a traffic level if one does not exist (click **New Traffic Level**).
- Step 10** For each traffic level, specify the amount of growth rate you want to use for forecasting purposes. For more information, see [Traffic Forecasting, on page 91](#).
- Step 11** Click **OK**.

Duplicating Demands

- Step 1** Choose one or more demands.

Duplicate selected demands (3/96)

Number of new demands: 1

Share current traffic evenly

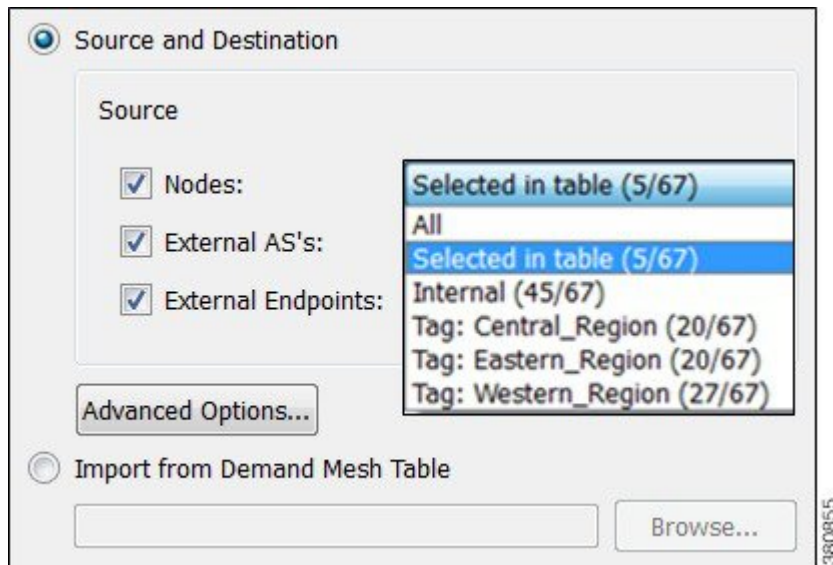
380867

- Step 2** Right-click one of them and choose **Duplicate**.
- Step 3** Enter the number of new demands you want to create per selected demand. For example, if you have three demands selected and you enter 2, each of the three demands is duplicated twice for a total of six new demands.
- Step 4** To evenly distribute the traffic of each demand being duplicated, select “Share current traffic evenly.” If this is unselected, the amount of traffic on the newly created demand is replicated. For example, if the original demand has 1000 Mbps of traffic and you share it with one newly created demand, the resulting traffic is 500 Mbps per demand. If you do not share it, both the original and the new demand have 1000 Mbps of traffic.
- Step 5** Click **OK**. Duplicated demands are named sequentially.

Creating Demand Meshes

Step 1

To restrict the demand mesh source to a specified set of nodes, external ASes, or external endpoints, select them before opening the Demand Mesh dialog box.



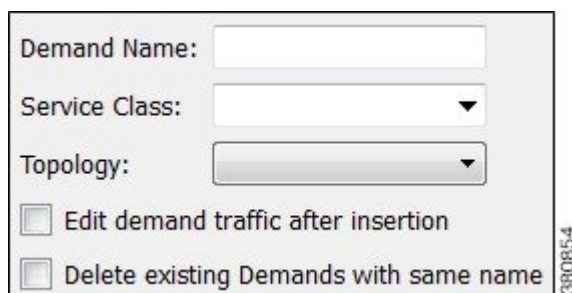
Step 2

Choose **Insert > Demands > Demand Mesh**, or right-click in the plot and choose **New > Demands > Demand Mesh**.

Step 3

Create a demand mesh based on the current plan file or import a Demand Mesh table.

- If creating a demand mesh based on the current plan file, select one or more sources from the Source drop-down lists. Options include using the selected objects (in Step 1) as the source, which is the default. Another option is to use all nodes, external ASes, or external endpoints as the source, which is the default if none were selected. The remaining option is to select the source based on tags. (Tags are set in an object's Properties dialog box.)
- If importing demands from a table, select the Import from Demand Mesh Table option. Then browse to the desired table. Note that this table specifies a mesh of sources and destinations, rather than creating demands. For information on how to create this table, see the *Cisco WAE Design Integration and Development Guide*.



Step 4

Enter a demand name. The default is to have no name to prevent large numbers of demands using the same name from being created. The names are useful if needing to identify a specific area of the network, such as a VPN. However, not using demand names helps ensure you do not create a large number of demands that all have the same name.

Step 5

Choose a service class. The default is the **Default** service class.

Step 6 Choose a topology. The default is that demands are applied to all topologies.

Step 7 If you want to create demand traffic immediately upon creating the demands, check **Edit demand traffic after insertion**. For information on entering traffic in the dialog box that appears, see [Generating Demand Traffic Using Node Traffic Estimates, on page 64](#). Note, however, if you have measured traffic in the plan file, a more accurate means of populating the demands with traffic is to use the Demand Deduction tool. For information, see [Demand Deduction, on page 51](#).

Step 8 If you want to delete all existing demands before new ones are created, check **Delete existing demands with same name**. The default (unselected) is to keep the existing demands and add only new ones.

Step 9 For any of the following options, click **Advanced Options**, make changes, and click **OK**.

- Destination—Choose this option if you want to create demands to destinations other than what has been selected as the source.

- Also create demands from destination to source—Uncheck this option if you want the demands created in only one direction. This applies only if you have selected a different set of destinations.
- Use interface endpoints to/from external AS nodes—When creating demands for external ASes, use a source/destination type of interface, and create a demand for all interfaces connected to each node in the external AS. For information on AS relationships and routing policies, see [BGP Simulation, on page 253](#).
- Respect AS Relationships—If checked, keep the existing AS relationships defined by the Routing Policy (default). If unchecked, recreate the AS relationships. The Routing Policy property is defined in the Edit AS Relationships dialog box. For information on AS relationships and routing policies, see [BGP Simulation, on page 253](#).
- Respect External Mesh Settings—If checked, keep the existing External Mesh settings defined for external AS meshes (default). If unchecked, recreate the external AS mesh. The External Mesh property is set in the AS Properties dialog box.
- Include Demands to Self—Creates demands that have the same source and destination node (default).

Step 10 Click **OK**.

Creating Demand Meshes for LSPs

Step 1 If applicable, choose LSPs over which you want to run the demands.

The screenshot shows a dialog box with the following elements:

- LSPs:** A dropdown menu with 'All' selected.
- Service Class:** A dropdown menu with 'Default' selected.
- Set Demand Traffic to:** A section containing three radio buttons:
 - LSP Setup BW
 - LSP traffic measurements
 - Zero
- Mark LSPs as private.

Step 2 Choose **Insert > LSPs > Demands for LSPs**, or right-click in the plot and choose **New > LSPs > Demands for LSPs**.

Step 3 Choose which LSPs to use: All, those selected, or tags.

Step 4 Choose the service class for the resulting demands.

Step 5 Choose the traffic for the newly created demands.

- Traffic equal to the LSP setup bandwidth
- Traffic equal to the LSP measurements
- Zero, which is appropriate if you need to insert demand traffic in other ways, such as using Demand Deduction, importing it, or manually modifying it.

Step 6 To remove the restriction of setting these demands to only these LSPs, uncheck **Mark LSPs as private**. Otherwise, the default is to restrict these LSPs so that they use only the resulting demands.

Step 7 Click **OK**.

Creating Demand Latency Bounds

You can set demand latency bounds to fixed values or to the following relative values ([Table 7: Editing Demand Latency Bounds](#), on page 62). All values are in ms (milliseconds).

- To set a fixed value or to remove demand latency bounds, use either the Properties dialog box or the Initializer tool.
- To set a relative value, use the Initializer tool.

Set latency bound for selected demands (95 / 95):

Remove

Fixed Value ms

Relative

value: % above shortest path

minimum: ms

maximum: ms above shortest path

380873

- **Minimum**—Minimum possible latency. Any latency bounds that are less than this value are set to exactly to it. The default is 20 ms.
 - **Maximum __ ms above shortest path**—Maximum possible latency. Any resulting latency bounds that are more than their minimum possible latency are set to exactly to this value above their minimum possible latency. The default is 35.
- For example, if the minimum possible latency is 20 ms and this maximum is 35 ms, if a demand's latency were to go up to 70 ms, it would be reset to 55 ms.
- **Value __ % above shortest path**—The latency bound is set to a percentage over the minimum possible latency for that demand. The default is 50%.

For example, if this value is set to 50% and if the minimum is 20 ms, then two demands with shortest paths by latency of 10 ms and 16 ms would have bounds set to 20 ms and 24 ms, respectively.

Table 7: Editing Demand Latency Bounds

Using Demands Properties Dialog Box	Using the Initializer Tool
<ol style="list-style-type: none"> 1. Select one or more demands in the Demands table. 2. Right-click a demand and choose Properties. 3. To set a fixed value for the latency bound, enter a value in the Latency Bound field. To delete a latency bound, delete the text in this field. 4. Click OK. 	<ol style="list-style-type: none"> 1. If applicable, select one or more demands in the Demands table. If you do not select demands, the result applies to all demands in the plan file. 2. Choose Initializers > Demand Latency Bounds. 3. You have several options: To delete latency bounds, click Remove. To set a fixed value, enter this number in the Fixed Value field. To set a relative value, click Relative and enter values. 4. Click OK.

Adding and Modifying Demand Traffic

Estimating Demand Traffic Using Demand Deduction

Demand Deduction calculates demand traffic when traffic measurements are available.

The options available can significantly affect the calculations. For information on improving accuracy of results, see [Minimizing Differences Between Measured and Simulated Traffic, on page 55](#). For information on setting up external endpoint members to be included in Demand Deduction calculations, see [Advanced Routing with External Endpoints, on page 265](#).

Demand Deduction

Estimate demand traffic based on measurements.

Traffic

Traffic Level: Selected: Default

Edit Traffic...

Measurements (measurement/total)

<input checked="" type="checkbox"/> Node Source and Dest	(66/66)	Priority 2
<input checked="" type="checkbox"/> Interfaces	(99/100)	Priority 1
<input type="checkbox"/> LSPs	(0/0)	Priority 2
<input type="checkbox"/> Flows	(0/0)	Priority 2

Fitting Parameters

Spread measurement errors evenly throughout network
(For operational use, once modelling is correct)

Concentrate errors in fewer places
(Slower, used for identifying measurement/modelling errors)

Keep interface utilization below %

Output

Demands: Use existing

Fix multicast demands

Remove demands with zero traffic

OK Cancel

Step 1 If applicable, choose one or more demands.

Step 2 Choose **Tools > Demand Deduction**.

Step 3 Choose the traffic level on which you are creating simulated traffic. The measurements used come from this traffic level, and the simulated traffic for each demand is placed in this traffic level.

- Step 4** If needed, modify traffic measurements for one or more demands by clicking the Edit Traffic button. In the dialog box that appears, you can modify measured traffic of interfaces, nodes, LSPs, flows, multicast flows, and/or ports. You can further change these measurements for a traffic level and interface queue.
- Another option in this dialog box is to enter growth percents for use with Forecasting tool. For information, see [Traffic Forecasting, on page 91](#).
- Step 5** Identify one or more types of measurements used in the calculations: nodes (source and destination), interfaces, LSPs, and flows.
- Step 6** For each type, set its priority. Select Priority 1 for high priority and Priority 2 for lower priority. You can have multiple measurements of the same priority. Like priorities are calculated simultaneously with equal consideration for the measurements.
- By default, all available measurements in the selected traffic set are used, and the interface measurements have priority over node, LSP, and flow measurements.
- Step 7** Choose the **Fitting Parameters**.
- Step 8** If you need to keep the traffic utilization below a different percentage than 100% (default), check **Keep interface utilization below** and enter a value.
- Step 9** Choose the demands for use in constructing the demand calculations.
- Use existing—Calculates demands using the existing demands only. This option is useful when simulating a pattern of demands that cannot be represented as a simple mesh between nodes. If you did not select one or more demands before opening this dialog box, use this option.
 - Use selected—Calculates demands for the selected rows in the Demand table. This option is helpful when you want to recalculate some of the demands, for example, such as a VPN submesh.
- Ignore Remaining—Does not include unselected demands in its calculations (that is, it treats those unselected demands as zero).
- Fix Remaining—Includes unselected demands in its calculations without changing their values. For example, if you select 6 out of 10 demands, then those 4 unselected demands would remain as they are, but would be included in the calculations.
- Step 10** Determine whether to fix multicast demands. If selected, the multicast demands are fixed at their current traffic value.
- Step 11** Determine whether to remove demands with zero traffic. The default is to remove them because Demand Deduction typically estimates a significant percentage of the simulated traffic to be zero when a large number of point-to-point utilizations in a mesh are extremely small. Using this default can substantially improve simulation and optimization performance in large plans. Do not remove demands with zero traffic if all demand routes are of interest, irrespective of traffic.
- Step 12** Verify your parameters, and click **OK**. Demand Deduction calculates the simulated traffic and lists the results in a Demand Deduction report.

Generating Demand Traffic Using Node Traffic Estimates

When a live network cannot be measured, you can estimate the demand traffic by editing the demand mesh. This is a simple way to create traffic in a mesh of demands by specifying the total source and destination traffic per node using one of the following algorithms. For traffic mesh options used in more sophisticated growth estimates, see [Traffic Forecasting, on page 91](#).

- Gravity Model—This estimation method allocates total traffic among the demands with roughly equal weights.
- Proportional to current traffic—This estimation method allocates total traffic among the demands in proportion to simulated traffic on existing demands.

After you manually enter total traffic data in the dialog box, you can export it to a file for later use.

Step 1 Open the Demand Mesh Traffic dialog box in one of two ways:

- Choose **Edit > Traffic > Demand Mesh**.
- When creating demand meshes, choose **Edit traffic after insertion** from the Insert Demand Mesh dialog box.

Step 2 Each node used in a demand is listed. The SrcDest column identifies whether it is a source, a destination, or both.

For each node, enter the traffic measurement on which you want the estimated demand traffic to be based. You can specify different values for source and destination traffic.

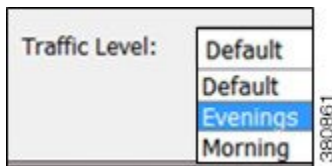
Step 3 You can set demand traffic based on an estimation model, a fixed value, or randomly based on a range.

- To base the demand traffic on an estimation model, click **Gravity Model** or **Proportional to current traffic**.
- To change the demand traffic to a fixed amount or on range of random values, follow these steps.



- Choose one or more demands.
- Right-click in a SrcTraffic or DestTraffic field, and choose **Set Value**.
- In the dialog box that appears, click **Fixed** or **Random**. If you choose Random, enter the minimum and maximum traffic values.
- Click **OK**.

Step 4 Choose the traffic level to which you want to apply the calculations, and then click **OK**.



Modifying Demand Traffic

You have numerous options for modifying demand traffic, all from the same dialog box. The changes that you make apply to the selected demands for the current traffic level.

You can modify demand traffic to either fixed values or to the following relative values.

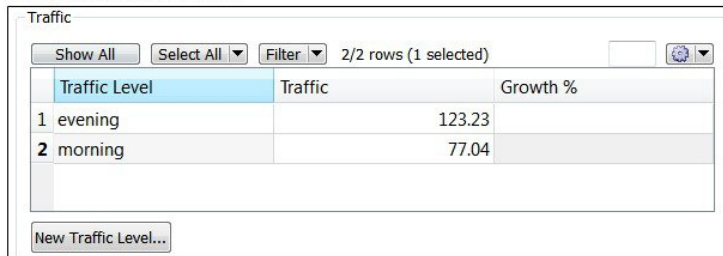
- To set a fixed value, use either the Properties or the Modify Demand Traffic dialog box ([Table 8: Modifying Demand Traffic](#), on page 67).
- To set a relative value, use the Modify Demand Traffic dialog box ([Figure 19: Modify Demand Traffic](#), on page 67 and [Table 8: Modifying Demand Traffic](#), on page 67). Following are the available options. Except for the percentage option, all values are in Mbps.
 - Change traffic by a specified percentage. Positive percentages add to the traffic, and negative percentages subtract. For example, if the traffic were 1000 Mbps and you entered –10, the traffic would be reduced to 900 Mbps.
 - Add a set amount of traffic spread over all demands in proportion to their current traffic. For example, if one demand had 1000 Mbps of traffic and the other had 2000 Mbps, and if you added 50 Mbps proportionally, one would have 1016.67 Mbps and the other would have 2033.33 Mbps.
 - Add a set amount of traffic uniformly to all the demands. For example, if one demand had 1000 Mbps of traffic and the other had 2000 Mbps, and if you added 50 Mbps uniformly, one would have 1025 Mbps and the other would have 2025 Mbps.
 - Set traffic to a fixed value.
 - Set traffic to a specific value that is spread proportionally over all demands. For example, if one demand had 1000 Mbps of traffic and the other had 2000 Mbps, and if you set them to 4000 Mbps proportionally, one would have 1333.33 Mbps and the other would have 2666.67 Mbps.
 - Set a specified amount of traffic, in Mbps, uniformly to all the demands. For example, if one demand had 1000 Mbps of traffic and the other had 2000 Mbps, and if you set them to 4000 Mbps uniformly, they would both be 2000 Mbps.

Table 8: Modifying Demand Traffic

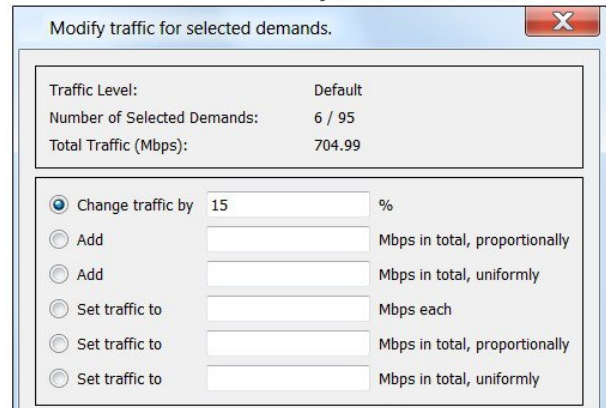
Modify Fixed Demand Traffic	Modify Fixed or Relative Demand Traffic
<ol style="list-style-type: none"> Choose one or more demands in the Demands table. Right-click one of these demands and choose Properties. For each applicable traffic level, enter the desired amount of simulated traffic in the Traffic field. If needed, click New Traffic Level to create a new traffic level. Click OK. 	<ol style="list-style-type: none"> Choose the traffic level you want the changes to apply to from the Traffic Level menu in the Visualization toolbar. If applicable, choose one or more demands in the Demands table. If you do not choose demands, the result applies to all demands in the plan file for the selected traffic level. Open the Modify Demand Traffic dialog box in one of two ways: Right-click a demand and choose Modify Demand Traffic. Choose Initializers > Modify Demand Traffic. Choose one of the options identified in the preceding list and enter a value accordingly. Click OK.

Figure 19: Modify Demand Traffic

Modify Fixed Values



Modify Fixed or Relative Values





CHAPTER 5

Simulation

WAE Design network simulations calculate demand routings and traffic distributions through the network based on the given traffic demand, network topology, configuration, and state. Simulation is the fundamental capability of WAE Design on which most of the other tools are built, including those for planning, traffic engineering, and worst-case failure analysis. A number of protocols and models are supported, including IGP, MPLS RSVP-TE, BGP, QoS, VPNs, Multicast, and Layer 1 (L1).

This chapter focuses on the general features of WAE Design simulations. The individual protocols and models are described in their respective chapters.

Simulations can be performed under different simulation convergence modes (Fast Reroute, IGP and LSP Reconvergence, and Autobandwidth Convergence), depending on which stage of the network recovery after failure is being investigated. If in Autobandwidth simulation mode, the results for simulations with failures are identical to those in the IGP and LSP Reconvergence simulation mode. The default simulation mode is IGP and LSP Reconvergence, and except where identified, the documentation describes this simulation mode. For more information, see [RSVP-TE Simulation, on page 135](#).

This section contains the following topics:

- [Traffic Simulation, on page 69](#)
- [State, on page 70](#)
- [Simulated Capacity, on page 75](#)
- [Simulated Delay and Distance, on page 76](#)

Traffic Simulation

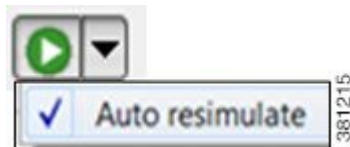
WAE Design uses demands to simulate traffic between source and destination endpoints within a network. Each demand has a specified amount of traffic.

By default, an up-to-date simulation of an open plan is always available. Any change in the plan that affects or invalidates the current simulation automatically triggers a resimulation. Types of changes that result in resimulations are those that would typically affect routing in a network.

- Changes in topology, such as adding and deleting objects or changing explicit paths.
- Changes to an object's state, such as failing an object or making it inactive.
- Changes in numerous properties, such as metrics, capacities, and delay.

To toggle automatic resimulation on and off, do one of the following:

- Click the drop-down arrow next to the Simulation On/Off icon (white arrow in green circle) in the Visualization toolbar. Check or uncheck **Auto resimulate** to toggle it on or off.



- Choose **Tools > Simulator > Switch On/Switch Off**.

State

The state of an object affects simulation and determines whether an object is operational.

- **Failed State**—Identifies whether the object is failed.
- **Active State**—Identifies whether the object is available for use in the simulated network. For example, an object might be unavailable because it has been set administratively down.
- **Operational State**—Identifies whether an object is operational. For example, an object might be non-operational because it is failed, is inactive, or because other objects on which it depends are not operational.

The Failed and Active columns in the tables show a visual representation of their status. Likewise, you can show the Operational column to show the calculated operational state. In each column, T means the object is in that state and F means it is not. Note that the T or F for Active state in the Interfaces table is reflective of the associated circuit. The plot shows graphical representations of these states with crosses of various colors and fills.

Failed State

The quickest way to see the effects of failures in a Simulated Traffic view is to have demands in place and then fail an object. The plot immediately displays where the traffic increases as a result ([Figure 20: Failed Circuit, on page 71](#)), and the Util Sim column in the Interfaces table reflects the traffic changes. Demands are then rerouted around the failure ([Figure 21: Reroute of Demand Around Failed Circuit, on page 72](#)).

To specify that a demand should not reroute around failures, uncheck the **Reroutable** check box in the demand's Properties dialog box. This can be used as a way of including L2 traffic on an interface. For example, a one-hop, non-reroutable demand can be constructed over the interface to represent the L2 traffic. Other reroutable demands can be constructed through the interface as usual. If the interface fails, the L2 traffic is removed and the L3 traffic reroutes.

You can simultaneously fail one or more objects of the same type. If you select an interface to fail, you are actually failing its associated circuit.

Objects That Can Be Failed	
<ul style="list-style-type: none"> • Circuits • Nodes • Sites • Ports • Port circuits 	<ul style="list-style-type: none"> • SRLGs • L1 links • L1 nodes • External endpoint members

Figure 20: Failed Circuit

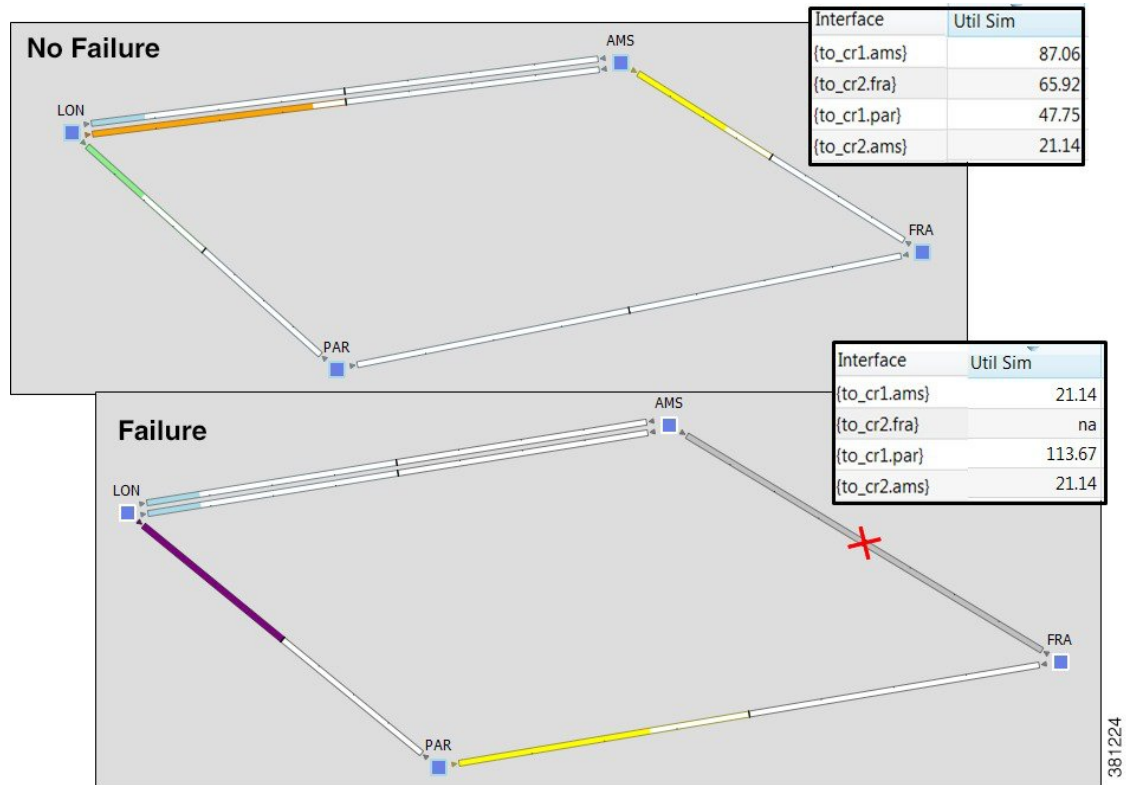
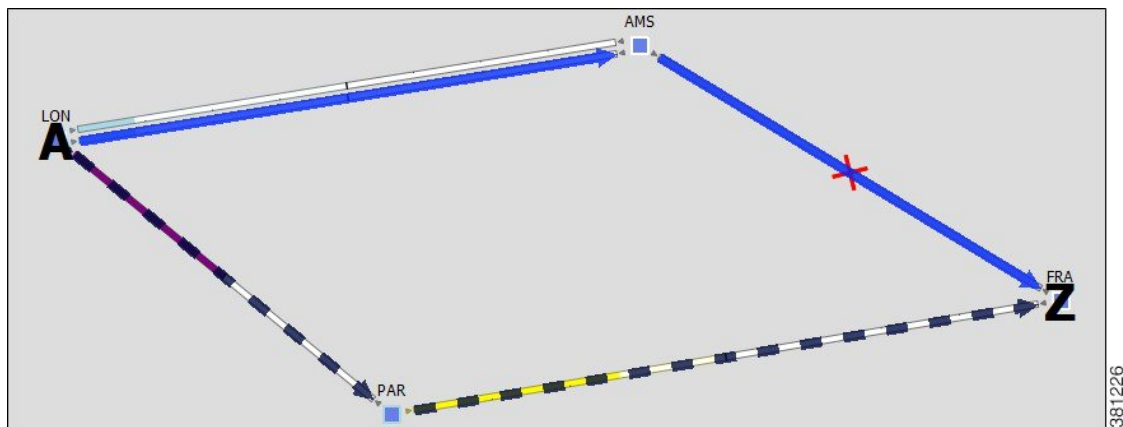


Figure 21: Reroute of Demand Around Failed Circuit



Failing and Recovering Objects

From the...	Fail Objects	Recover Objects
Context menu	<ol style="list-style-type: none"> 1. Select one or more objects. 2. Right-click one and choose Fail. 	<ol style="list-style-type: none"> 1. Right-click in an empty plot area. 2. Choose All to recover all failed objects or individually select an object to recover it.
Properties dialog box	<ol style="list-style-type: none"> 1. Select one or more objects. 2. Right-click one and choose Properties. 3. Check Failed and click OK. 	<ol style="list-style-type: none"> 1. Select one or more objects. 2. Right-click one and choose Properties. 3. Uncheck Failed and click OK.

Protecting Circuits Within SRLGs

You can protect circuits from being included in SRLG failures and SRLG worst-case analysis, though there are differences in behavior, as follows:

- These circuits do not fail if you individually fail an SRLG as described in the preceding steps. However, if you fail the circuit itself, it will fail.
- These circuits are protected from being included in Simulation Analysis regardless of whether they are in an SRLG.

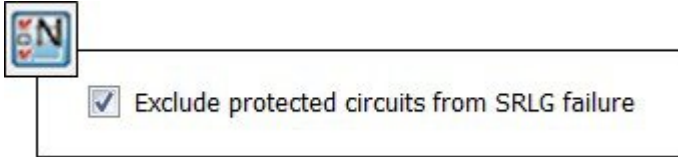
Note that this setting has no effect on how FRR SRLGs are routed. For information on FRR SRLGs, see [RSVP-TE Optimization, on page 323](#).

Step 1 Set the Protected property for circuits.

- a) Select one or more circuits.
- b) Right-click a selected circuit and choose **Properties**.
- c) Check **Protected** and click **OK**.

Step 2 Set the Network Option property for protecting circuits included in SRLGs.

- Choose **Edit > Network Options** or click the **Network Options** icon in the Visualization toolbar.
- Click the **Simulation** tab.
- Check **Exclude protected circuits from SRLG failure** and click **OK**.



Active State

An active/inactive state identifies whether an object is available for measured or simulated traffic calculations. An object can be inactive because:

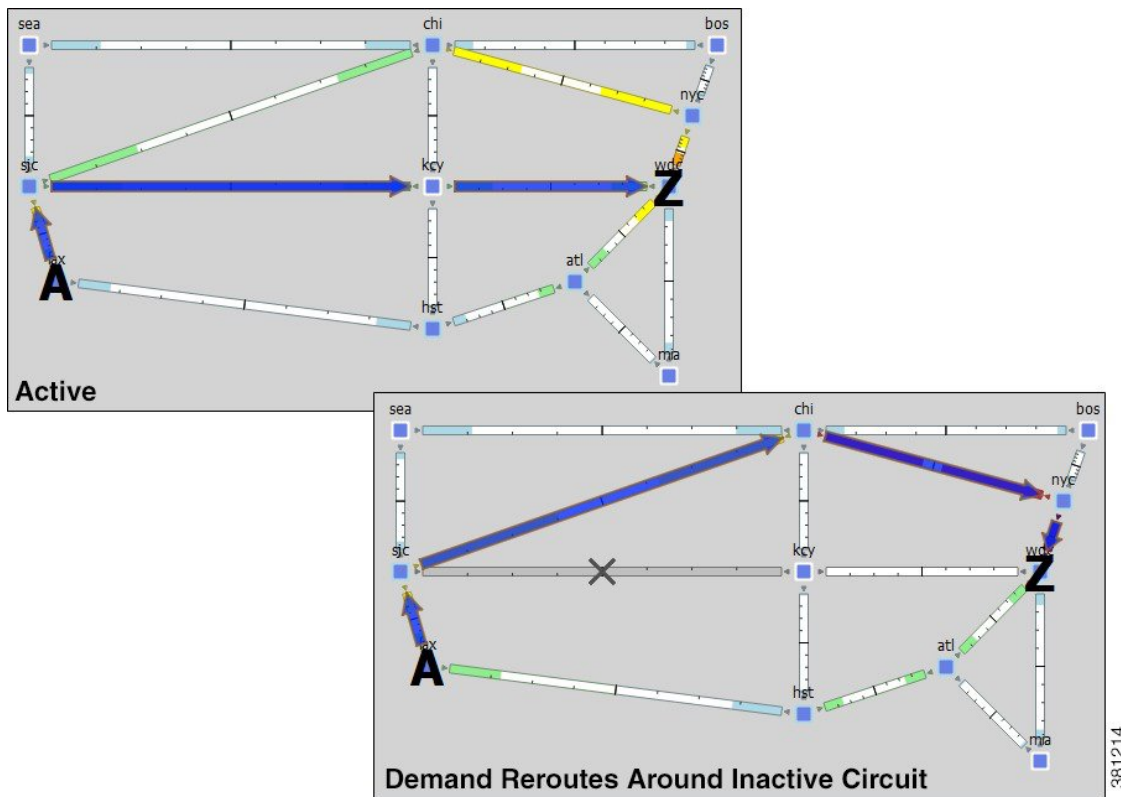
- It is administratively down.
- It is a placeholder. For example, you might be planning to install an object and want its representation in the network plot.
- It exists in a copied plan, but was not in the original plan that was discovered.

You can simultaneously change the active state of one or more objects. If you change the active state of an interface, you are actually changing its associated circuit.

Objects That Can Be Set to Inactive	
<ul style="list-style-type: none"> • Circuits • Nodes • Sites • Ports • Port circuits • SRLGs 	<ul style="list-style-type: none"> • L1 links • L1 nodes • External endpoint members • Demands • LSPs • LSP paths

Like failures, changing an object from active to inactive immediately affects how demands are routed, and affects the Util Sim column in the Interfaces tables ([Figure 22: Inactive Circuit, on page 74](#)).

Figure 22: Inactive Circuit



Setting Active or Inactive States

- Step 1** Right-click one or more like objects from their respective tables.
- Step 2** In the Properties dialog box, check the **Active** check box to toggle it on or off. A check mark means the object is active. Then click **OK**.

Operational State

The operational state identifies whether the object is functioning. You cannot set an operational state; rather, it is automatically calculated based on the failed and active states.

- Any object that is failed or inactive is operationally down.
- If the object relies on other objects to function, its operational state mirrors the state of those objects.

If This Object Fails or Is Inactive	These Objects Are Operationally Down
Node	Circuits connected to the failed node
Site	Sites, nodes, and circuits within the failed site

If This Object Fails or Is Inactive	These Objects Are Operationally Down
SRLG	Objects within the failed SRLG
Port	Port circuits that contain the failed port

Simulated Capacity

The Capacity column displays the configured physical capacity of interfaces, circuits, ports, and port circuits. Each circuit, port, and port circuit has a physical capacity that you can set in the Capacity field of the Properties dialog box. The interfaces have a configurable capacity that you can set in the Configured Capacity field. From these properties, a simulated capacity (Capacity Sim) is derived for each object.

The Capacity Sim column is the calculated capacity of the object given the state of the network, which could include failures that reduce capacity. All utilization figures in the Interfaces, Circuits, and Interface Queues tables are calculated based on this Capacity Sim value. When referencing the Capacity Sim value, there are a few rules to note regarding its calculation.

- If a circuit's Capacity is specified, this becomes the Capacity Sim of the circuit, and all other capacities (interface and constituent port capacities) are ignored. Specifying circuit capacities (rather than interface capacities) is the simplest way to modify existing capacities, which is useful, for example, for build-out planning.
- If the circuit has no Capacity, then its Capacity Sim is the minimum of its constituent interface Capacity values. The interface Capacity is the sum of the Capacity values of the associated ports. If the interface has no ports or if the ports have no Capacity, it is the same as the interface's Configured Capacity property. (Note the field in the interface Property dialog box is "Configured Capacity," while the column name in the Interfaces table is "Capacity.")
- If two ports are connected explicitly by a port circuit, the Capacity Sim of the port circuit is set to the minimum capacity of the three, which effectively negotiates down the capacity of each side of the connection.
- If two ports are connected implicitly by an L1 circuit, the Capacity Sim of the two ports is set to the minimum Capacity of the two, which again, effectively negotiates down the capacity of each side of the connection.
- In a LAG interface, if any of the constituent LAG members are operationally down, the interface Capacity Sim column displays a value that is reduced by the aggregate capacity of all the LAG members that are down. For example, if a 1000-Mbps port of a four-port 4000-Mbps LAG is operationally down, the simulated capacity for that LAG interface becomes 3000 Mbps.



Note If a pair of ports is considered in Capacity Sim calculations, both must be operational to be considered.

Simulated Delay and Distance

Both Delay and Distance are properties that can be set in the circuit Properties dialog box. To automate the process of setting these, use the Latency and Distance Initializer.

Delay:	<input type="text" value="11.8"/>	Distance:	<input type="text" value="9447.58"/>
--------	-----------------------------------	-----------	--------------------------------------

All WAE Design delay calculations that use the L3 circuit delay, such as Metric Optimization, use the Delay Sim value.

Circuits Table Column	Description
Delay	One-way transmission latency over the circuit in milliseconds (ms).
Distance	Distance of the circuit in kilometers (km).
Delay Sim	<ul style="list-style-type: none"> • (Derived) If the circuit Delay value is entered, it is copied to the Delay Sim column. • If no Delay is explicitly set, the Delay Sim column in the Circuits table is copied from the L1 Delay Sim column of the mapped L1 circuit (if one exists), which is based on the L1 circuit path that is currently the active path. • If no Delay is explicitly set and there is no mapped L1 circuit, the Delay Sim value in the Circuits table is copied from the maximum Delay Sim property of all associated port circuits. If none of these port circuits is mapped to an L1 circuit, then the Delay Sim value is set to 0. • If there is no circuit Delay value, no associated L1 circuits, and no associated port circuits with Delay Sim values, the Delay Sim value is 0.
Distance Sim	<ul style="list-style-type: none"> • (Derived) If the circuit Distance value is entered, it is copied to the Distance Sim column. • If no Distance is explicitly set for the circuit, the Distance Sim column in the Circuits table is copied from the L1 Distance Sim column of the mapped L1 circuit (if one exists), which is based on the L1 circuit path that is currently the active path. • If the Distance is not specified (na) and if there are no L1 circuits mapped to the circuit, the Distance Sim column is 0.

Latency and Distance Initializer

Following is the default behavior for the Latency and Distance Initializer. If there are no Longitude and Latitude available for use, the Distance and Delay properties are set to zero.

- L1 defaults

- The current L1 circuits Delay and Distance properties are removed unless **Keep existing distances if available** is checked.
 - The L1 node and L1 waypoint Longitude and Latitude properties are used to estimate the L1 distances. If an L1 node does not have these properties set, the shortest distance calculation is based these same properties of its immediate parent site, if applicable. (L1 nodes are not required to have parent sites.) The L1 link delays are calculated from these distances.
 - Each L1 circuit path's Distance Sim and Delay Sim columns are calculated as the sum of the distances and delays of L1 links over which the L1 circuit path is routed.
- L3 defaults
 - L3 circuit delays and distances are inherited from L1 circuits if there are L1 circuits mapped to the L3 circuits. Otherwise, the L3 delay and distance are estimated using the shortest distance between nodes at the endpoints of the circuits. If a node does not have Longitude and Latitude properties, the calculation is based on these same properties of its immediate parent site, if applicable. (Nodes are not required to have parent sites.) Thus, if you have an L1 model, the delays and distances are more precise because the physical paths that the L3 circuits take are known.
 - ITU G.826 adjustments are used to convert L3 circuit distance (d) to an adjusted circuit distance (r) as follows.
 - If $d < 1000$, set $r = 1.5 * d$
 - If $1000 \leq d \leq 1200$, set $r = 1500$
 - If $d > 1200$, set $r = 1.25 * d$
 If this option is unchecked, set r equal to d.
 - Both the L3 and L1 delays are calculated to be the speed of light along a distance r, multiplied by the speed of light in medium correction factor (between 0 and 1). The default for this correction factor is 0.67.

Step 1

This initializer can act on two different selected object types (L1 links and L3 circuits), so the defaults for the dialog box differ, depending both on selections and on how you access it. Regardless of the method, you can modify your selection options once the dialog box is opened. Following are guidelines for having the dialog box open with the most desirable defaults.

To Calculate...	Select These Objects	Choose
L1 link distance and delay, and L3 circuit delay	<ul style="list-style-type: none"> • No selections or <ul style="list-style-type: none"> • One or more L1 links and one or more L3 circuits 	Initializers > Latency and Distance.
L1 link distance and delay	<ul style="list-style-type: none"> • No selections or <ul style="list-style-type: none"> • One or more L1 links 	Right-click an L1 link and choose Initialize Latency.

To Calculate...	Select These Objects	Choose
L3 circuit delay	<ul style="list-style-type: none"> No selections or <ul style="list-style-type: none"> One or more L3 circuits 	Use one of these options: <ul style="list-style-type: none"> Right-click an L3 circuit and choose Initialize Latency. Right-click a circuit or interface and choose Properties. Click the Estimate button. This option is specific to L3 circuits.

Step 2 In each area, Layer 1 and Layer 3, verify or change your selection to act on the desired objects.

Estimate propagation delay and distance using longitude, latitude of nodes.

Layer 1

- Set distance (km) and delay (ms) on L1 Links
- Keep existing distances if available
- L1 Circuit Paths inherit distance and delay from Links

L1 Links

Layer 3

- Inherit delay and distance from mapped L1 Circuits
- If no mapped L1 Circuit, estimate delay and distance using shortest distance between nodes
- Use ITU G.826 adjustments

Circuits

Speed of light in medium correction factor (0-1):

381228

- For L1 links, the default is to delete distances. To keep them, check **Keep existing distances if available**.
- For L1 circuit paths, the default is to inherit distance and delay from L1 links. If desired, use the L1 Links drop-down list to base these calculations on all, selected, or tagged L1 links. If you do not want the distance and delay set, choose **None**.
- For L3 circuits, use the Circuits drop-down list to calculate L3 circuit delay and distance for all, selected, or tagged L3 circuits. If you do not want the delay or distance set, choose **None**.

Verify the inclusion ITU G.826 adjustments in the calculation, which is the default and which is usually left unchanged.

Step 3 For both L1 links and L3 circuits, verify or change the default speed of light factor of 0.67. This is usually left unchanged. Then click **OK**.



CHAPTER 6

Simulation Analysis

The Simulation Analysis tool combines the simulation results of a large set of failure scenarios. These results are useful for determining how vulnerable a network is to congestion and high latencies under failures, thus allowing you to plan sufficient capacity for any given failure scenario.

Simulation Analysis is run across a set of failure scenarios that include selected objects, such as circuits, and traffic levels. WAE Design calculates these failure scenarios across all service classes. Each scenario is simulated and results in the following available analyses, which vary depending on whether the network has QoS parameters and depending on which options are selected when running the simulation.

- [Worst-Case Traffic Utilization, on page 80](#) on interfaces per service class
- (Optional) VPN worst-case utilization and latency
- (Optional) [Worst-Case Demand Latency, on page 84](#)
- [Failure Impact, on page 85](#), which analyzes the impact that each failed object has on interface utilizations throughout the network

Upon completion, a report window opens with a summary of each analysis, along with the list of simulations performed. Each time you run a simulation, this information is updated (replaced).

After performing a Simulation Analysis across multiple failure sets, you can fine-tune the analysis for a subset of failure scenarios and a subset of service classes without running a new analysis. For example, if you run a simulation for nodes, circuits, and ports, you can later go back and view the results for any one of those three objects. See [Viewing Simulation Analysis on a Subset of Failure Scenarios or Service Classes, on page 89](#).

Simulation Analysis can be performed under different simulation convergence modes (Fast Reroute, IGP and LSP Reconvergence, Autobandwidth Convergence, and Autobandwidth Convergence Including Failures), depending on which stage of the network recovery after failure is being investigated. The default simulation mode is IGP and LSP Reconvergence, and except where identified, the documentation describes this simulation mode. For more information, see [RSVP-TE Optimization, on page 323](#).

This section contains the following topics:

- [Worst-Case Traffic Utilization, on page 80](#)
- [Worst-Case Demand Latency, on page 84](#)
- [Failure Impact, on page 85](#)
- [Simulation Analysis Reports, on page 86](#)
- [Protecting Objects, on page 87](#)
- [Running Simulation Analysis, on page 88](#)

Worst-Case Traffic Utilization

The default analysis is to identify up to 10 failures for the worst-case utilization on each interface in the network.

Worst case is the highest utilization that a particular interface experiences over all the failure sets and traffic levels that you selected. WAE Design also determines which combination of failures would cause this worst-case utilization.

Alternatively, you can record failures causing utilizations within a specified percent of the worst-case utilization.



Note To control the number of threads that WAE Design processes in parallel when examining failure scenarios, set the “Maximum number of threads” field.

Upon finishing the analysis, WAE Design switches to the Worst-Case Traffic view and updates the plot to simultaneously display all worst-case failures ([Figure 23: Worst-Case Traffic Utilization for All Interfaces, on page 81](#)). It also updates the following columns in the Interfaces and Circuits tables.

- **WC Util**—The worst-case utilization for that interface. The worst-case for a circuit is defined to be whichever of the worst cases of the two constituent interfaces results in the larger utilization. Thus, for circuits, this value is the larger of the WC Util values for the two interfaces in the circuit.
- **WC Traffic**—The actual traffic (Mbps) through the interface under the worst-case scenario.
- **WC Traff Level**—The traffic level under which this worst-case scenario occurs.
- **WC Failure**—List of one or more failures that cause the worst-case failure of the circuit. An easier way to read this list is to right-click an interface and choose **Fail to WC**.

Calculate worst-case utilization per interface

- Record failures causing utilizations within % of worst case
- Record up to failure scenarios per interface

Example: The circuit between cr1.lon to cr1.par, the cr1.lon node, and the L1 link between lon and par would all cause the worst-case utilization failure. `ct{cr1.lon|to_cr1.par}|cr1.par|{to_cr1.lon}};`
`nd{cr1.lon};L1lnk{lon|par|lon-par}`

If you record failures causing utilizations within a given percent of worst case, this column shows QoS violations as a percent. (See [Worst-Case QoS Violations, on page 81](#).) If the number is positive, then the allotted capacity has been surpassed. If negative, the capacity has not been surpassed. For example, if a circuit has 10,000 Mbps capacity, and if the amount of traffic on it as a result of three different failures is 11,000, 8000, and 4000 Mbps, the utilizations are 10%, -20%, and -60%, respectively and in descending order.

Calculate worst-case utilization per interface

- Record failures causing utilizations within % of worst case
- Record up to failure scenarios per interface

Example: The circuit between cr1.ams and cr2.lon is the worst-case failure and would cause this interface to exceed its capacity by 24.75%. The failure of the cr2.lon node would cause the interface traffic to go to its second highest utilization, which is 2.8% less than its capacity.

ct{cr1.ams|{to_cr2.lon}|cr2.lon|{to_cr1.ams}} (-24.75%);nd{cr2.lon} (2.8%)

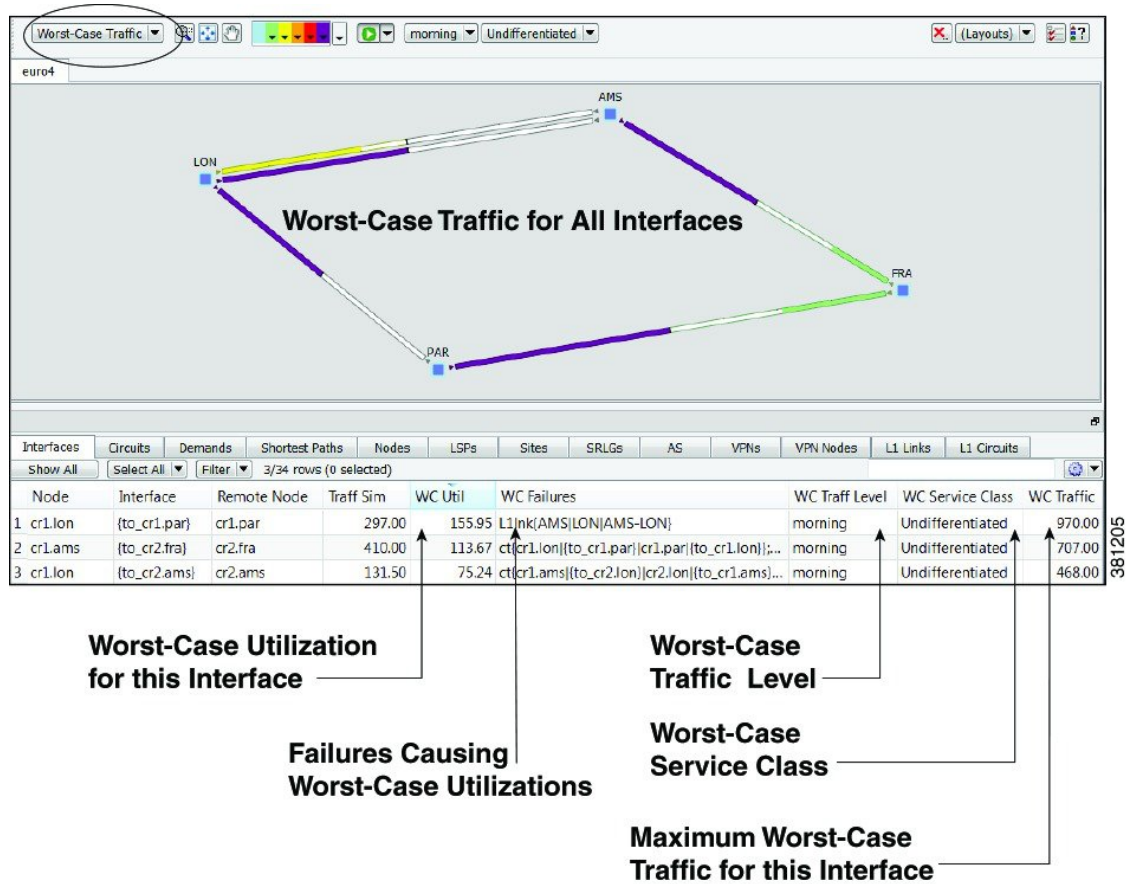
For information on reading notation of objects within tables, see the *Cisco WAE Design Integration and Development Guide*.

- WC Service Class—The service class for which this worst-case scenario occurs. For information on running a Simulation Analysis with QoS, see [Worst-Case QoS Violations, on page 81](#).



Note For information on worst-case calculations for VPNs, see [VPN Simulation, on page 279](#).

Figure 23: Worst-Case Traffic Utilization for All Interfaces



Worst-Case QoS Violations

WAE Design includes QoS bound (maximum available capacity) as part of the worst-case calculations. If there are no QoS parameters set, then the QoS bound is 100% and violations occur if utilization goes over

that 100%. However, if a worst-case policy has been set on a service class or if interface queue parameters have been set, then worst-case QoS violations are calculated. In these instances, WAE Design identifies the interface with the highest percentage of QoS violation as the worst-case possibility. The following columns are updated accordingly.

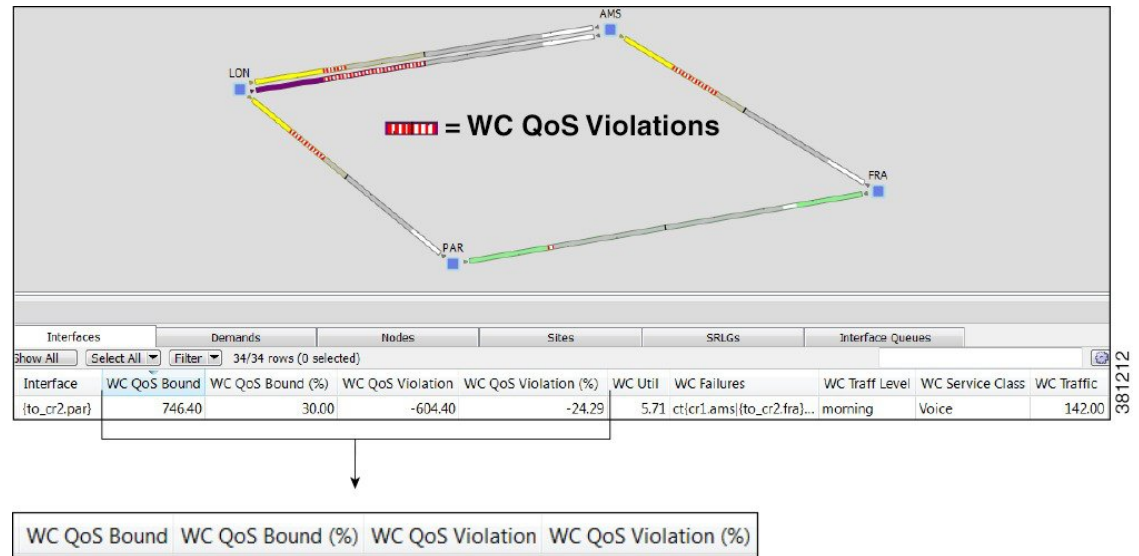
- **WC QoS Bound**—The worst-case interface capacity available without violating these QoS requirements. This value is based on available capacity, traffic utilization, worst-case policies set on service classes, and interface queue parameters. The **WC QoS Bound (%)** column identifies this same value as a percentage of the total capacity.
- **WC QoS Violation**—The worst-case traffic minus the worst-case capacity permitted (worst-case QoS bound). A violation occurs if the QoS capacity allotted through worst-case policies for service classes is exceeded or if QoS capacity allotted through interface queue parameters was exceeded. If the number appearing in the **WC QoS Violation** column is positive, then the allotted capacity has been surpassed. If negative, the capacity has not been surpassed. The **WC QoS Violation (%)** column identifies this same value as a percentage of total capacity.

To see the cause of worst-case QoS violations, right-click a circuit and choose **Fail to WC**. The table that appears lists all causes of this interface's worst-case utilization and its worst-case QoS violations. Choose the worst-case failure to view, and click **OK**.

- **WC Service Class**—The service contributing to the worse-case QoS violation.

For More Information...	See...
<ul style="list-style-type: none"> • QoS parameters and QoS calculations • Set worst-case policies on service classes • Set interface queue parameters 	Quality of Service Simulation, on page 239
Worst-case QoS calculations for VPNs	VPN Simulation, on page 279

Figure 24: Worst-Case QoS Violations



Failing Circuits to Worst-Case Utilization

You have the option to selectively view each failure scenario that causes the worst-case utilization or worst-case QoS violation for a single interface.

- Right-click an interface or a circuit and choose **Fail to WC**. If there is only one worst-case failure (listed in the WC Failure column), the object causing the worst-case utilization is listed. You can select it to fail it immediately.

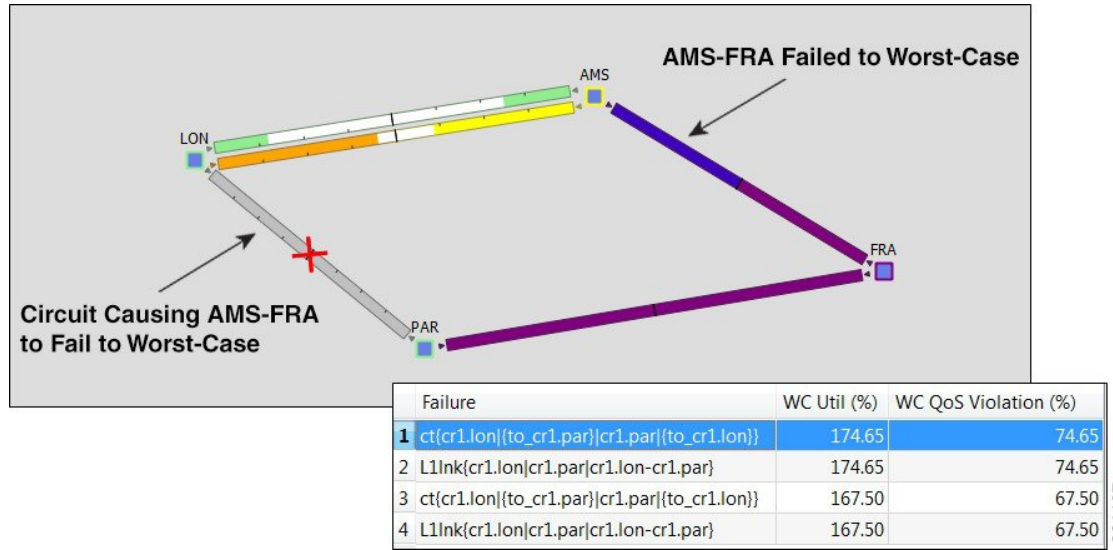
If there are multiple possibilities for a worst-case failure, or if there is a range of failures within a percentage of the worst-case failure, a Worst-Case Failures dialog box lists each failure, its worst-case utilization percent, and its QoS violation percent (Figure 25: Failure of Single Circuit to Its Worst-Case, on page 84). Choose the worst-case failure to view, and click **OK**. The network plot changes to show this particular failure scenario. If you choose to fail an L1 link or L1 node, switch to the L1 view to see the failure.



Note If you choose an interface, you are actually failing its associated circuit to its worst case.

- Alternatively, to filter to these worst-case failures for an interface without invoking a failure, right-click an interface or a circuit, choose **Filter to > Filter to WC**, and then choose the worst-case failure scenario of interest. If you fail this object from here, it achieves the same as if you had selected it from the Worst-Case Failures dialog box.

Figure 25: Failure of Single Circuit to Its Worst-Case



381197

Worst-Case Demand Latency

When running a Simulation Analysis, you have the option to simulate worst-case latency for each demand in the plan. WAE Design calculates the maximum latency of each demand under the failure scenarios selected. The result does not depend on service classes or traffic levels because demand routing is independent of these plans. The simulation also records the failures that cause this maximum latency.

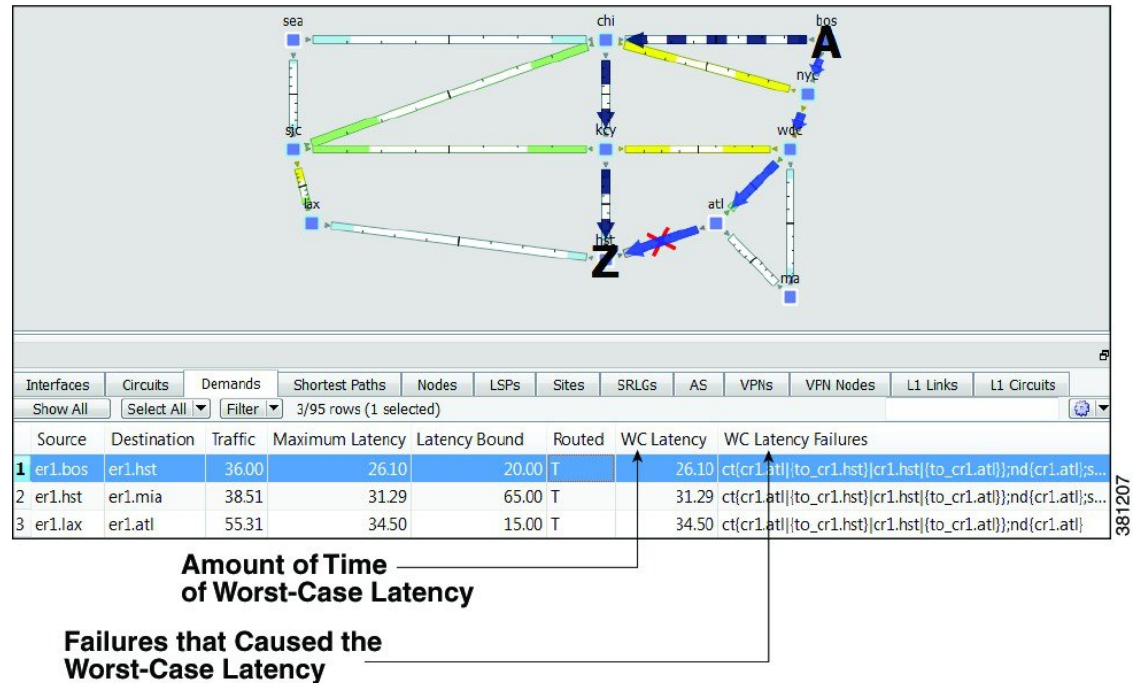
The following columns in the Demands table are updated when you check **Calculate demand worst-case latency** for Simulation Analysis:

- WC Latency—The highest demand latency over all failure scenarios in the analysis.
- WC Latency Failures—The failures that caused this worst-case latency. Up to 10 failures are identified.

Failing Demands to Worst-Case Latency

After running Simulation Analysis to calculate demand worst-case latency, you have the option to fail a single demand to its worst-case latency. Right-click the demand and choose **Fail to WC Latency** (Figure 26: Example of Worst-Case Demand Latency, on page 85).

Figure 26: Example of Worst-Case Demand Latency



Failure Impact

The Failure Impact view is available upon running a Simulation Analysis (Figure 27: Example Failure Impact, on page 86). The plot in this view colors the nodes and circuits according to the maximum utilization level that would be caused elsewhere in the network should the node or circuit fail. The color indicates the resulting utilization and severity of the congestion.

Example: In the Failure Impact view, a PAR-LON has a utilization of 90-100% and its color representation is red. This means that if PAR-LON were to fail, one or more interfaces would react by exceeding a 90% utilization level and correspondingly, would turn red in the plot.

The Node, Interface, and Circuit tables contain the Failure Impact and Failure Impact Interface columns. In the Interfaces table, the information describes the failure impact of the circuit containing the interface.

- Failure Impact—The failure impact of each node or circuit. For example, if the value is 80%, it means that if this node or circuit failed, the resulting traffic utilization on one or more interfaces would exceed 80%.
- Failure Impact Interface—The interface that will experience the highest utilization as a result of the node or circuit going down.

Format = if{Node|Interface}

Example: if{cr2.lon|to_cr1.ams} means if the circuit goes down, it will have the greatest traffic impact on the cr2.lon to cr1.ams interface.

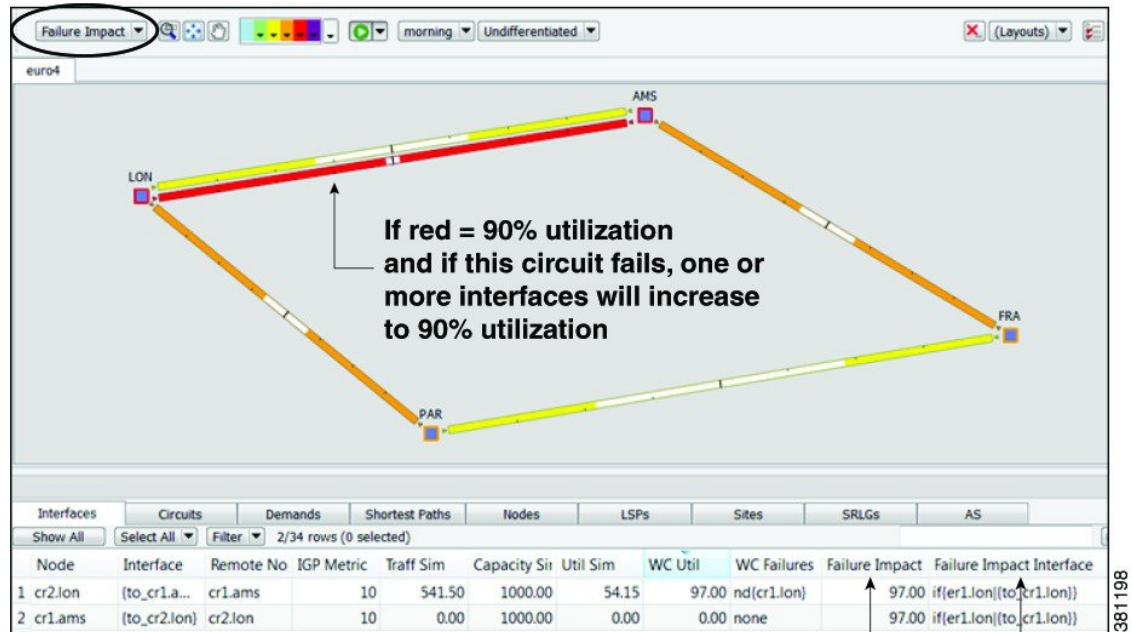
If you are showing only an L3 view, site borders show the maximum utilization level that would be caused elsewhere in the network should nodes within it fail or should intrasite circuits within it fail. If you are showing

a joint L3 and L1 view, site borders show the maximum utilization level that would be caused elsewhere in the network should a contained L3 node, L3 circuit, L1 node, or L1 link fail.



Note The Failure Impact view only shows the impact of circuit and node failures. It does not show failures of other objects.

Figure 27: Example Failure Impact



Highest utilization of the two associated interfaces as a result of the failure

Node and interface with highest utilization as a result of the failure

Simulation Analysis Reports

Each time Simulation Analysis is run, a report is automatically generated. You can access this information again by choosing **Window > Reports**. Note that new reports replace previous ones.

The Summary information details the options used in the analysis and summarizes the most important problems identified, such as QoS violations and latency bound violations.

The Max Util tab shows the impact of failures on maximum utilization in the form of a pie chart.

The Simulations table lists each simulation that was performed in the Simulation Analysis ([Table 9: Simulations Table in Simulation Analysis Report](#), on page 87).

Table 9: Simulations Table in Simulation Analysis Report

Simulation Data Point	Description
Failure	Failure scenario used in the analysis.
Service Class	Service class used in the analysis.
Traffic Level	Traffic level used in the analysis.
Network Breakpoint	Identifies whether there are network breaks resulting from the failures in this simulation. If multiple network breakpoints occur, the most serious one is listed. <ul style="list-style-type: none"> • Yes (Total)—A break exists that completely partitions the network into two or more disconnected sections. • Yes (AS)—A break exists that completely partitions an AS into two or more sections. However, routes exist between the sections of the AS through other ASes. • Yes (OSPF Area 0)—A break exists that completely partitions Area 0 of an AS running OSPF. Under OSPF, traffic cannot route between the partitions even if a path is available through non-zero areas in the AS. • No—No break in the network.
Num Unrouted Demands	Number of demands that cannot be routed under this failure for any of the reasons identified by the network breakpoint.
Unrouted Traffic	Total amount of demand traffic that cannot be routed under this failure for any of the reasons identified by the network breakpoint.
Max Util	Maximum utilization over all interfaces in this simulation. Utilization is the traffic through the interface as a percentage of the capacity of the interface.
Max QoS Bound Percent	Worst-case capacity available without violating QoS bounds, expressed as a percentage of the total capacity.
Num QoS Violations	Number of times the QoS bound is violated. QoS bounds are set through service class policies and interface queue parameters.
Latency Bound Violations	Number of demands with maximum latency in excess of the latency bound specified for the demand.
Num Unrouted LSPs	Number of unrouted non-Fast Reroute (FRR) LSPs in the analysis.
Num Unrouted FRR LSPs	Number of unrouted FRR LSPs in the analysis.
Num Unrouted L1 Circuits	Number of unrouted L1 circuits in the analysis.

Protecting Objects

To exclude an object from the list of those objects failed when performing a Simulation Analysis, you can mark it as *Protected* in its Properties dialog box. For example, if you want to run a Simulation Analysis only on core nodes, you could first protect all edge nodes.

You can protect nodes, sites, circuits ports, port circuits, external endpoint members, parallel circuits, L1 links, and L1 nodes.



Note If you select an interface, you are actually protecting its associated circuit.

- Step 1** Right-click one or more like objects from their respective tables.
- Step 2** In the Properties dialog box, check the **Protected** check box to toggle it on or off. A check mark means the object is protected. Then click **OK**.

Running Simulation Analysis

The Simulation Analysis tool is the basis of four failure analysis options: worst-case utilization on interfaces, worst-case VPN utilization and latency, worst-case demand latency, and failure impact.



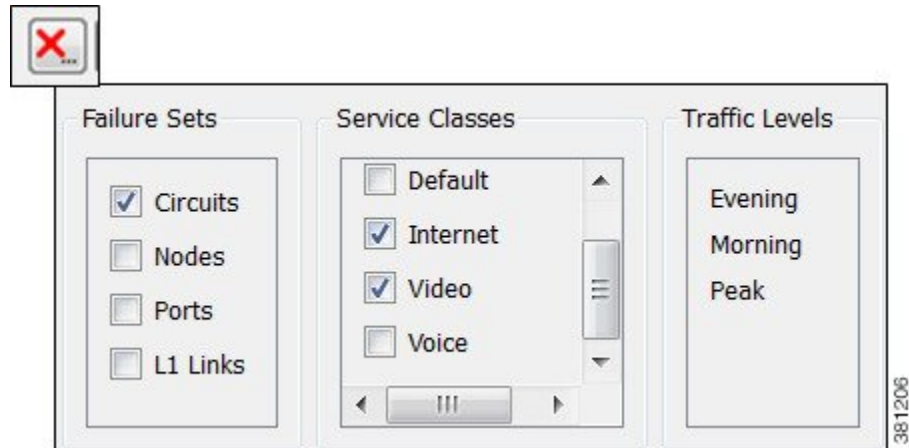
Note Recording worst-case latencies or VPN worst-case utilizations increases the time it takes to perform a worst-case analysis.

- Step 1** Choose **Tools > Simulation Analysis**.
- Step 2** Select one or more failure sets.
- Step 3** Select one or more traffic levels.
- Step 4** In the **Record failures causing utilizations within ___% of worst case** field, enter 0 to record only worst-case failures, or enter a number to find all failures causing utilizations within that percentage range of the worst-case failure.
- Step 5** Define the maximum number of failure scenarios to record per interface. The default is 10.
- Example: If you record failures causing utilizations within 10% of the worst case, and if the worst-case utilization for an interface is 90%, then WAE Design records failures on this interface resulting in utilization of 81% or higher (90 -(90/10)). In this same scenario, if you record 10 failure scenarios per interface, and if there are failures that could cause utilizations of 90%, 85%, 82%, and 76% for an interface, WAE Design does not record the failure causing 76% utilization.
- Step 6** Select whether or not to record demand worst-case latency calculations.
- Step 7** Select whether or not to record VPN worst-case utilizations and latencies. For more information, see [VPN Simulation, on page 279](#).
- Step 8** Enter the value for **Maximum number of threads**.
- Step 9** Click **OK**.

You can now use the Worst-Case Traffic view to analyze the worst-case traffic utilization, worst-case QoS violation, and worst-case latency information. Here you can also fail interfaces and nodes to their worst case, and fail demands to their worst-case latency. You can also use the [Failure Impact, on page 85](#) view to identify the circuits that are responsible for worst-case traffic congestion.

Viewing Simulation Analysis on a Subset of Failure Scenarios or Service Classes

After running a Simulation Analysis, you can select a subset of failure scenarios or service classes, and view the results for just these subsets. This practice saves time because you do not have to rerun the analysis.



After running a Simulation Analysis, follow these steps:

-
- Step 1** Click the red X in the top, right toolbar. A list of failure scenarios, service classes, and traffic levels appears. These are the options on which you last ran a Simulation Analysis.
- Step 2** Click one or more failure sets and service classes. Then click **OK**.
If you need to change the traffic level selection, you must rerun a Simulation Analysis.
-



CHAPTER 7

Traffic Forecasting

WAE Design traffic forecasting tools let you create plans containing estimates of future traffic based on projected growth of current measured or simulated traffic. From these new plans you can determine the impact of the new traffic on the network and plan for upgrades. The traffic growth estimates used in this process can be manually entered, for example, based on knowledge of new services being introduced. After creating these growth plans, you can analyze their capacity requirements using Simulation Analysis, and use the GUI to upgrade the network to meet these requirements.

One example use of these growth plan tools would be to predict the effects of a certain percentage of increased traffic for each quarter of the coming year. You could generate a separate plan for each of these traffic predictions to forecast the impact of each quarter's anticipated growth. Another example application would be to group the demands into demand groupings to predict the effects of different traffic growth at each site, such as 10% for Paris, 20% for London, and 30% for Amsterdam.

This section contains the following topics:

- [Demand Groupings, on page 91](#)
- [Growth Plans, on page 100](#)

Demand Groupings

Demand Groupings define a group of demands. They provide a convenient way of specifying aggregated traffic in a plan, which can be used as a basis for traffic reports and growth plans. For example, you can use demand groupings to represent the following.

- Total traffic sourced from one specific site
- Total VPN traffic, or any defined service class, sourced from one specific site to another
- Total traffic destined for a particular AS

To generate a report on demand groupings, including costs associated with them, choose **Tools > Reports > Demand Groupings**. For detailed information, see [Demand Groupings Reports, on page 98](#). For a description of the cost modeling columns in the report output, see [Cost Modeling, on page 389](#).

Demand Groupings and Their Filters

The term *demand grouping* refers to a set of filters (rules) that determine a group of demands. For example, you could create a demand grouping that filters all demands sourced from Paris, and name it ParisSrc. When

you apply the filter, if 52 of the existing demands contain Paris as a source, those 52 demands combined comprise a single demand grouping called ParisSrc.

Each row in the Demand Groupings table identifies a uniquely named demand grouping and the associated filters that define which demands are in it.

- Each column other than Name is a criterion that a demand must meet to be in that demand grouping.
- A demand must meet all criteria defined in these columns in order for it to be a member of the demand grouping.
- If the field is empty, it is not used as a criterion.
- The column headings with “Matches” in the name use regular expressions and the filters are applied accordingly.

Example: If Destination Node Matches is `*\.atl$`, a demand can be a member of this demand grouping only if it has destination nodes ending in `.atl` and if it meets all other criteria (listed in the other columns).

- The column heading that use “Equals” in the name must be exact matches.

Example: If Source Site Equals is `chi`, a demand can be a member of this demand grouping if its source site is named exactly `chi` and if it meets all other criteria.

- If the Tags Include column has an entry, a demand must contain this tag to be included in this demand grouping. (Demands can have multiple tags.)

Example: If Tags Include is `atlantic`, and if a demand has three tags named `europa`, `atlantic`, and `apac`, then the demand would be included in this demand grouping provided it meets all other criteria.

There are other object-based Tags Include columns for source and destination sites, nodes, and ASes. If one of these columns has an entry, the source or destination object must contain this tag to be included in this demand grouping.

Example: If Source Site Tags Include is `Europe` and if the Dest Site Tags Include is `Asia`, the demand would be included in this demand grouping if its source site is tagged with `Europe`, if its destination site is tagged with `Asia`, and if it meets all other criteria.

- Demand Grouping columns can contain information that does not exist in the current plan. This flexibility allows a demand grouping to be applied to any network.

To see a list of demands that comprise a demand grouping, in the Demands Grouping table, right-click the row containing the unique demand grouping name and select Filter to Associated Demands.

- In [Figure 28: Demand Groupings Contain Demands, on page 93](#), if you were to filter to associated demands on the first row (`chi` to `bos`), the resulting Demands table would show four demands because they are the only ones with both `chi` as the source and `bos` as the destination.
- If you were to filter to associated demands on the second row (`chi` to `All`), the resulting Demands table would show all demands with `chi` as the source, no matter the destination.

To see all demand groupings in which a demand exists, in the Demands table, right-click the row containing the demand of interest and select Filter to Demand Groupings.

- In [Figure 29: Demands Contained in Demand Groupings, on page 94](#), if you were to filter to associated demand groupings on the second row (`cr1.chi_cr1.mia`), the resulting Demand Groupings table would show the two demand groupings that contain this demand (`chi` to `All`, and `All` to `mia`).
- If you were to filter to associated demand groupings on the first row (`cr1.bos_cr1.mia`), the resulting Demand Groupings table would show the only demand grouping that contains it (`All` to `mia`).

Figure 28: Demand Groupings Contain Demands

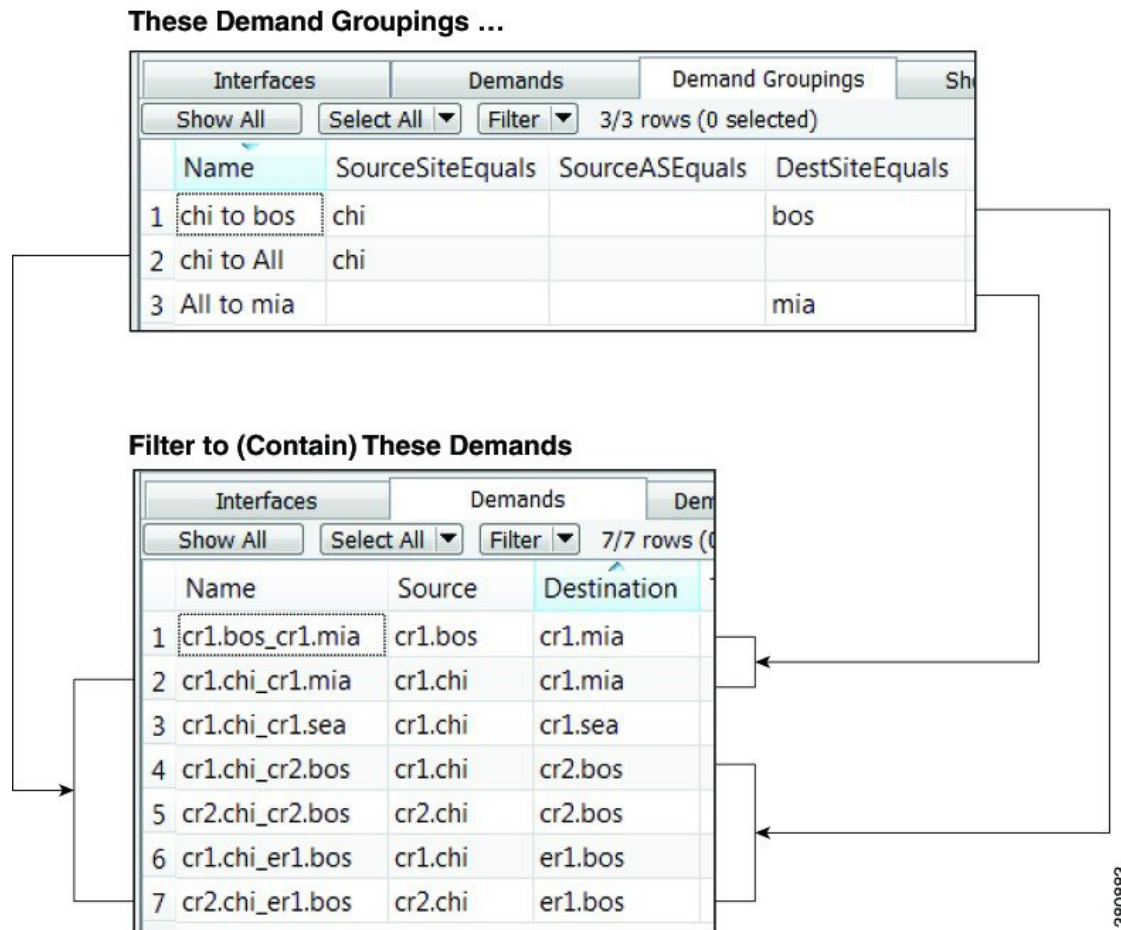
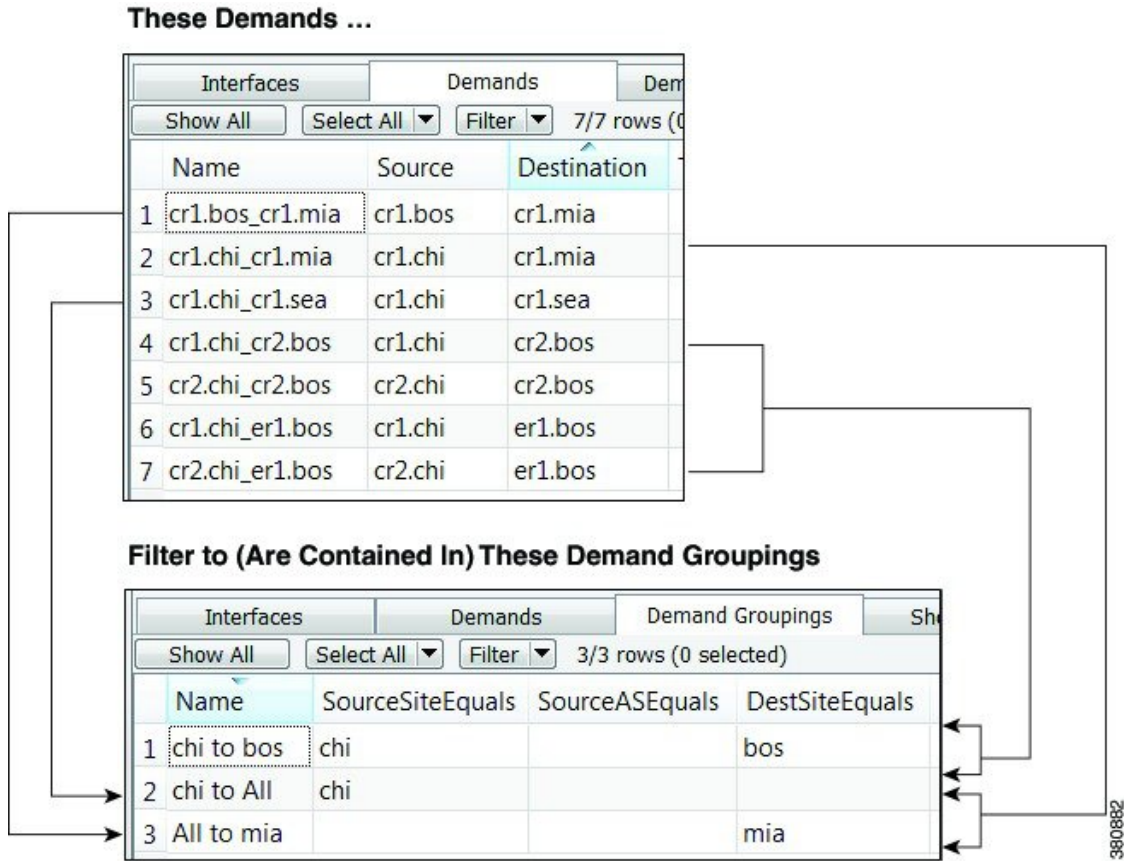


Figure 29: Demands Contained in Demand Groupings



Creating a Single Demand Grouping

This method lets you create a single demand grouping. All fields other than the Name field combine to create a set of rules, or filters, for the demand grouping. A demand must meet all of the specified criteria for it to be included in the demand grouping. If you specify one or more growth values, the demand grouping can be used in creating growth plans.

Note that you can also import demand groupings. For information, see the *Cisco WAE Design Integration and Development Guide*.

- Step 1** Choose **Insert > Demands > Demand Grouping**, or right-click in the empty plot area and choose **New > Demands > Demand Grouping**.
- Step 2** Enter a unique name for the demand grouping. If you do not add a name, WAE Design renames the grouping to make it unique.

Demands Filter

Demand Name

Service Class

Tags Include

<p>Source</p> <p><input type="checkbox"/> Site <input type="text"/></p> <p><input checked="" type="checkbox"/> Site Tags Include <input type="text" value="Midwest_Region"/></p> <p><input type="checkbox"/> AS <input type="text"/></p> <p><input type="checkbox"/> AS Tags Include <input type="text"/></p> <p><input checked="" type="checkbox"/> Node Matching <input type="text" value="\{gbd\}^*"/></p> <p><input type="checkbox"/> Node Tags Include <input type="text"/></p> <p><input type="checkbox"/> External Endpoint <input type="text"/></p>	<p>Destination</p> <p><input type="checkbox"/> Site <input type="text"/></p> <p><input checked="" type="checkbox"/> Site Tags Include <input type="text" value="Southern_Region"/></p> <p><input type="checkbox"/> AS <input type="text"/></p> <p><input type="checkbox"/> AS Tags Include <input type="text"/></p> <p><input type="checkbox"/> Node Matching <input type="text"/></p> <p><input type="checkbox"/> Node Tags Include <input type="text"/></p> <p><input type="checkbox"/> External Endpoint <input type="text"/></p>
--	---

380884

Step 3 Enter information as needed in the Demands, Source, and Destination fields to create the set of rules that define the demand grouping. Each of these is a demand filter.

- You can enter only one parameter per field.
- If a field is unchecked, the filter is not used.
- If you make a selection, you must enter a value in the empty field.
- All fields using “Tags Include” require a single tag to be specified. The list of tags for the corresponding object must include this tag.
- All fields using “Matching” allow you to use regular expressions.
- All fields without “Matching” require an exact match.

Step 4 If you want to use this demand grouping when creating growth plans, enter one or more values in the Growth area. See [Single-Period Demand Grouping Traffic, on page 101](#) for information on these values.

Step 5 Click **OK**.

Creating a Demand Grouping Mesh

A mesh of demand groupings is a set of demand groupings created between a specified set of sources and destinations. It is often more convenient to create an entire mesh at once than to create the demand groupings individually.

There are two methods for quickly creating a demand grouping mesh:

- Choose **Insert > Demands > Demand Grouping Mesh** or **New > Demands > Demand Grouping** as described the following steps.
- Importing existing demand groupings from another plan file or table file. For information on importing demand groupings, see the *Cisco WAE Design Integration and Development Guide* .

Because you are creating a set of filters, not all steps are required. By default, the Insert Demand Grouping Mesh uses selected sites (or all sites if none are selected) as the source and does not group by destination. To change the default and further restrict the demand grouping, you can choose from any combination of source, destination, service class, and demand name, as well as use object tags.

If you select multiple sites before opening the dialog box and use those sites for either the source, destination, or both, a unique demand grouping is created for each selection.

Examples: If you select both *chi* and *mia* sites for the source and nothing for destination, two demand groupings are generated: one named *chi to All* and one named *mia to All* . If you select both *bos* and *nyc* as both source and destination, four demand groupings are created: *bos to bos* , *bos to nyc* , *nyc to nyc* , and *nyc to bos* .

-
- Step 1** Select sites, nodes, or ASes if you want to create the demand grouping for a specific set of selected objects, whether as source, destination, or both.
- Step 2** Choose **Insert > Demands > Demand Grouping Mesh**, or right-click in an empty area and choose **New > Demands > Demand Grouping Mesh**.
- Step 3** For both source and destination, you have these options:
- Sites—All, those selected prior to opening the dialog box, internal sites only (those not containing external AS nodes), or a sites with a specific tag. For example, if you selected source sites tagged with Acme, only demand groupings whose source site was tagged with Acme would be included.
 - AS's—All, those selected prior to opening the dialog box, external only, or ASes with a specific tag.
 - Nodes—All, those selected prior to opening the dialog box, internal nodes only, or nodes with a specific tag.
 - External Endpoints—All, those selected prior to opening the dialog box, or external endpoints with a specific tag.
 - Site Tags, AS Tags, Node Tags—Objects that are tagged with any value. For example, if you select “Node Tags” for destination nodes, demand groupings would include those demands whose destination nodes have one or more tags of any name. If a destination node was tagged with APAC and another destination node was tagged with Europe, both would be included in the demand grouping mesh.
 - Do not group by—Ignore sources or destinations as filters for the demand grouping meshes.

The screenshot shows a configuration panel with two main sections: Sources and Destinations. Each section contains a list of radio buttons for selection and a set of dropdown menus. The Sources section includes options for Sites (Internal 12/12), AS's (External 0/1), Nodes (Internal 33/33), External Endpoints (Selected in table 0/0), and Do not group by source. The Destinations section includes options for Sites (Internal 12/12), AS's (External 0/1), Nodes (Internal 33/33), External Endpoints (Selected in table 0/0), and Do not group by destination. To the right of each section are checkboxes for Site Tags, AS Tags, and Node Tags. A vertical ID number 383290 is visible on the right side of the panel.

Step 4 In addition to or instead of selecting sources or destinations, you can create the grouping by restricting it to specific demands.

- Enter the exact demand name.
- Select a service class.
- Enter the exact source and destination node names or use regular expressions.

The screenshot shows a 'Restrict to' configuration panel. It contains four rows, each with a checkbox and a corresponding input field or dropdown menu:

- Demand Name: followed by an empty text input field.
- Service Class: followed by a dropdown menu showing 'Default'.
- Source Nodes Matching: followed by an empty text input field.
- Destination Nodes Matching: followed by an empty text input field.

 A vertical ID number 380887 is visible on the right side of the panel.

Step 5 The demand grouping names are created based on the name or variables used in the Name of Grouping field.

- Because each demand grouping created must have a unique name, we recommend that the string “\$1 to \$2” be used as the name, which is the default. The \$1 variable is source site, node, or AS, or All if none of these are selected but you have chose to include sources in the grouping. Likewise, \$2 is the destination site, node, AS, or All if neither is selected. Thus, a name can potentially be *All to All* .

If a demand grouping is selected using source and destination tags, then \$1 is the name of the source tag, and \$2 is the name of the destination tag.

- You can this field to create uniquely named demand groupings for the same site selections by entering extensions.

Example: You could select and use *chi* as the source, leave the destination empty, and add an extension of *voice*, thus creating a demand grouping named *chi to All voice* .

Grouping Names: \$1 to \$2-voice \$1 is Source, \$2 is Destination	380886
---	--------

Step 6 Click **OK**.

Demand Groupings Reports

You can generate Demand Groupings reports on plan files that contain demand groupings. Each report has two sections:

- **Demand Groupings**—For each demand grouping in the plan file, this section lists the name, traffic level to which it applies, and the number of demands. All other columns are based on the total (aggregate) of all demands in the demand grouping, traffic, capacity bound, capacity violation, QoS bound, QoS violation, cost by capacity, and cost by utilization. These costs are the sum of the demands in the demand grouping.
 - **Capacity Bound**—The maximum amount that the sum of the demands in the demand grouping can be without causing over 100% utilization on any interface in the network. The calculation is based on the assumption that all demand traffic increases proportionally.
 - **Capacity Violation**—Total traffic minus the capacity permitted for the demand grouping (Capacity Bound). A violation occurs if the Capacity Bound is exceeded. If the number appearing in the Capacity Violation column is positive, then the Capacity Bound has been surpassed. If the number is negative, the Capacity Bound has not been reached.
 - For information on QoS bound and QoS violation, see [Quality of Service Simulation, on page 239](#).
 - For information on how these costs are calculated, see [Cost Modeling, on page 389](#).
- **Demands**—For each demand in the plan file, regardless of whether it is in a demand grouping or not, this section lists the name, source and destination, service class, and number of demand groupings to which it belongs.

To generate a Demand Groupings report, choose **Tools > Reports > Demand Groupings**, and then click **OK**.

Example of Capacity Bound and Capacity Violation

Each interface imposes a limit on the traffic that can be carried by all demands in the demand grouping.

The goal of this example is to determine which interface is limiting the demand grouping's ability to deliver traffic. To do this, WAE Design looks at the interfaces over which each demand is routed. WAE Design then calculates the maximum traffic that can be used for the demand without exceeding 100% utilization for any routed interface.

In this example, the demands and interfaces are as follows.

- All interfaces have 1000 Mbps of traffic.
- The demand grouping is Acme_Region.
 - sjc_wdc_demand has 300 Mbps of traffic, and it traverses A_interface and B_interface.
 - sjc_hst_demand has 200 Mbps of traffic, and it traverses A_interface and C_interface.

- There are other demands outside this demand grouping.

A_interface has 600 Mbps of traffic. Of this, 500 Mbps of it is from the Acme Region demand grouping, and the remaining 100 Mbps is from demands outside this Acme_Region demand grouping.

B_interface has 500 Mbps of traffic, of which 300 Mbps is from the sjc_wdc_demand. The remaining 200 Mbps of traffic is from demands outside this demand grouping.

C_interface has 400 Mbps of traffic, of which 200 Mbps is from the sjc_hst_demand. The remaining 200 Mbps of traffic is from demands outside this demand grouping.

The calculation is as follows:

- A_interface

- A_interface can hold 900 Mbps of demand grouping traffic.

A_interface capacity – non-demand grouping traffic

$$1000 - 100 = 900$$

- The amount of sjc_wdc_demand traffic that A_interface can hold is 540 Mbps.

Total available capacity for the demand grouping * (sjc_wdc_demand/total demand grouping traffic)

$$900 * (300/500) = 540$$

- The amount of sjc_hst_demand traffic that A_interface can hold is 360 Mbps.

Total available capacity for the demand grouping * (sjc_hst_demand/total demand grouping traffic)

$$900 * (200/500) = 360$$

- B_interface can hold 800 Mbps of sjc_wdc_demand traffic.

B_interface capacity – non-demand grouping traffic

$$1000 - 200 = 800$$

- C_interface can hold 800 Mbps of sjc_hst_demand traffic.

C_interface capacity – non-demand grouping traffic

$$1000 - 200 = 800$$

- The limiting interface is A_interface because it imposes the strictest limits in the demand grouping's ability to deliver traffic. It supports only 360 Mbps of sjc_hst_demand traffic, whereas B_interface supports 800 Mbps of sjc_hst_demand traffic. Also, A_interface can support only 540 Mbps of sjc_wdc_demand traffic, whereas C_interface supports 800 Mbps of sjc_wdc_demand traffic.

- The Capacity Bound is 900 Mbps. If the sum of the demands in the demand grouping becomes 900 Mbps, the utilization of A_interface (the limiting interface) will be 100%. This 900 Mbps is the total available capacity for the demand grouping based on the limiting interface.

A_interface capacity – non-demand grouping traffic

$$1000 - 100 = 900$$

- The Capacity Violation is –400.

Total A_interface traffic permitted for the demand grouping – Capacity Bound

$$500 - 900 = -400$$

Figure 30: Example Demand Groupings Report

Reports							
<ul style="list-style-type: none"> ▲ Demand Groupings <li style="background-color: #e0e0e0;">Demand Groupings (9) Demands (95) 		Name	Traffic Level	Traffic	Number of Demands	Cost by Capacity	Cost by Util
		sjc to All	Default	3650.60	6	58925.16	187860.95
		sjc to All	Peak	14725.30	6	235547.56	369908.59
		sjc to All	Weekend	9695.55	6	149069.67	344901.80
		All to lax	Default	5924.67	5	137336.14	451218.90
		All to lax	Peak	9290.06	5	210953.77	354144.69
		All to lax	Weekend	9453.31	5	213061.40	450689.63
		All to chi	Default	7105.47	4	97903.69	279012.97
		All to chi	Peak	11232.84	4	144068.49	236757.12
		All to chi	Weekend	8696.71	4	115934.92	266750.57

380890

Growth Plans

The Create Growth Plan tool lets you generate new plan files containing predictions of future traffic, based on growth forecasts for the current plan. The predicted traffic growth can be generated in one of three ways.

- Per demand grouping using demand grouping traffic growth rates
- Per demand using demand traffic growth rates
- Per interface using interface traffic growth rates

Regardless of the method, the growth is compounded per period increment identified in the Create Growth Plan tool.

A series of plan files is created for each growth period, and each plan contains a report on the prediction process used to generate it. Each newly created plan shows the effects of the traffic growth visually in the plot using default traffic utilization colors, as well as in the Traffic column in the Demands table. By automatically running a simulation analysis on the new plans, you can incorporate worst-case performance into planning decisions.

Demand Grouping Traffic Growth

Growth plans for network traffic, such as those produced either by trending analyses or by financial or sales forecasts, are typically created for traffic aggregates. You can specify these aggregate growth forecasts or trending estimates as demand grouping traffic growth information. The Create Growth Plans tool allocates traffic to demands to match the aggregate traffic growth specified for each demand grouping. If two demand groupings contain a common demand (for example, a grouping of demands sourced from one site, and a grouping of demands destined for another site), this demand traffic is balanced between the requirement to match each of the demand grouping traffic aggregates.

WAE Design supports two methods of creating growth plans based on demand groupings: one is based on a single period, while the other is for multi-period traffic forecasting.

Single-Period Demand Grouping Traffic

The basic single-period method grows demands groupings based on the values identified in the Growth Traffic Total, Growth Traffic (%), and Growth Traffic Inc (increment) columns of the Demand Groupings table in the current plan. This approach is useful when all traffic in a given demand grouping is forecast to grow at the same rate or when you want to create growth plans for only one growth period.

If Growth Traffic Total is specified, the base traffic is set to this value. Otherwise, the base traffic is set to the sum of the current demand traffic in the demand grouping. The Create Growth Plans tool then applies the Growth Traffic (%), followed by the Growth Traffic Inc.

Example: If the demands in the demand grouping have a total of 15,000 Mbps of traffic, the Growth Traffic (%) is 5, and the Growth Traffic Inc is 800, the resulting total demand traffic in the first period would be 16,550 Mbps, which is $(15,000 * 1.05) + 800$. If you created growth plans for two period increments, the total demand traffic in second increment would be 18,177.50 Mbps, which is $(16,550 * 1.05) + 800$.

If the Growth Traffic Total is specified as 20,000 Mbps, then the current demand grouping traffic total is ignored, and the new demand grouping traffic total for the first period would be 21,800 Mbps, which is $(20,000 * 1.05) + 800$.

You can populate these growth columns in the Demand Groupings table as follows:

- Choose **Import > Demand Grouping Growth**.
- Choose **Edit > Traffic > Demand Groupings**.
- By editing the Demand Groupings Properties dialog box as described in the following steps.

-
- Step 1** If setting the same growth for multiple demand groupings, select them from the Demand Groupings table.
- Step 2** Double-click a demand grouping, or right-click one and choose **Properties**.
- Step 3** Enter one or more values in the Growth area.

- Traffic Total—Specifies an exact traffic total in Mbps.
- Traffic (%)—Specifies the percent by which to grow current traffic in Mbps.

Example: If the Traffic Total is 30,000 and Traffic (%) is 5, then the growth plan grows the traffic to 31,500 Mbps.

- Traffic Increment—Specifies the amount of traffic to increment by in Mbps.

Example: If Traffic Total is 11,000 and Traffic Increment is 800, the growth plan grows the traffic to 11,800 Mbps.

- Step 4** Click **OK**.
-

Multi-Period Demand Grouping Traffic

The Demand Grouping Traffic table is a way of specifying growth based on demand groupings for one or more specified periods. Example use scenarios could be as follows.

- Traffic sourced and destined for CHI will increase by 10% next quarter
- VPN traffic between MIA and BOS will increase by 1000 Mbps in the third quarter
- Transit traffic will reach 2 Gbps within a year

This table has the flexibility to specify the traffic total, traffic growth in specified increments (for example 10,000 Mbps), or traffic growth at a specified rate (percentage) per demand grouping.

The Demand Grouping Traffic table is created outside of WAE Design (for example, in a spreadsheet). You can then use it, including all of its periods, when creating growth plans. When used with the Create Growth Plans tool, the demand groupings in the current plan must match the names of those you are importing.

For information on creating this external table, see the *Cisco WAE Design Integration and Development Guide*.

Demand Traffic Growth

This method grows demands based on the value identified in the Growth (%) column of the Demands table in the current plan. This approach is useful when all traffic in the network, all traffic within a given traffic level, or all traffic within a service class is forecast to grow at the same rate. When generating the plans, you have the option of how many plans to create, and each plan will show growth at the same specified rate relative to the previous plan.

Example: If traffic on a demand is 1000 Mbps, if the growth for that demand is defined as 10%, and if you create two growth plans, the first will show 1100 Mbps and the second will show 1210 Mbps on that same demand, representing 10% growth per period.

You can populate the Growth (%) column in the Demands table using either of these two methods.

- Select one or more demands, right-click one, and choose **Properties**. In the Traffic section, enter the growth rate in the Growth % column.
- Import growth rates from a different plan file or table file. For instructions, see the *Cisco WAE Design Integration and Development Guide*.

Interface Traffic Growth

This method grows measured traffic based on the Growth (%) column in the Interfaces table in the current plan. It differs from demand-based growth methods because measured traffic, rather than demands (simulated traffic), is grown. This is useful for basic forecasting of the current network state, rather than worst-case failure states, which require simulation. When generating the plans, you have the option of how many plans to create, and each plan will show growth at the same specified rate relative to the previous plan.

You can also use this method in conjunction with Demand Deduction to create demands from the forecast measurements.

Example: If traffic on an interface is 525 Mbps, if the growth for that interface is defined as 5%, and you specify three growth plans, the first will show 551.25 Mbps, the second will show 578.81 Mbps, and the last will show 607.75 on the same interface, representing 5% growth per period.

You can populate the Growth (%) column in the Interfaces table using either of these two methods.

- Choose **Edit > Traffic > Measured Traffic**. In the Growth (%) column of the Interfaces table, enter the percent by which you want to grow the interface traffic for each applicable interface.

To set growth rates for multiple interfaces to the same value or to random values, select one or more interfaces, right-click, and choose **Set Value**. You then have the option to set fixed values for all selected interfaces or have WAE Design randomly set them within the range that you specify.

- Import growth rates from a different plan file or table file. For instructions, see the *Cisco WAE Design Integration and Development Guide*.

Creating Growth Plans

Step 1 Choose **Tools > Create Growth Plans**.

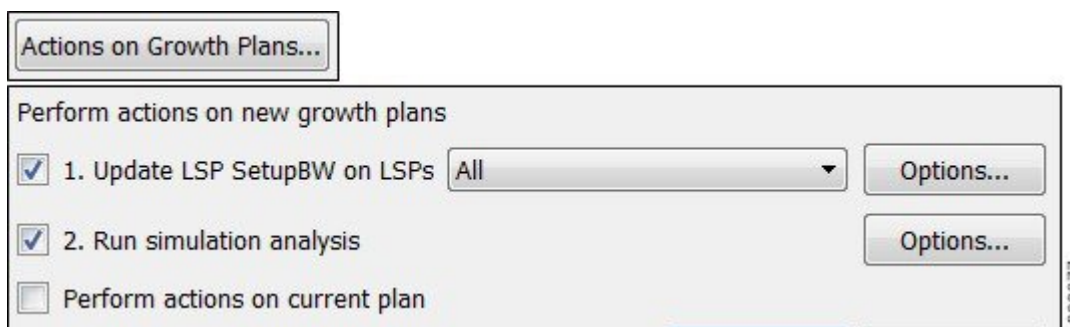
Step 2 Specify one of the following growth methods:

- Demand Groupings—For information on how to set demand grouping growth based on fixed, incremental, or percents, see [Demand Grouping Traffic Growth, on page 100](#).
- Demand Growth (%) in current plan—For information on how to set this percent, see [Demand Traffic Growth, on page 102](#).
- Interface Growth (%) in current plan—For information on how to set this percent, see [Interface Traffic Growth, on page 102](#).
- Demand Grouping Traffic Table—Browse to or enter the name of the file containing the Demand Grouping Traffic Table. For information on how to create these tables, see the *Cisco WAE Design Integration and Development Guide*.

Step 3 If creating growth plans based on demand groupings, demands, or interfaces, enter the number of plans to create and the increment by which you want to create them.

- Period increment—Number of percentage increments per period. For example, if a demand is set to grow at 10%, and the period increment is 2, the demand will grow by 21% in the first growth plan.
- Number of periods—Number of growth plan files to create.

Plans are named using the period increment. For example, if you enter 4 as the increment and 2 as period, WAE Design generates two growth plan files: one ending with _4 and one ending with _8.



Step 4 Click **Actions on Growth Plans** to perform these two tasks on the newly created growth plan files. To further specify how to execute them, click their associated **Options** button. You can also choose to execute these tools on the current plan.

- Upgrade LSP setup bandwidth on LSPs—For more information, see [RSVP-TE Simulation, on page 135](#).
- Run simulation analysis—For more information, see [Simulation Analysis, on page 79](#).

New plans named: euro_yearly_growth

Step 5 By default, WAE Design uses the current plan filename as the root name and adds the extensions generated by Step 3 for each growth plan. If you want the new plan files to use a different root name, enter that name in the New plans named field.

Open growth plans in GUI

Save growth plans

C:/CARIDEN_HOME/Voice_Growth

3800880

Step 6 To open these new plans upon exiting the dialog box, check **Open growth plans in GUI**.

Step 7 To automatically save these new plan files, check **Save growth plans**, and then enter or browse to the directory in which you want to save the files.

Step 8 Click **OK**.



CHAPTER 8

IGP Simulation

This chapter describes options for simulating an IGP. WAE Design can simulate IS-IS, OSPF, and EIGRP, as well as supports multi-topology routing for IS-IS and OSPF IGPs.

All simulations use demands. For detailed information on demands, see [Traffic Demand Modeling, on page 43](#)

This section contains the following topics:

- [OSPF and IS-IS, on page 105](#)
- [EIGRP, on page 107](#)
- [IGP Multipath Simulation, on page 107](#)
- [Multi-Topology Routing, on page 108](#)
- [IGP Metric Initializer , on page 109](#)
- [Viewing Shortest IGP, Latency, and TE Paths, on page 110](#)
- [Setting IGP Simulation, on page 112](#)

OSPF and IS-IS

OSPF and IS-IS simulations are identical with the following exceptions.

- OSPF routing uses OSPF areas, if specified. By default all interfaces are assigned area zero.
- IS-IS routing uses IS-IS levels, if specified. By default, all interfaces are set to Level 2. Interfaces can belong to Level 1, Level 2, or both. If both, then an alternate metric for Level 1 can also be specified. You can change these properties using the Advanced tab in an interface Properties dialog box.

OSPF Area Simulation

OSPF area membership can be specified per interface. The two interfaces on each circuit must belong to the same area. Area names can be any string. Area zero, the backbone area, must be denoted by “0”, “0.0.0.0”, or an empty string. WAE Design simulates the OSPF area routing configuration in which the areas import LSAs from the backbone. A demand from a source node to a destination node in a different area will only be routed if it can reach the destination by passing through the source area, directly to area zero, and from there directly to the destination area.

By default, all nodes in a single AS are assumed to belong to a single OSPF area. Nodes are assigned to areas, as follows.

- If OSPF areas are not defined for interfaces, all nodes are assumed to be in the same area.
- Each interface can be assigned to only one OSPF area. Note that each node can be assigned to one or more areas.
- If a node has an interface in an OSPF area, the node is assigned to that area.
- An Area Border Router (ABR) is a node belonging to both area 0 and other OSPF areas.

IS-IS Multi-Level Simulation

By default, IS-IS interfaces are assigned to Level 2, though you can assign them to Level 1 or to both Levels 1 and 2. (For instructions, see [Setting IS-IS Multi-Level Simulation, on page 106.](#))

- The IGP metric defines the Level 1 metric for interfaces in Level 1, in Level 2, or in both Level 1 and 2 if these have equal metrics. However, you can change the Level 1 metric.
- If an interface is in both Level 1 and Level 2 with uneven metrics, the IGP metric defines the Level 2 metric and the Level 1 metric defines the Level 1 metric.

The IS-IS level is listed in the ISIS Level column of the Interfaces table, and the Level 1 metric is listed in the Metric Level 1 ISIS column.

By default, all nodes in a single AS are assumed to belong to a single IS-IS level. Nodes are assigned to levels, as follows.

- Each node can be assigned to one or more levels.
- Two Level 1 nodes are placed in different areas if any route between them (under normal operation) passes through a Level 2 node.
- A node is assigned to a single Level 2 area if there is at least one interface that is Level 2 or both Level 1 and Level 2.
- A node is assigned to one of potentially multiple Level 1 areas if any interface is Level 1 or both Level 1 and Level 2.
- An ABR is a node belonging to both the Level 2 area and another IS-IS area.

Setting IS-IS Multi-Level Simulation

Follow these steps to set IGP metrics, IS-IS levels, and Level 1 metrics.

- Step 1** Choose one or more interfaces. Right-click one and choose **Properties**, or double-click an interface.



IGP Metric:

- Step 2** To set the IGP metric, enter its value in the IGP metric field or choose it from the drop-down list.



- Step 3** To set IS-IS multi-level simulation, click the **Advanced** tab.

- a) In the IGP area, select the level from the ISIS Level drop-down list.
- b) In the Level 1 Metric field, enter the metric, and click **OK**.

EIGRP

The IGP Metric value for each interface is not used when EIGRP routing is selected. Instead, EIGRP uses the following formula to derive total feasible distance (total cost) from a node to a destination subnet.

$$\text{path metric to destination} = (10,000/(\text{bandwidth}) + (\text{delay}) * 256$$

- The bandwidth is the minimum interface Capacity value along the path. This is in Mbps. For each interface, if there is no Capacity value, the Capacity Sim value is used instead.
- The delay is the sum of the interface delays in 10s of microseconds. This is calculated by taking the sum of the EIGRP Delay values and dividing this sum by 10.

You can set this delay in the EIGRP Delay field of an interface's Properties dialog box. Here, the value is in microseconds.

If EIGRP Delay is not set, WAE Design uses 10 as the delay in the preceding calculation, which means 10 "10s of microseconds."

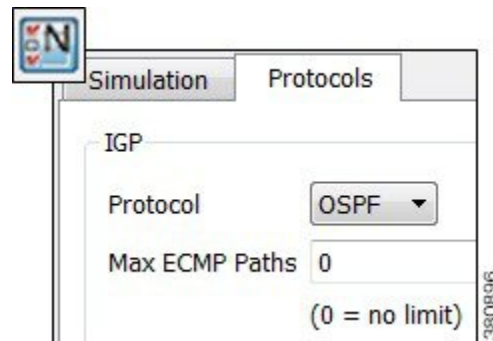
Example delay: If there are 8 interfaces, each with an EIGRP Delay value of 15, the delay used in the calculation is $(8 \times 15) / 10 = 120 / 10 = 12$.

Demands show the EIGRP path metric of the path over which they flow in the Path Metric column.

IGP Multipath Simulation

By default, demands are split equally between all paths from a route of equal distance to the destination, and there is no ECMP limit.

You can specify a maximum number of ECMP paths. In this case, demands transiting through a router are distributed among the available paths up to this maximum ECMP value. Paths are chosen by lowest next-hop IP address, which is the interface IP address for IGP or the destination IP address for LSPs. Paths with no IP address are chosen last.



To configure ECMP, follow these steps:

-
- Step 1** Choose **Edit > Network Options** or click the **Network Options** icon in the Visualization toolbar.
- Step 2** Click the **Protocols** tab.
- Step 3** In the Max ECMP Paths field, enter the maximum number of ECMPs. A 0 (zero) means there is no limit.
- Step 4** Click **OK**.
-

Multi-Topology Routing

WAE Design supports simulation of multi-topology routing. Interfaces can be assigned to one or more IGP. Demands and LSPs can be assigned to a specific IGP, and will only route through interfaces belonging to that IGP. This multi-topology simulation uses these rules.

- When an interface is common to more than one topology, all IGP properties of the interface (including the metric) must be the same in both topologies.
- The IGPs must all be defined to be of the same type, for example OSPF or IS-IS. This is not a restriction in practice unless either the OSPF topology or the IS-IS topology uses multiple areas.
- Demands and LSPs defined with a specific topology can only route over circuits belonging to that topology. A circuit belongs to a topology if either one of its interfaces does.
- Demands and LSPs with no defined topology have no restrictions on routing. They are routed using the default topology, to which all interfaces belong.

Editing Topologies

To create, rename, or delete topologies, choose **Edit > Topologies**.

To...	Do This...	Enter...	Click...
Create a new topology	Click New	A topology name	OK and OK
Rename an existing topology	Select the topology and click Rename	A different topology name	OK and OK
Remove a topology	Select the topology and click Delete	—	OK

Adding Topologies to Objects

Follow these steps to associate interfaces, demands, or LSPs with topologies. Once you have associated the topologies, you can view them by showing the Topology column.

-
- Step 1** Select one or more interfaces, demands, or LSPs from their respective tables.
- Step 2** Right-click one of the selected objects and choose **Properties**.
- Step 3** Select the desired topology or topologies, as follows.

Interfaces	Demands and LSPs
<ol style="list-style-type: none"> a. Click the Advanced tab. b. In the Topologies area, click the Edit button. <ol style="list-style-type: none"> 1. For each topology that you want on this interface, click the F (false) to change it to T (true). 2. Click OK. c. Click OK. 	<ol style="list-style-type: none"> a. Select the desired topology from the Topology drop-down list. You can associate one topology to each demand or LSP. b. Click OK.

By selecting from a similar Topology drop-down list, you can also add a topology to LSP meshes when creating them.

IGP Metric Initializer

The IGP Metric Initializer is a quick way of setting IGP metrics based on circuit latency values. WAE Design calculates the metric by multiplying each circuit's Delay Sim value by the multiplier you specify. You also set a minimum value for the resulting IGP metric. WAE Design then uses the higher of the two values.

Set Interface IGP metrics to $\max(\text{"Delay Sim"} * \text{Multiplier}, \text{Minimum})$

Multiplier

Minimum

Example: If the circuit Delay Sim value is 15, the multiplier value is 2, and the minimum value is 1, the resulting IGP metric is 30 (Delay Sim * Multiplier). However, if the minimum value were 50, the resulting IGP metric would be 50.

Because WAE Design updates all metrics on all selected interfaces, you can modify interfaces individually or in groups. For example, you might select all core routers in the Interface table and then simultaneously calculate their IGP metrics.

Step 1

Make your selections.

- Select one or more interfaces.
- Select one or more circuits. This updates IGP metrics for both associated interfaces.
- If you make no selections, IGP metrics are calculated for all interfaces.

Step 2

Choose **Initializers > IGP Metric**, or right-click an interface or circuit and choose **Initialize IGP Metric**.

Step 3

Enter a multiplier and minimum value. Then click **OK**.

Viewing Shortest IGP, Latency, and TE Paths

The Shortest Paths table is dynamically generated to show routes that traffic would take between selected nodes or between all nodes in selected sites. Columns include information about the total IGP metrics used in the routes, as well as numerous latency columns. Because shortest paths are not objects in the plan file, there is no associated Properties dialog box. You can, however, right-click paths to filter to interfaces or to source and destination nodes to change their properties, as needed, to change the path.

- Shortest IGP paths are the shortest paths as calculated by IGP metric.
- Shortest latency paths are the shortest paths in an IGP network as calculated by the circuit latencies.
- Shortest TE paths are the shortest paths as calculated by the interface TE metric.

To view these shortest paths in the network plot, select them from the table ([Figure 31: Shortest IGP, Latency, and TE Paths, on page 111](#)) using the following steps.



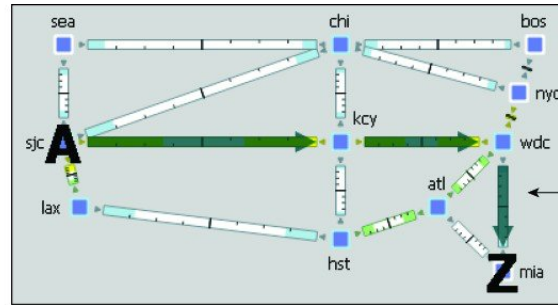
Note Filtering to the shortest IGP Path between two nodes when EIGRP is used does not show the shortest EIGRP path. Rather, it shows the shortest path using the traditional IGP metrics for each interface.

-
- Step 1** Select one or more nodes, or select one or more sites. If you select both, WAE Design uses the object type from which you select the context menu. It does not use both simultaneously.
- Step 2** Right-click one of the objects. For nodes, choose **Filter to > Shortest Paths > <selection>**. For sites, choose **Filter to intersite Shortest Paths > <selection>**. The selection is any one of the following:
- Between selected nodes/sites (applicable only for multiple selections)
 - With selected nodes/sites as source
 - With selected nodes/sites as destination
 - Through all selected nodes/sites
 - Through any selected nodes/sites
- Step 3** The Shortest Paths table is populated according to your selections: shortest IGP, shortest TE, or shortest latency. To view a path in the plot, select it from this table. The Type column identifies the path type.
-

What to do next

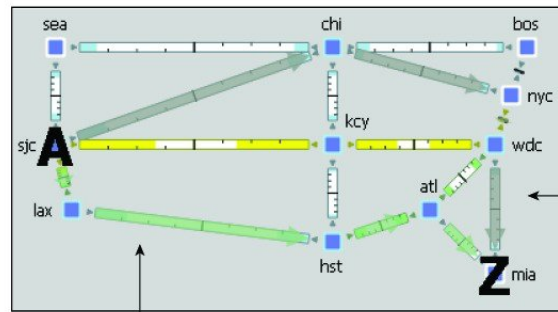
Figure 31: Shortest IGP, Latency, and TE Paths

Shortest Paths			
	Source	Destination	Type
1	er1.sjc	er1.mia	IGP
2	er1.sjc	er1.mia	TE
3	er1.sjc	cr1.mia	Latency



Shortest IGP Path

Shortest Paths			
	Source	Destination	Type
1	er1.sjc	er1.mia	IGP
2	er1.sjc	er1.mia	TE
3	er1.sjc	cr1.mia	Latency



Shortest TE Path

Shortest Latency Route

380808

Plotting Shortest Paths

From the Shortest Paths table, right-click one or more paths and choose **Plot Shortest Paths**. From here, you can see each node in the shortest path, as well as the traffic utilization on interfaces between each node in the path. Clicking **Select Shortest Paths** highlights the path in the network plot.

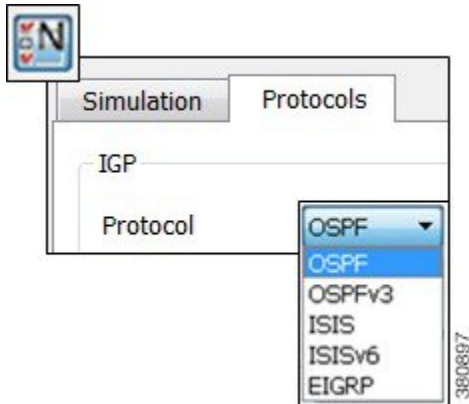
us_wan_L1 Shortest Paths Plot

Select Shortest Paths

Source	Destination	Type	Topology
1 er1.atl	er1.chi	IGP	
2 er1.atl	er1.hst	IGP	
3 er1.atl	er1.lax	IGP	

Setting IGP Simulation

Step 1 Choose **Edit > Network Options** or click the **Network Options** icon in the Visualization toolbar.



Step 2 Click the **Protocols** tab.

Step 3 Choose the appropriate protocol from the IGP Protocol list.

Step 4 Click **OK**.



CHAPTER 9

MPLS Simulation

This chapter describes how WAE Design simulates MPLS routing. All LSPs other than SR (segment routed) LSPs are routed like RSVP LSPs. For MPLS simulation information specific to these types of LSPs, see [RSVP-TE Simulation, on page 135](#) and [Segment Routing Simulation, on page 179](#).

- LSPs are established under normal operation: that is, with failures not taken into account.
- Any LSPs that are affected by the failures are rerouted. Depending on the LSP path settings, reroutes might involve moving to a secondary path, dynamically rerouting the LSPs, or rerouting based on a segment list.
- Demands are routed using the established LSPs using the specified IGP protocols given the specified failure scenarios.
- LSP utilizations are calculated from the demand traffic using the specified traffic level.

This section contains the following topics:

- [LSP Types, on page 113](#)
- [Creating and Visualizing LSPs and LSP Paths, on page 114](#)
- [Path Latency Calculations, on page 117](#)
- [Explicit LSP Paths Initializer, on page 117](#)
- [Routing Demands, on page 118](#)
- [Routing Inter-Area LSPs, on page 129](#)
- [Simulation Modes, on page 132](#)
- [LSP Establishment Order, on page 132](#)
- [Reports and Diagnostics, on page 133](#)

LSP Types

- SR LSPs—Segment routing LSPs, which do not use RSVP for routing. You can create SR LSPs using the WAE Design GUI. They are identified with a Type property of SR. For more information, see [Segment Routing Simulation, on page 179](#).
- RSVP LSPs—LSPs that are established through RSVP. These are commonly known as MPLS TE tunnels. WAE discovers RSVP LSPs, and you can also create them using the WAE Design GUI. They are identified with a Type property of RSVP. For more information, see [RSVP-TE Simulation, on page 135](#).

WAE Design does not model LDP tunnels as LSPs.

Creating and Visualizing LSPs and LSP Paths

To simulate MPLS LSPs in the network, you must first set up the LSPs to be routed. In the WAE Design GUI, enter LSPs into the plan file in the following ways:

- By choosing **Insert > LSPs > LSP**, or by right-clicking in the plot and choosing **New > LSPs > LSP**. This opens a dialog box for entering properties for a single LSP. One such property is the **Type**, which determines whether this is an RSVP LSP or an SR LSP.
- By choosing **Insert > LSPs > LSP Mesh**, or by right-clicking in the plot and choosing **New > LSPs > LSP Mesh**. This opens a dialog box for adding a mesh of LSPs between the selected nodes.



When selected, LSPs appear in the plot as a brown arrow. To view the details of LSP routes, as shown in [Figure 32: LSP Plot View of Three LSPs with the Same Source, on page 115](#), follow these steps.

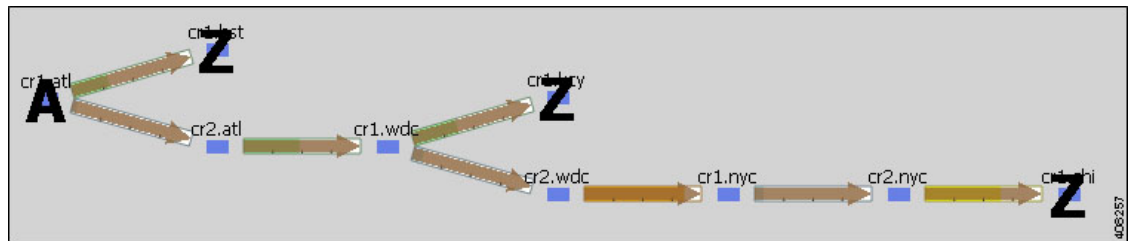
-
- Step 1** In the LSPs table, select one or more LSPs.
- When you select LSPs in the LSPs table, their associated L1 circuit paths are highlighted in the network plot.
- Step 2** Right-click and choose **Plot LSPs**.
- Step 3** To move the plot horizontally or vertically, use the left/right arrows and up/down arrows.
- Step 4** To select all LSPs within the LSP plot view, click **Select LSPs**. To deselect all of these LSPs, click in an empty plot area.
- Step 5** To visualize multilayer L1 paths when plotting LSPs, use the following drop-down options:
- **L3**—Displays the path in terms of interfaces.
 - **L1**—Displays the path in terms of L1 links.
- To better align the L1 links, WAE Design checks for L3-L1 links. If they do not exist, WAE Design checks for collocated L3/L1 nodes within sites.
- **L3+L1**—Displays the L3 plot above and the L1 plot below.
- WAE Design does not perform alignment between L3 and L1 nodes. When you click an L3 interface, the associated L1 links are highlighted automatically.
-

What to do next

You can plot one or more LSP paths in the same manner by selecting them from the LSP Paths table.

To filter to related information, such as related interfaces, source and destination nodes, or demands, right-click one or more LSPs and choose the option.

Figure 32: LSP Plot View of Three LSPs with the Same Source



LSP Paths

LSPs can be assigned one or more LSP paths. Like LSPs, LSP paths have properties that vary depending on whether the path is for an RSVP LSP or SR LSP. If these properties are omitted, then they are inherited from the LSP. If these properties are set in the LSP path, they override the LSP settings. For information on these properties, see [RSVP-TE Simulation, on page 135](#) and [Segment Routing Simulation, on page 179](#).

After an LSP path is created, you can use an initializer to create explicitly routed paths. For information, see [Creating LSP Paths, on page 115](#) and [Explicit LSP Paths Initializer, on page 117](#).

Path Options and Active Path

Each LSP path has a Path Option property. The LSP is routed using the first LSP path that can successfully be established. LSP paths are established in increasing order of their path option, where path option 1 is established first.

LSP Name:	cr1.lon_cr2.fra
Path Option:	2

Alternatively, you can choose which LSP path to use from the LSP Active Path drop-down list in the LSP Property dialog box.

LSP Property

Active Path:	Not set
	Not set
	1
	2
	3

Creating LSP Paths

Step 1 In the LSPs table, choose the LSP to which you are adding LSP paths.

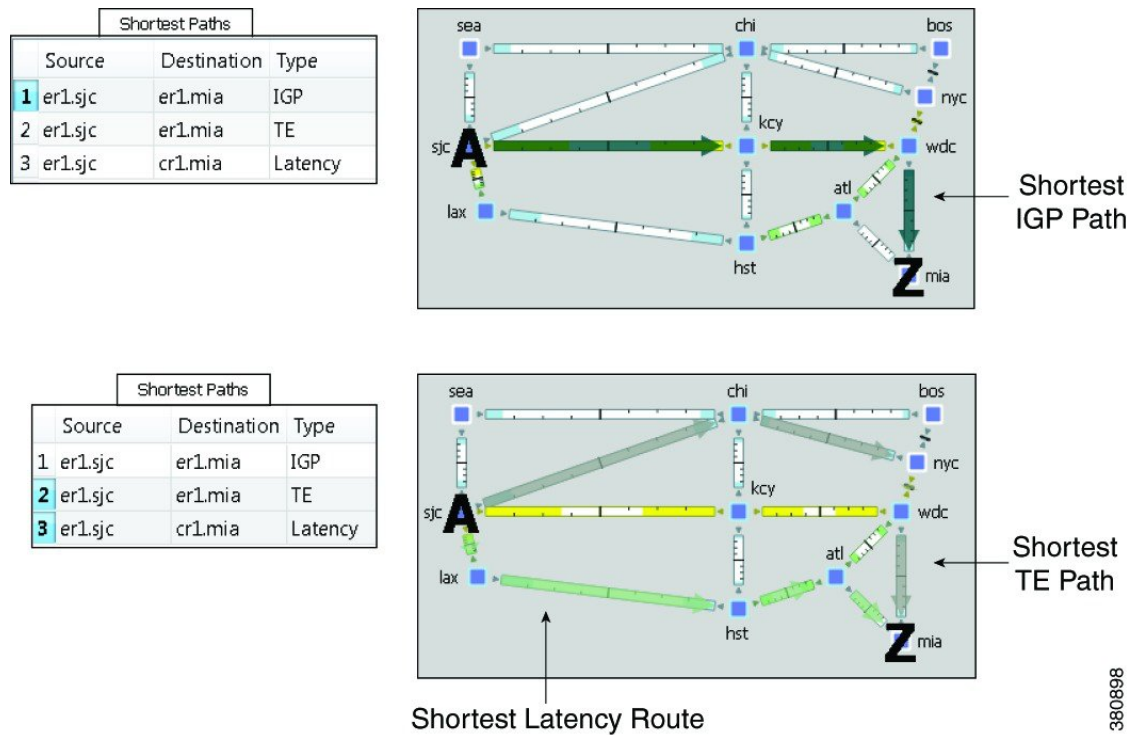
- Step 2** Right-click in the plot choose **New > LSPs > LSP Paths**, or choose **Insert > LSPs > LSP Paths**.
- Step 3** In the Path Option field, enter the order in which the LSP path is activated. Lower numbers have preference over higher numbers.
- Step 4** If this is an RSVP LSP path, optionally set these properties. For information on these properties, see [RSVP-TE Simulation, on page 135](#).
- If this is a standby LSP path, check **Standby**, which ensures the paths are always active.
 - To set the bandwidth, specify the **Setup Bandwidth** for the LSP path, or specify that it should inherit the bandwidth from the LSP.
 - To create associated named paths, check this option and complete the name using \$1 for the LSP name and \$2 for the path option.
 - To associate the LSP path with affinities, click **Edit**. For example, you might want to associate a standby path with an affinity that forces it on a different path than the primary path. LSP paths can inherit the affinity from the LSP or can be assigned a specific affinity. Click **OK**.
- Step 5** Click **OK**.

Visualizing LSP Shortest Paths

To view the LSP shortest paths, use one of these methods. The source is marked with an “A” and the destination is marked with a “Z”.

- Using the **View > LSPs** submenus, you can choose to see the shortest actual paths, shortest latency paths, or shortest TE paths each time that you choose an LSP.
- To view all shortest IGP paths, shortest TE paths, or shortest latency paths for selected LSPs, use the context menu. Choose one or more LSPs, right-click, and choose **Filter To > Shortest Paths** ([Figure 33: LSP Shortest IGP, Shortest TE, and Shortest Latency Paths, on page 117](#)).

Figure 33: LSP Shortest IGP, Shortest TE, and Shortest Latency Paths



380898

Path Latency Calculations

WAE Design calculates shortest latency paths for LSPs and demands as follows:

- By default, WAE Design uses the shortest path from the source to the destination of the LSP or demand, where the weight on each interface is the delay value for that interface.
- If the delay for any interface is zero (as it is by default), a small (0.00001 ms) delay is used instead. This means that if all the delays for the interfaces in the plan are zero, the path with the smallest number of hops is selected.
- If two paths have the same latency, paths with the smaller number of hops are preferred.

Explicit LSP Paths Initializer

The Explicit LSP Paths Initializer automates the process of explicitly routing existing LSP paths to follow their currently simulated routes.



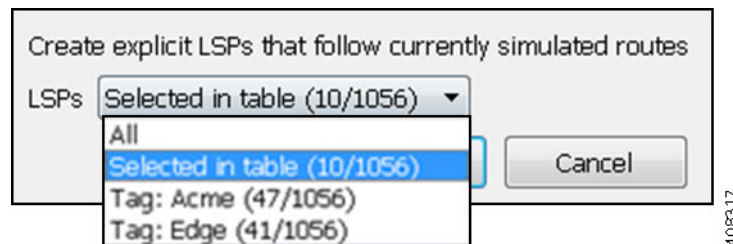
Note Explicit LSP Paths Initializer supports RSVP only (in terms of ABR hop creation).

- For RSVP LSPs, the initializer converts a set of dynamically routed LSPs to explicitly routed LSPs along their current paths.

The initializer creates named paths and named path hops for both primary and secondary paths. The explicit paths are named *<LSP Name> _<Path Option>*. Orphaned named paths are deleted. (Named paths are *orphans* if they are no longer used by any of the LSP paths.) For more information on RSVP LSPs, see [RSVP-TE Simulation, on page 135](#).

- For SR LSPs, the only use case is if you want the initializer to create a final segment list hop on the destination node. For more information on SR LSPs, see [Segment Routing Simulation, on page 179](#).

WAE Design writes a report identifying the number of selected LSP paths. To access this information later, choose **Window > Reports**



-
- Step 1** Choose **Initializers > Explicit LSP Paths**, or right-click an LSP and choose **Explicit Path Initializer**.
- Step 2** Choose whether to create explicit paths for all LSPs, selected LSPs, or tagged LSPs, and then click **OK**.
-

Routing Demands

Intra-Area LSPs

Intra-area LSPs are modeled as IGP shortcuts. That is, the source of the demand using the LSP can be a node other than the ingress node into the IGP, and the destination node of the LSP can be a node other than the egress node from the IGP. A demand through intra-area LSPs does not require the LSP to traverse the full demand path.

Each intra-area LSP has a metric that helps determine which traffic is routed through the LSP. By default, autoroute LSPs have a metric equal to the shortest IGP distance from the source to the destination. However, you can configure static metrics for these LSPs that override this default. Static metrics can also be defined relative the shortest IGP distance, but these relative metrics are not currently supported in WAE Design.

Forwarding adjacency LSPs always have a metric. If not specified, the default is 10. Forwarding adjacency LSP metrics are injected into the IGP so that nodes other than the source node of the LSP are aware of the path length through the forwarding adjacency LSP, and use it in their shortest path calculations.

To edit autoroute and forwarding adjacencies, choose one or more LSPs from the LSP table; double-click to bring up the Properties dialog box and edit the values in the Routing area.

Inter-Area LSPs

Because it is not possible to define metrics distances across ISP areas, there can be no well-defined metric for inter-area LSPs. Therefore, in WAE Design, a demand only routes through an inter-area LSP if the demand's endpoints are nodes matching the source and destination of the LSP, interfaces on these nodes, or external ASes whose ingress and egress points through the IGP are these nodes. This requirement is true regardless of the network option settings.

To be routed, these demands must also match the privacy, tags, and class-based forwarding requirements. Autoroute, forwarding adjacency properties, and LSP metrics are ignored. For information on privacy, see [Private LSPs, on page 126](#), for information on tags, see [Plan Objects, on page 23](#), and for information on class-based forwarding, see [Class-Based Forwarding, on page 120](#).

Loadsharing

Two or more LSPs with the same source and destination (and metrics, if these are defined) loadshare traffic between them. How the load is shared between the LSPs is determined by the LSP Loadshare property. By default, LSPs have a Loadshare property of 1, and thus route traffic between them in equal proportions. Changing the Loadshare value changes the distribution of LSP traffic and interface traffic in proportion to these values.

Example: If two LSPs are parallel, and one has a Loadshare property of 2 and one has a Loadshare property of 1, there will be a 2-to-1 ratio of traffic shared between them. The top half of [Figure 34: Example of Two Parallel LSPs with Equal Loadsharing, on page 120](#) shows an example of two parallel LSPs that are routed using strict explicit paths. Each has a Loadshare value of 1, which means the traffic is routed using a 1:1 loadshare ratio so that each LSP carries 50% of the traffic. In contrast, the lower half shows the same parallel LSPs with a 2:1 ratio. That is, one LSP has a Loadshare value of 2, and one has a Loadshare property value of 1. The LSP with a Loadshare value of 2 carries 67% of the traffic, while the other carries 33%.



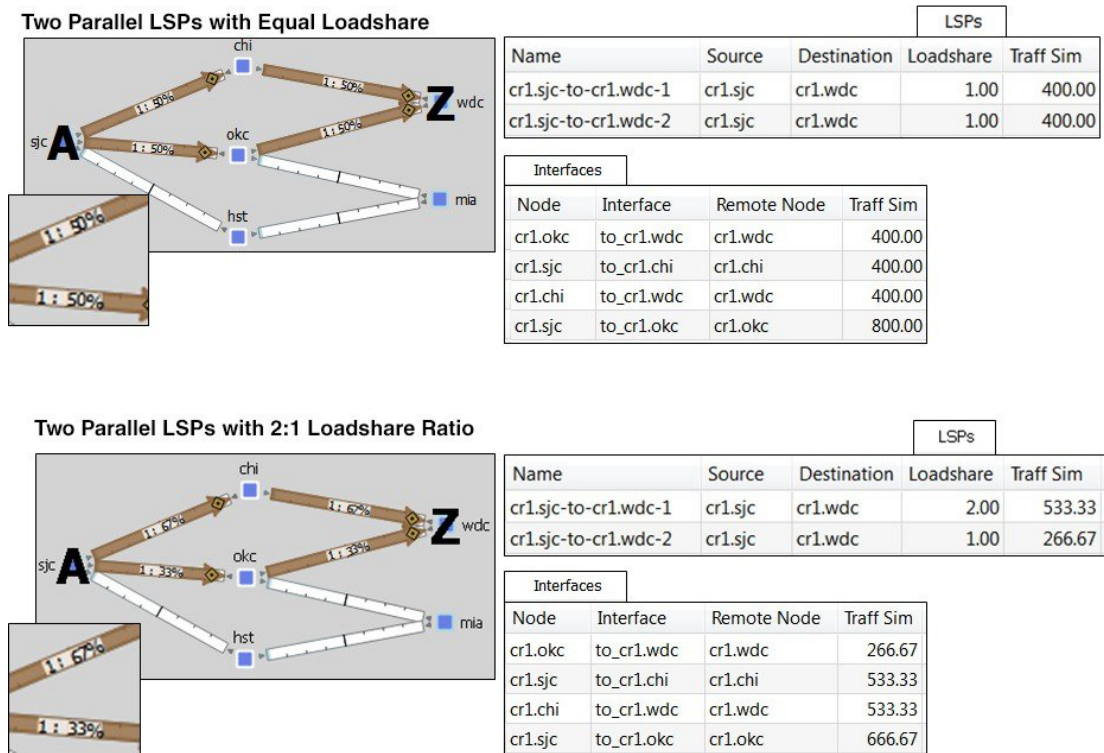
Each LSP has a path showing the percentage of traffic that crosses the LSP as a result of the Loadshare value for that LSP. The format is *<path option> :<loadshare percentage> %*. Although loadshare applies only to path option 1, the LSP itself might be routed along a different path.



Note that this percentage may or may not be the same as the Loadshare value in the LSPs table. This is because the Loadshare value is relative to the other Loadshare values for parallel LSPs. For this reason, if manually editing the Loadshare property, we recommend that you consider the entire set of parallel LSPs. Note also that because an LSP can be in multiple sets of parallel LSPs, the percentage on its path can differ.

To optimize these Loadshare values, use the LSP Loadshare Optimization tool. For information, see [LSP Loadshare Optimization, on page 316](#).

Figure 34: Example of Two Parallel LSPs with Equal Loadsharing



381100

Class-Based Forwarding

Class-based forwarding (CBF) lets you manage differentiated service requirements by restricting or allowing demands into LSPs according to a set of user-defined mappings between demand service classes and LSPs. For example, a demand with an Assured Forwarding service class could be mapped to an LSP saved for carrying low-latency voice traffic, while a different LSP using that same demand could be used for only best-effort services, such as file transfers.

Class-Based Forwarding Simulations

Demands can be assigned a Service Class property. LSPs can be assigned a Class property. CBF is enabled by creating a table that maps the demand service class to the LSP classes. The resulting table defines a set of LSPs to which a demand with a specific service class has access.

Figure 35: Example Service Class LSP Class Mapping Table

Service Class LSP Class Mapping			
<input type="button" value="Show All"/>		<input type="button" value="Select All"/> ▼	<input type="button" value="Filter"/> ▼ 4/4 rows
	Service Class	Priority	LSP Class
1	AF	1	LSPClass_AF
2	AF	2	LSPClass_BE
3	AF	3	Drop
4	BE	1	LSPClass_BE

- Each demand service class in the table is mapped to one or more LSP classes, or with a Drop value.
- If a demand's service class is in the mapping table, the demand can only use LSPs with LSP classes that match that service class.

If a demand service class is not in the mapping table, it can use any LSP.

Example: There are three demand service classes: AF, BE, and EX. The mapping table contains only AF and BE (Figure 35: Example Service Class LSP Class Mapping Table, on page 121). This means that the demands with EX service class can use any LSP in the plan file.

- If an LSP class does not appear in this mapping table, then any demand can use any LSP of that class.
- If multiple LSPs are sourced from one node, demands routed through that node can use any LSP to which the demand's service class is mapped.
 - If all available LSPs have LSP classes with equal priority, the traffic is load balanced between these LSPs.
 - If the LSPs have prioritized LSP classes (1, 2, 3, etc.), those with the highest priority (lowest number) are routed first.

Example: A demand with an AF service class must first use LSPs with an LSPClass_AF because within the AF service class, this is the only Priority 1 LSP class. If no LSPs with LSPClass_AF are available, the demand tries to route across LSPs with an LSPClass_BE class (Figure 35: Example Service Class LSP Class Mapping Table, on page 121).

- If there are no available LSPs with matching LSP classes, the demand follows these rules.
 - If the demand's service class is not mapped to a Drop value, the demand forgoes using LSPs and takes the shortest IGP path.
 - If the demand's service class is mapped to a Drop value, the demand is unrouted.

Example: If there are no available LSPs for the AF demand to route across, it will not route due to the Drop value associated with it (Figure 35: Example Service Class LSP Class Mapping Table, on page 121). If the BE demand cannot route across an LSP that has a class of LSPClass_BE, it will route across the shortest IGP path available.

Example Simulations

In these examples, we want to route the assured-forwarding (AF) Europe-to-Japan traffic directly over the more expensive, but lower latency, circuits through Russia. We want to route the best-effort (BE) traffic along the cheaper, but higher latency, circuits through the US.

Demands			
Name	Source	Destination	Service Class
Lon-Tok-D1	cr1.lon	cr1.tok	AF
Lon-Tok-D2	cr1.lon	cr1.tok	BE

LSPs			
Name	Source	Destination	Class
LSP_A	cr1.lon	cr1.tok	LSPClass_AF
LSP_B	cr1.lon	cr1.tok	LSPClass_BE

381094

In these example simulations, there are only two demands and two LSPs. Both demands and both LSPs go from London to Tokyo.

- Lon-Tok-D1 demand uses an AF service class.
- Lon-Tok-D2 demand uses an BE service class.
- The LSPs are named LSP_A and LSP_B. Each LSP is explicitly routed to reach Tokyo via a different path.
 - LSP_A is assigned to LSPClass_AF.
 - LSP_B is assigned to LSPClass_BE.

[Figure 36: Example Reroute Around Failure Using IGP Path, on page 123](#) shows a simple one-to-one mapping between a service class and an LSP class.

- The AF service class is mapped to LSPClass_AF.
- The BE service class is mapped to LSPClass_BE.
- If the circuit between London and Tokyo fails, the Lon-Tok-D1 demand reroutes using the shortest IGP path. The reason is because there are no alternative LSPs available.

Figure 36: Example Reroute Around Failure Using IGP Path

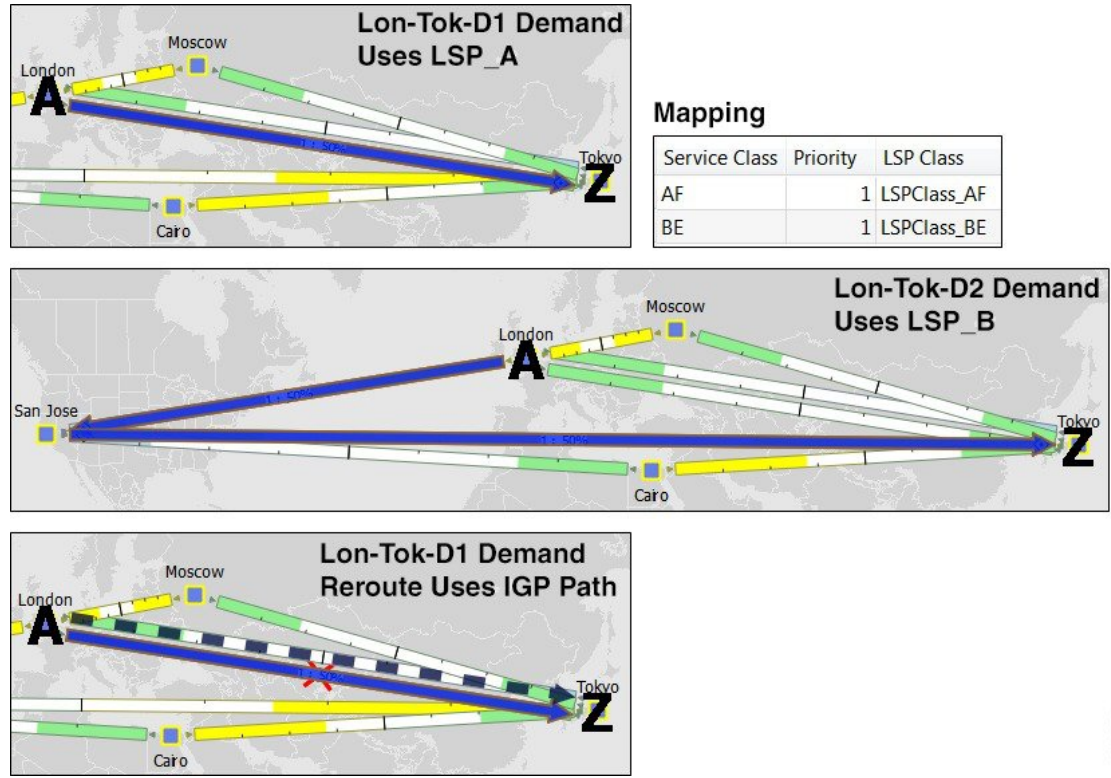


Figure 37: Example Reroute Around Failure Using Alternative LSP Class, on page 123 shows that the AF service class is mapped to both LSPClass_AF and LSPClass_BE. The Lon-Tok-D1 demand now reroutes around a failure between London and Tokyo by using LSP_B. This is because the AF service class now has a Priority 2 mapping to the LSPClass_BE class, and the fact that LSP_B is assigned to this LSPClass_BE class.

Figure 37: Example Reroute Around Failure Using Alternative LSP Class

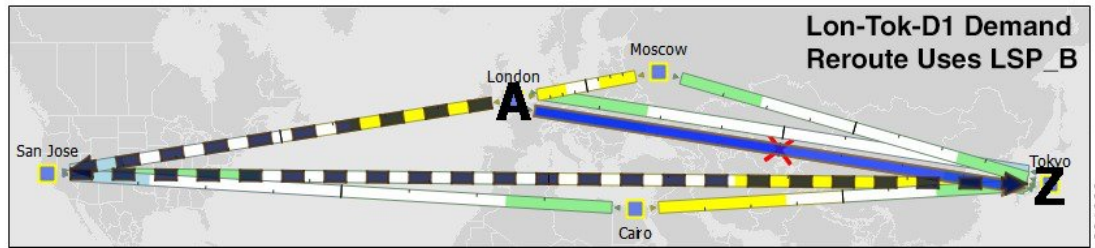


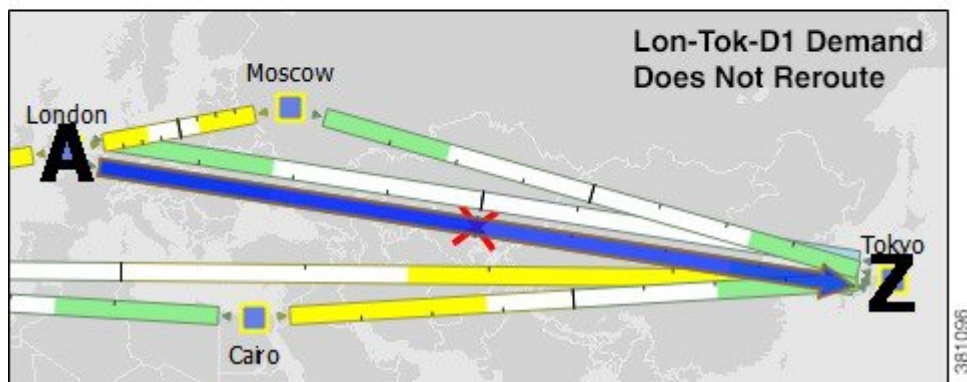
Figure 38: Example of Not Rerouting Based on Drop LSP Class, on page 124 shows that the AF service class is mapped to both LSPClass_AF and a Drop value. The Lon-Tok-D1 demand now does not reroute around a

failure between London and Tokyo. This is because demands using AF service class are mapped so that if an LSP with an LSPClass_AF class is not available, the traffic is dropped.

Figure 38: Example of Not Rerouting Based on Drop LSP Class

Mapping

Service Class	Priority	LSP Class
AF	1	LSPClass_AF
AF	2	Drop
BE	1	LSPClass_BE



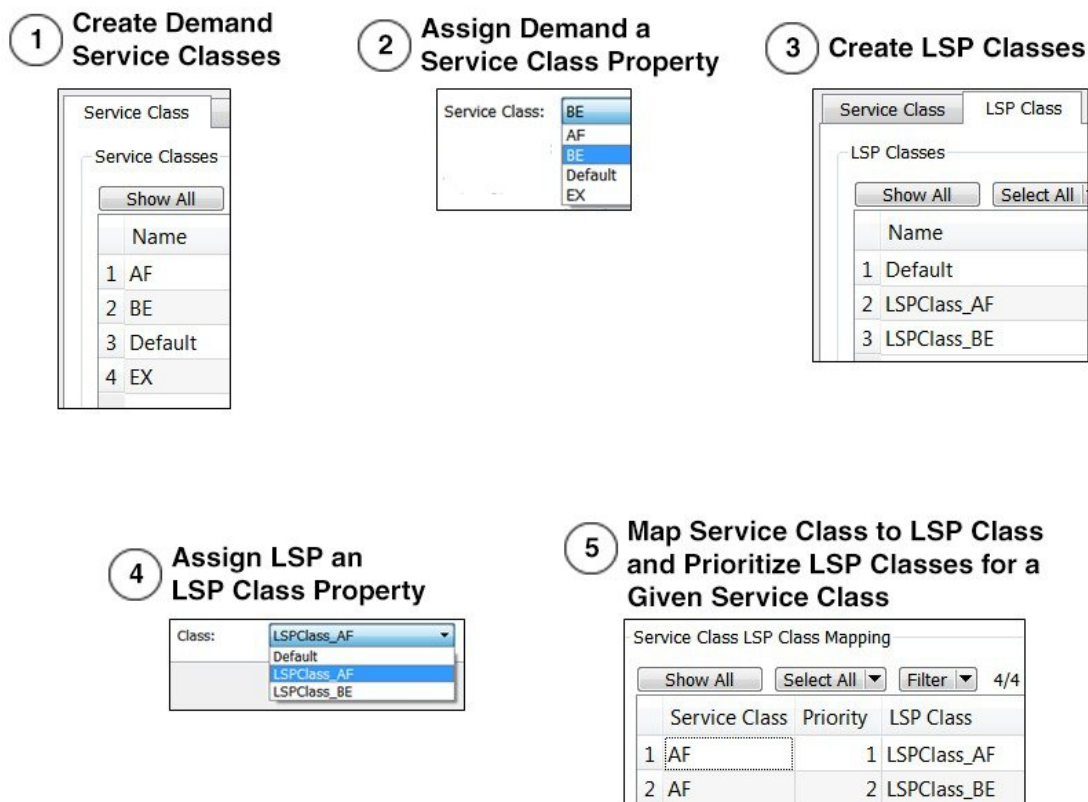
Workflow for Creating Service Class and LSP Class Mappings

The workflow for creating mappings between service classes and LSP classes is as follows (Figure 39: [Workflow Process for Creating Service Class to LSP Class Mappings, on page 125](#)).

-
- Step 1** Create the service classes using the Manage QoS dialog box. For information, see [Quality of Service Simulation, on page 239](#).
 - Step 2** Assign demands to the service classes. To assign this property, use the Service Class menu in the demand's Properties dialog box. (Right-click one or more selected demands and choose **Properties**.)
 - Step 3** Create the LSP classes using the Manage QoS dialog box. For information, see [Creating LSP Classes, on page 125](#).
 - Step 4** Assign LSPs to the LSP classes. For information, see [Assigning LSP Classes to LSPs, on page 125](#).
 - Step 5** As needed, assign a service class to one or more LSP classes using the Manage QoS dialog box. This creates a mapping table between the demands with a given service class and LSPs with a given LSP class. Optionally, prioritize multiple LSP classes within a service class. For information, see [Creating Mappings Between Service Classes and LSP Classes, on page 126](#).
-

What to do next

Figure 39: Workflow Process for Creating Service Class to LSP Class Mappings



381148

Creating LSP Classes

- Step 1** Open the Manage QoS dialog box in one of two ways:
- Choose **Edit > Manage QoS**.
 - Choose **Manage QoS** from the **QoS** drop-down menu in the Visualization toolbar.
- Step 2** Click the **LSP Class** tab.
- Step 3** In the LSP Classes area, click **New**.
- Step 4** Enter the name of the LSP class, and click **OK**.
- Step 5** Click **OK** in the Manage QoS dialog box.

Assigning LSP Classes to LSPs

- Step 1** Choose one or more LSPs.
- Step 2** Right-click one of the LSPs and choose **Properties**.

Step 3 Choose the LSP class from the Class drop-down list and click **OK**.

Creating Mappings Between Service Classes and LSP Classes

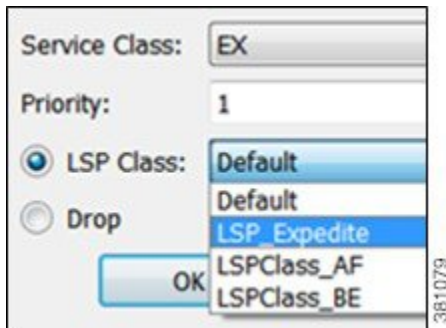
Step 1 Open the Manage QoS dialog box in one of two ways:

- Choose **Edit > Manage QoS**.
- Choose **Manage QoS** from the **QoS** drop-down menu in the Visualization toolbar.

Step 2 Click the **LSP Class** tab.

Step 3 In the Service Class LSP Class Mapping area, click **New**.

Step 4 In the Service Class drop-down list, select the service class to which you are mapping LSP classes.



Step 5 Enter a priority (number) for the LSP class. This determines the order in which the demand routes available LSPs. The lower the number, the higher the priority.

Step 6 Click one of the following radio buttons:

- To map the selected service class to an LSP class, click **LSP Class** and select the class from the drop-down list.
- To set the mapping so the demand doesn't reroute if LSPs are unavailable, click **Drop**. Drop should only be used on the lowest priorities.

Step 7 Click **OK**, and then click **OK** again in the Manage QoS dialog box.

Private LSPs

WAE Design provides two ways to route selected traffic demands through specific LSPs. One way is to dedicate the traffic for specific demands to a private LSP, which is a special LSP that carries those demands only. This type of LSP models an MPLS Layer 2 VPN, providing an exclusive route for the associated demands. If the LSP goes down, all traffic associated with the LSP is interrupted.

The other way to dedicate traffic to an LSP is to filter the demands that the LSP can carry. You can do this by using the WAE Design tag feature to identify the desired demands, and then configuring the LSP with a list of tagged values. This type of LSP makes it possible to route specific types of traffic over the LSP, and the type of traffic is any arbitrary group that you define.

To configure LSPs that simulate Layer 2 VPNs, use one of these two tools.

- One tool creates dedicated demands for existing LSPs. The created demands will match the LSPs in source and destination.
- One tool creates private LSPs from existing demands. The created LSPs will match the existing demands in source and destination.

The LSP Private property is set to T (true) in the LSPs table, and the Private LSP Node and Private LSP Name properties in the Demands table are set.

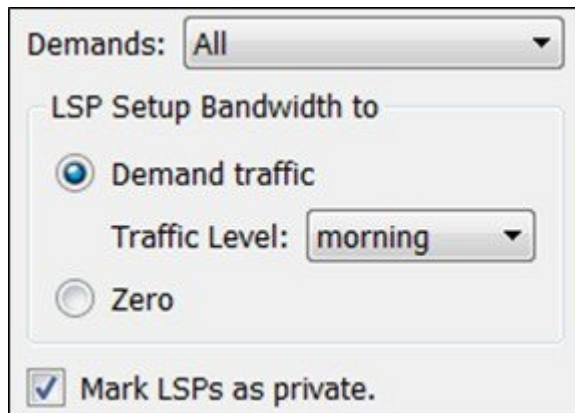
Creating Private Demands for Existing LSPs

LSPs must currently exist in the plan file.

-
- Step 1** If you are creating demands for only selected LSPs, select those LSPs from the LSP table. If they are tagged, you can skip this step and specify the tag value in Step 3 instead. If you are creating demands for all LSPs, you can skip this step.
- Step 2** Choose **Insert > LSPs > Demands for LSPs**, or right-click in the plot and choose **New > LSPs > Demands for LSPs**.
- Step 3** In the LSPs list, choose **All**, **Selected in table** (Step 1), or **Tag name**.
- Step 4** Select the service class to which these LSPs belong.
- Step 5** Set the demand traffic to equal the LSP setup bandwidth, the LSP traffic measurements, or zero.
- Step 6** Check **Mark LSPs as private**.
- Step 7** Click **OK**. The newly created demands are highlighted in the Demands table.
-

Creating Private LSPs for Demands

Demands must currently exist in the plan file. See [Traffic Demand Modeling, on page 43](#).



-
- Step 1** If you are creating LSPs for only selected demands, select those demands from the Demands table. If they are tagged, you can skip this step and specify the tag value in Step 3 instead.
- Step 2** Choose **Insert > LSPs > LSPs for Demands**, or right-click in the plot and choose **New > LSPs > LSPs for Demands**.
- Step 3** In the Demands list, choose **All**, **Selected in table** (Step 1), or **Tag name**.
- Step 4** Set the bandwidth traffic to a specific traffic level or to zero.
- Step 5** Check **Mark LSPs as private**.
- Step 6** Click **OK**. The newly created LSPs are highlighted in the LSPs table.
-

Deleting Demands when Private LSPs are Deleted

You can choose to delete a demand when deleting a Private LSP. By default, when a private LSP is deleted, the corresponding demand is not deleted.

-
- Step 1** Choose **Edit > Network Options**.
- Step 2** Click **Advanced** tab.
- Step 3** Under the Demands section, choose:
- Unrouted if the Private LSPs are unrouted
 - Removed if the Private LSPs are removed
- Step 4** Click **OK**.
-

Configuring LSPs to Transport Specific Traffic Types

This section describes how to configure LSPs that only route specific types of traffic. For example, you might want to route traffic for a voice service class through an LSP with a low latency, and route traffic for an Internet service class through another LSP with best-effort only. Such LSPs only carry the specified demands, but those demands might also be routed elsewhere.

Step 1

In the Demands table, select and double-click the demands that you want to group.

- a) Click the **Tags** tab.

If a tag exists that you want to use, toggle its F (false) value to T (true) to enable the tag for the selected demands. If a tag does not exist that you want to use, click **New Tag**.

Enter the name of the tag.

Click **OK**.

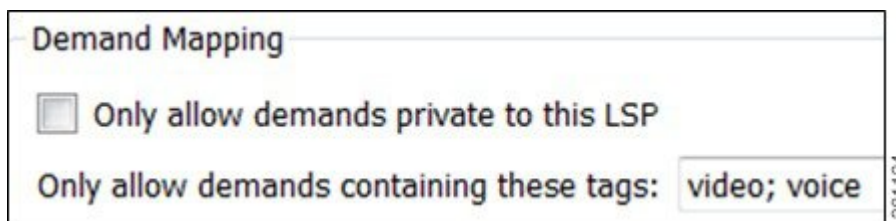
- b) Click **OK** to save and exit the Properties dialog box.

Step 2

In the LSP table, select and double-click the LSPs that are to transport demands that were tagged (Step 1).

- a) In the Include Any Demands Tags field, enter a list of demand tags and separate each entry with a semicolon.

- b) Click **OK**.



Routing Inter-Area LSPs

In WAE Design, all nodes in a single AS are assumed to belong to a single IGP. If the plan file contains more than one AS, all IGPs defined in these ASes are of the same type. For information on how nodes are assigned to areas or levels, see [Traffic Demand Modeling, on page 43](#).

An inter-area LSP is an LSP whose source node and destination node have no areas in common. If available, inter-area LSPs follow actual paths, regardless of whether doing so violates the required order of routing through areas. For example, if following an actual path, an inter-area can enter and leave an OSPF area 0 more than once.

Other factors that determine how the inter-area LSPs are routed include whether the LSP type is RSVP or SR, and whether you have selected to require explicit hops at ABRs. (See [Explicit Versus Dynamic Inter-Area LSP Routing, on page 130](#).)



Note Inter-area Fast Reroute LSPs and inter-area IGP shortcut LSPs are not supported. If the source and destination nodes of a Fast Route LSP are in different areas, the LSP is not routed.

Order of Routing Through Areas

Regardless of the LSP type and whether explicit hops are required, inter-area LSPs route through the backbone areas, as follows, where “backbone” means area 0 for OSPF or the Level 2 area for IS-IS.

- If there are three or more areas, backbone areas must be between the source and destination nodes. Typically, there are no more than three areas and therefore, the backbone area must be in the middle. For example, an OSPF inter-area LSP would route from area 1 to area 0 (backbone) and then to area 2 ([Figure 40: Example Inter-Area Routing Requiring Hops at ABRs, on page 131](#)).
- If there are only two areas, there can only be one backbone area, and either the source or the destination node must be in the backbone area.

Explicit Versus Dynamic Inter-Area LSP Routing

An OSPF ABR is a node that belongs to both area 0 and other OSPF areas. An IS-IS ABR is a node that belongs to both the Level 2 area and another IS-IS level.

There are two modes of routing inter-area LSPs. One requires that explicit hops be set on ABR nodes. This mode correctly simulates actual router behavior where ABR explicit hops are required. The other mode does not require explicit hops at ABR nodes and can route an LSP fully dynamically across multiple areas. While this mode does not simulate actual router behavior, it is useful for planning inter-area LSP routes. For example, you can use a fully dynamic simulation to find an appropriate path, which can then be converted into a (routable) fully explicit path using other tools such as the Explicit LSP Paths initializer. These modes are specified using the network option labeled “LSP routing requires ABR explicit hops.”

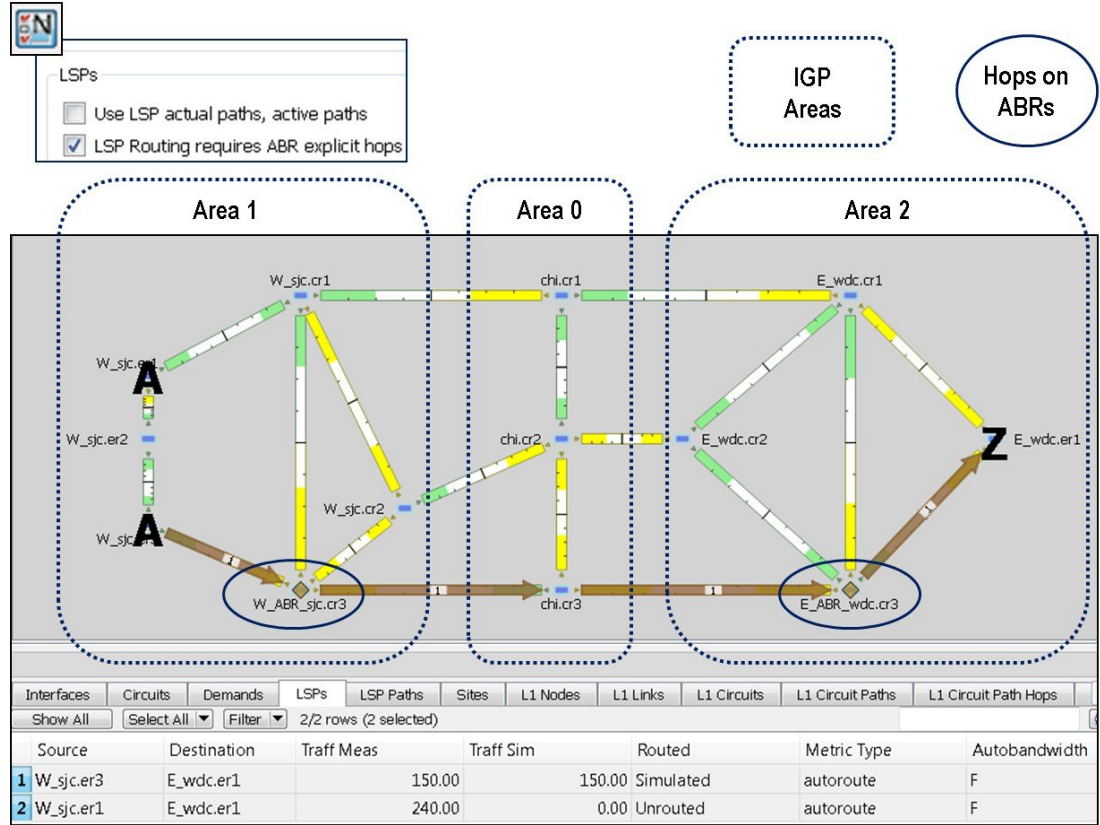
If this option is selected, inter-area LSPs are routed based on explicit hops set on the ABR nodes.

- An inter-area RSVP LSP must contain a named path, and the named path must contain explicit hops at ABRs for each required area crossing.
- An inter-area SR LSP must contain a segment list, and the segment list must contain explicit node hops at ABRs for each required area crossing.

If this option is not selected, inter-area LSPs are routed dynamically and explicit hops at ABRs are not required. To leave one area and enter another, the inter-area LSP routes to the closest ABR in the current area that also borders the area it is entering.

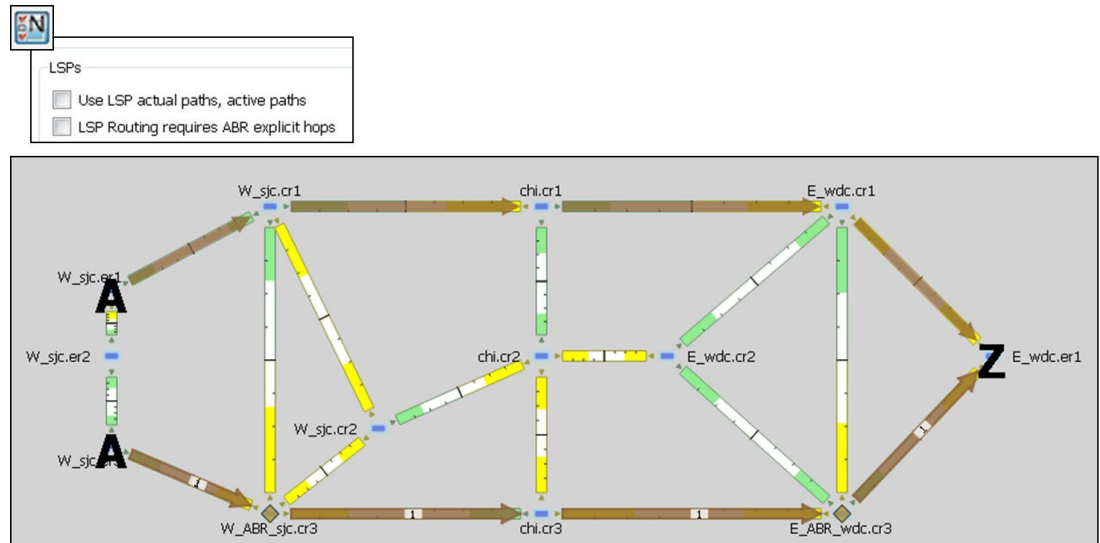
Example: [Figure 40: Example Inter-Area Routing Requiring Hops at ABRs, on page 131](#) shows two inter-area LSPs selected, although only the LSP with explicit hops set on ABR nodes routes. The reason is because the network option is set to require ABR hops when routing inter-area LSPs. [Figure 41: Example Inter-Area Routing Without Requiring ABRs, on page 131](#) shows that by deselecting this network option, the inter-area LSP that does not have hops set at ABRs can dynamically route.

Figure 40: Example Inter-Area Routing Requiring Hops at ABRs



404548

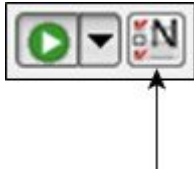
Figure 41: Example Inter-Area Routing Without Requiring ABRs



404547

Simulation Modes

WAE Design lets you set global parameters that affect how LSPs are routed or rerouted. To access these options, choose **Edit > Network Options** or click the **Network Options** icon in the Visualization toolbar. Then click the **Simulation** tab.



By default, WAE Design simulates the state of the network once it has fully responded to a failure. Specifically, this is the network state after LSPs have re-established new routes around the failure, and the IGP has fully reconverged. This is called the *IGP and LSP Reconvergence* simulation mode.

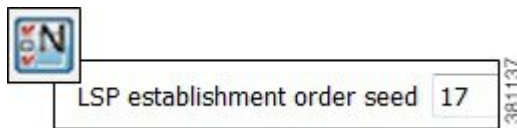
Other simulation modes include Fast Reroute (FRR), Autobandwidth Convergence, and Autobandwidth Convergence Including Failures. For information, see [RSVP-TE Simulation, on page 135](#). For information on the network option for rerouting of non-standby L1 circuit paths under failure, see [Layer 1 Simulation, on page 191](#).

The Optimization tools only work in IGP and LSP Reconvergence mode. If you try to run one while in a different convergence mode, you are prompted whether to continue, and if you do, the simulation changes to IGP and LSP Reconvergence mode.

The status bar in the lower left of the GUI identifies which simulation modes you are currently using. Note that unless specified otherwise, the documentation describes behavior in the IGP and LSP Reconvergence mode.

LSP Establishment Order

WAE Design establishes LSPs in the order in which they appear in the plan. The routing of a specific LSP might depend on previously established LSP routes. You can modify this order by changing a random seed. WAE Design then establishes LSPs in a random order that is determined by this number. Although you cannot predict the order based on the number, if you use the same number multiple times, WAE Design establishes the LSPs in the same order each time. Varying the LSP establishment order lets you check, for example, whether certain orders result in higher utilizations.



-
- Step 1** Choose **Edit > Network Options** or click the **Network Options** icon in the Visualization toolbar.
 - Step 2** Click the **Simulation** tab.
 - Step 3** In the LSPs area, enter a number in the LSP establishment order seed field. The default is 0.
 - Step 4** Click **OK**.
-

Reports and Diagnostics

LSP Path Reports

You can create reports on the LSP path routes that the LSP is actually taking across both L3 and L1 topologies. Each report identifies the complete path route, as well as disjointness information, including whether it contains common nodes or common circuits. Additionally, the report contains all other columns in an LSP Paths table, though they are initially hidden.

-
- Step 1** Select one or more LSP paths. If you do not select one, all LSP paths are used.
 - Step 2** Choose **Tools > Reports > LSP Path Routes**.
 - Step 3** Check one or both hop options, and click **OK**.
 - Step 4** To export the report, right-click the **Routes** report name in the left Reports navigation pane, and choose **Export Routes**.
-

Simulation Diagnostics

To help troubleshoot RSVP LSP and LSP path simulations, WAE Design analyzes simulation routes and provides reasons for certain types of routing behavior. The Simulation Diagnostics tool identifies why an LSP is not routed, why an LSP is routed away from its actual path, and why an LSP is not following the shortest TE path.

These routing diagnostics assume that all other LSPs, except for the one being tested, have been routed. That is, WAE Design calculates whether an LSP can route on the actual path after all other routed LSPs have had their bandwidth reserved.

Note that running a report overwrites the previous report of the same type. For example, an LSPs diagnostic report would overwrite the previous LSPs report, but not the LSP Path diagnostic report.

Using Simulation Diagnostics to Troubleshoot

The following information is provided to help you troubleshoot LSP and LSP path issues.

Reason	Description
Affinities	The affinity settings prevent the LSP or LSP path from routing.
Available BW	There is insufficient bandwidth to route the LSP or LSP path.
Explicit Hops	The route is determined by a named path that contains one or more explicit hops.
Hop Limit	The hop limit is too low.
Invalid Actual Path	The actual path is invalid and cannot be interpreted as an LSP route.
No Actual Path	The LSP does not have an actual path.

Reason	Description
No Attempt	The LSP path is not routed because it is not a standby path and it is not the lowest routable path option for the LSP.
No Destination	There is no destination defined. The collected network might not contain the destination node of the LSP.
Simulation Option	The simulation option for following actual paths is not enabled.
TE not enabled	There are interfaces that the LSP or LSP path traverses that are not TE enabled.
Topology	The source and destination are disconnected.

Running LSP Simulation Diagnostics

Step 1 Choose one or more LSPs or LSP paths from their respective tables.

Step 2 Open the diagnostics information in one of two ways:

- Choose **Tools > Diagnostics > LSP Simulation** or **LSP Path Simulation**.
 - Right-click the selection in the table and choose **Simulation Diagnostics**.
-



CHAPTER 10

RSVP-TE Simulation

This chapter describes how WAE Design simulates RSVP-TE routing. Unless otherwise noted, *LSPs* refer to RSVP-TE LSPs. Note that WAE Design does not distinguish between LDP and RSVP-TE LSPs.

This section contains the following topics:

- [Dynamic LSP Routing and CSPF](#), on page 135
- [RSVP LSP Paths](#), on page 138
- [Named Paths and Explicit LSP Routing](#), on page 140
- [Actual Paths](#), on page 143
- [Affinities](#), on page 144
- [Simulation Modes](#), on page 148
- [Advanced RSVP-TE LSP Simulations](#), on page 163

Dynamic LSP Routing and CSPF

If an LSP contains no LSP paths (also called *MPLS TE tunnel paths*), it is routed dynamically using CSPF. The weights used for the CSPF calculation are the TE metrics per interface. If the TE metric is not configured for an interface, the IGP metric is used.

If there are equal-cost routes, the following selection criteria apply in this order:

- Bandwidth available—The route with the greatest reservable bandwidth is chosen.
- Hop count—The route with the fewest hops is chosen.
- Random—If neither of the above criteria can be used, the route is randomly chosen.

CSPF properties are set in the LSP Properties dialog box and are viewable from the LSPs table.

CSPF	
Setup bandwidth:	<input type="text"/> <input checked="" type="checkbox"/> Autobandwidth
Setup priority:	<input type="text" value="7"/> <input type="text" value="7"/> Hold priority:
Hop limit:	<input type="text" value="2"/> <input type="checkbox"/> TE Metric Disabled

- Setup Bandwidth—The amount of traffic the source node requests for this LSP in Mbps. The requested bandwidth is available from the reservable bandwidth of each interface in the path.

- **Autobandwidth**—Dynamically update the Setup BW Sim value when using the Autobandwidth Convergence mode.
- **Setup Priority**—A priority for allocating reservable bandwidth. This is the order in which the LSPs are signaled. The lower the number, the higher the priority.
- **Hold Priority**—Priority that can preempt LSPs that are on the shortest path. This is typically set for a particular service or traffic type. The lower the number, the higher the priority.
- **Hop Limit**—The maximum number of hops permitted in the LSP route. If no path is available with this number of hops or fewer, the LSP is not routed.
- **TE Metric Disabled**—If checked, the LSP is routed using IGP metrics. If unchecked (default), the LSP is routed using TE metrics.

Interface MPLS Properties

Related MPLS properties are set in the MPLS tab of the interface or circuit Properties dialog box, and are viewable in the Interfaces table.

The screenshot shows a dialog box with five tabs: Basic, Advanced, MPLS (selected), User, and Tags. Below the tabs is a section titled 'Selected Interfaces (1/32)'. Inside this section, there are four rows of configuration fields:

- Reservable BW:** An empty text input field.
- Reservable BW (%):** A text input field containing the value '25'.
- TE Metric:** A text input field containing the value '10'.
- Affinities:** A text input field containing the value '0'. To the right of this field is a button labeled 'Edit...'.

At the bottom of the dialog box, there are two checked checkboxes: TE Enabled and FRR Enabled. A vertical ID number '381119' is visible on the right side of the dialog box.

- **TE Metric**—Determines which path an LSP takes.
- **TE Metric Sim**—Derived column that shows the effective TE metric. If the TE Metric is empty, this is set to the IGP metric.
- **TE Enabled**—Identifies whether an LSP can be routed over the interface. If checked, the value is set to T (True), and TE metrics are used in routing the LSP provided the LSP's TE Metric Disabled is unchecked (False).
- **Resv BW**—Shows the amount of bandwidth that can be reserved by LSPs routed over the interface.
- **Resv BW Percent**—Shows the percentage of bandwidth that can be reserved by LSPs routed over the interface.
- **Resv BW Sim**—Derived column.
 - If a Resv BW value is entered, this is copied to the Resv BW Sim column.

- If Resv BW and Resv BW Percent are “na,” the Resv BW Sim is copied from the Capacity Sim column.
- If Resv BW is “na,” but Resv BW Percent has a value, the Resv BW Sim value is derived by this formula:

$$\text{Capacity Sim} * (\text{Resv BW Percent} / 100)$$

If needed, you can change this behavior so that reserved bandwidth constraints are not used during simulations, which could be useful for planning purposes. For information, see [Ignoring Reservable Bandwidths for Capacity Planning, on page 163](#).

- Affinities—Lists which affinities are set. For more information, see [Affinities , on page 144](#).
- Loadshare—Value for redistributing either traffic across LSPs based on the ratio of this value with other LSPs that are parallel (have the same source and destination).
- FRR Enabled—Designated interface to be avoided by the FRR LSP. Only interfaces with this property set are used by the FRR LSPs initializer when creating FRR LSPs. This property does not affect FRR LSPs that are manually created. For more information, see [Fast Reroute Simulations, on page 149](#).

To determine how much of the interface traffic is contained in LSPs, show these columns in the Interfaces table. The total of the two equals Traff Sim.

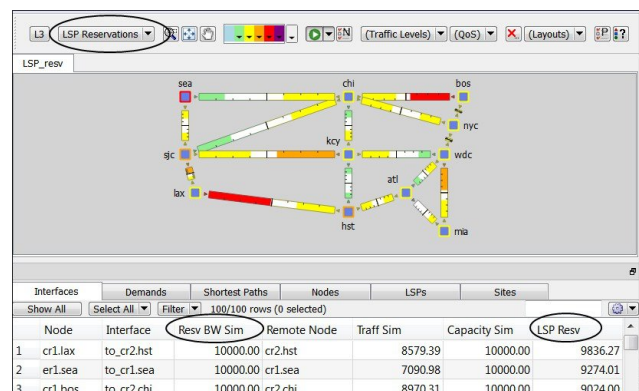
- Traff Sim LSP—The total amount of LSP traffic on the interface.
- Traff Sim Non LSP—The total amount of non-LSP traffic on the interface.

To determine which interfaces carry LSP traffic, sort the columns. To determine the demands associated with one or more selected interfaces based on whether or not they have LSP traffic, right-click one and choose **Filter to Demands Through Interfaces**. Then choose either **In LSP** or **Not in LSP**

Viewing LSP Reservations

To view the LSP reservations on the interfaces, choose **LSP Reservations** from the Network Plot View in the Visualization toolbar ([Figure 42: LSP Reservations View, on page 137](#)). The width of the circuit displayed in the network plot is set according to the Resv BW Sim column in the Interfaces table. The utilization colors are based on the LSP Util column in the Interfaces table. The LSP Util is the sum of all setup bandwidths for all LSPs through the interface, as a percentage of the Resv BW Sim.

Figure 42: LSP Reservations View



LSP Setup BW Initializer

The LSP Setup BW Initializer calculates the setup bandwidth for existing RSVP-TE LSPs. The setup bandwidth is set to the maximum amount of demand traffic passing through the LSP for the traffic levels selected.

If using only one traffic level, the best practice is to use the Autobandwidth Convergence mode. For information, see [Advanced RSVP-TE LSP Simulations, on page 163](#).

-
- Step 1** Choose one or more LSPs from the LSPs table.
 - Step 2** Choose **Initializers > LSP Setup BW**, or right-click one of the selected LSPs and choose **LSP Setup BW Initializer**.
 - Step 3** Choose which traffic levels to use for demand traffic.
 - Step 4** Choose whether to set the load share values to be equal to the setup bandwidth.
 - Step 5** Click **OK**.
-

RSVP LSP Paths

Like LSPs, LSP paths have CSPF properties, such as Setup Priority and Hop Limits. For a description of LSP properties, see [Dynamic LSP Routing and CSPF, on page 135](#).

If these properties are omitted, then they are inherited from the LSP. If these properties are set in the LSP path, they override the LSP settings. Note, however, if operating in the Autobandwidth Convergence simulation mode and if the LSP has Autobandwidth set to T (true), the LSP path Setup BW property (like the LSP Setup BW) is ignored, and the LSP's Setup BW Sim value is calculated and used.

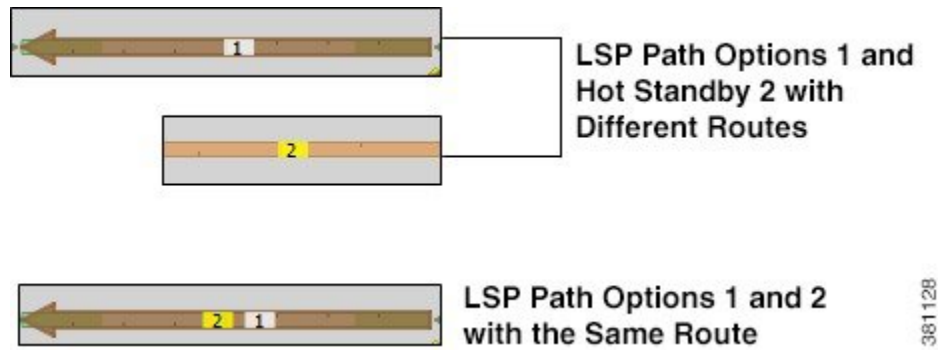
If an LSP path does not have an associated named path, it is routed dynamically. However, you can use named paths to fully or partially describe a path from the LSP source to destination.



Note This chapter refers to LSPs and LSP paths as *dynamic*, *explicit*, or *loose explicit*, depending on whether their associated named paths have no defined hops, all hops defined, or only some hops defined.

Hot Standby

An LSP path can be defined as a “hot” standby path. Standby paths are always established even if the associated LSP is routed using a path with a different path option. For RSVP LSPs, setup bandwidth, if any, is reserved for them. These standby LSP paths are then immediately available should the currently routed path become unavailable.



381128

Example Response to Failures

Figure 43: Example RSVP LSP and Associated LSP Paths, on page 139 shows an RSVP LSP (cr1.lon_cr2.fra) that uses four LSP paths. The cr1.lon_cr2.fra LSP has a primary and standby secondary path options (100 and 200), each with a setup bandwidth of 0. They are both established, using their defined named paths. Both primary and secondary LSP paths have associated named paths, which appear in the Named Path column. The other two paths do not have associated named paths, and so are dynamically routed.

Figure 43: Example RSVP LSP and Associated LSP Paths

Interfaces	Demands	Shortest Paths	Nodes	LSPs	LSP Paths	Named Paths
Show All	Select All	Filter	1/1 rows (0 selected)			
Name	Source	Destination	Setup BW	Active Path	Named Paths	
1 cr1.lon_cr2.fra	cr1.lon	cr2.fra	125.00	100	cr1.lon_cr2.fra_100	

Interfaces	Demands	Nodes	LSPs	LSP Paths	Sites	SRLGs	AS
Show All	Select All	Filter	4/4 rows (0 selected)				
LSP	Path Name	Setup BW	Path Option	Standby			
1 cr1.lon_cr2.fra	cr1.lon_cr2.fra_100	0.00	100	T			
2 cr1.lon_cr2.fra	cr1.lon_cr2.fra_200	0.00	200	T			
3 cr1.lon_cr2.fra		na	300	F			
4 cr1.lon_cr2.fra		0.00	400	F			

381101

This example LSP responds to failures as follows:

1. If the primary path (cr1.lon_cr2.fra_100) fails, the secondary path (cr1.lon_cr2.fra_200) is used.
2. If both paths with associated name paths fail, a dynamic path is established with a reserved bandwidth of 125 Mbps, which it inherits from the LSP. This path is not set up initially because it is not a standby.
3. If all paths fail, the path of last resort is a dynamic path with zero bandwidth. This ensures that there is always an LSP up so that services relying on the LSP are always available.

- The simulation changes if the LSP Active Path has been set. For example, if `cr1.lon_cr2.fra_200` is the active path (that is, 200 appears in the LSP's Active Path column), WAE Design uses `cr1.lon_cr2.fra_200` first, and then goes through the previous sequence.

Named Paths and Explicit LSP Routing

Named paths specify a route through the network using an ordered list of adjacencies. The route is defined by named path hops, which can be nodes or interfaces, and each hop type specifies whether the route should be strict, loose, or excluded. Named path hops can be nodes or interfaces. The hop type is identified visually on the plot, as well as listed in the Type column of the Named Path Hops table.

- **Strict**—The LSP must reach the named path hop directly from the previous one, with no intermediate interfaces.
- **Loose**—The LSP is routed to this hop from the previous hop using CSPF. Intermediate interfaces can be used.
- **Exclude**—The node or interface is excluded from the LSP path. This hop type cannot be combined with strict or loose hops in the same named path.



The named paths and their hops are listed and selectable from the Named Paths and Named Path Hops tables. The advantage of having separate tables is that you can have named paths that have missing or unresolved hops. Also, the path name can be reserved even if it is not part of the WAE Design simulation.

Named Path Hops Example

Figure 44: Example Named Paths for the `cr1.lon_cr2.fra` LSP, on page 141 extends the earlier example (Figure 43: Example RSVP LSP and Associated LSP Paths, on page 139), showing two named paths for the `cr1.lon_cr2.fra` LSP. This shows that the first two LSP paths each have a named path. The naming convention is `<LSP_Name> _<Path_Option>`, which in this case is `cr1.lon_cr2.fra_100` and `cr1.lon_cr2.fra_200`.

Figure 44: Example Named Paths the cr1.lon_cr2.fra LSP

Interfaces		LSPs		LSP Paths		Named Paths	
Show All		Select All ▼		Filter ▼		2/2 rows (0 sele	
	Name	Source	Active				
1	cr1.lon_cr2.fra_100	cr1.lon	T				
2	cr1.lon_cr2.fra_200	cr1.lon	T				

The hops for each named path are defined in a separate table called Named Path Hops. [Figure 45: Example Named Path Hops for cr1.lon_cr2.fra_100, on page 141](#) shows the named path hops for the cr1.lon_cr2.fra_100 named path. The Steps column shows the hop order.

- The first hop is an interface hop: a strict hop on the interface from cr1.par to cr2.par.
- The second is a node hop, also strict, to cr2.par.
- The third hop is an interface hop: a strict hop on the interface from cr1.fra to cr2.fra.

Hops for cr1.lon_cr2.fra_200 are defined similarly.

Figure 45: Example Named Path Hops for cr1.lon_cr2.fra_100

Interfaces		LSPs		LSP Paths		Named Paths		Named Path Hops		Site
Show All		Select All ▼		Filter ▼		3/3 rows (0 selected)				
	Name	Source	Step	Node	Interface	Type				
1	cr1.lon_cr2.fra_100	cr1.lon	1	cr1.par	{to_cr2.par}	strict				
2	cr1.lon_cr2.fra_100	cr1.lon	2	cr2.par		strict				
3	cr1.lon_cr2.fra_100	cr1.lon	3	cr1.fra	{to_cr2.fra}	strict				

Creating Named Paths and Their Hops

Named paths can be created in the WAE Design GUI in several ways.

Create associated named paths: \$1_\$2
 (\$1 is LSP Name, \$2 is Path Option)

- Check **Create associated named paths** when creating LSP paths. Using this option does not create hops.
- Use the Explicit LSP Paths Initializer. Using this initializer creates named paths with strict hops.
- Optimize LSP paths using the Explicit RSVP-TE LSP Optimization tool or RSVP-TE Optimization tool.

Editing Named Paths and Named Path Hops

Once the named paths are created, you can create, edit, or delete their named path hops. Note that discovered named paths can contain *unresolved* hops, which are nodes and interfaces that are not in the plan file. For information on resolving named paths, see [Unresolved LSP Destinations and Hops, on page 176](#).

The recommended method of editing the named path hop type is to access it directly from the Named Path table as described in the following steps. This is the most efficient way of viewing the entire path. However, if needed, you can double-click a named path hop in the Named Path Hops table, which takes you directly to the named path hops Properties dialog box, which is described in Step 4.

Step 1 In the Named Paths table, double-click a named path or right-click one and choose **Properties**.

Step 2 Named path options:

- To change the name, enter it in the Name field.
- To change the source site or source nodes, choose them from the drop-down lists.
- To activate or deactivate, check **Active**.

Step 3 Hop options:

- To add a new hop, choose an existing hop (if applicable) from the Hops list, and then click **Insert Before** or **Insert After** to determine where the new hop goes sequentially. Go to Step 4.
- To edit an existing hop, choose it from the Hops list and then click **Edit**. The Properties dialog box opens for named path hops. Go to Step 4.
- To delete an existing hop, choose it from the Hops list, click **Delete**, and then **OK**. You need not continue.
- To resolve nodes and interfaces that do not exist in the plan file, click **Resolve**. For information, see [Resolving Plan File to Objects, on page 177](#).

Step 4 To continue creating or editing the named path hop, use the following options and then click **OK**

- As needed, choose the site, node, and interface. For node hops, do not choose an interface.
- Choose the named path hop type as Loose, Strict, or Exclude. See [Named Paths and Explicit LSP Routing, on page 140](#) for a description of each.

Actual Paths

In WAE Design, actual paths are the actual route that routed LSP paths are taking. They are read from the live network, per LSP path. RSVP-TE routing is not always completely predictable because it depends on the order in which LSPs are established and reserve their bandwidth on the network. WAE Design uses actual paths in LSP routing simulations to match the current LSP routing state of a network as accurately as possible.

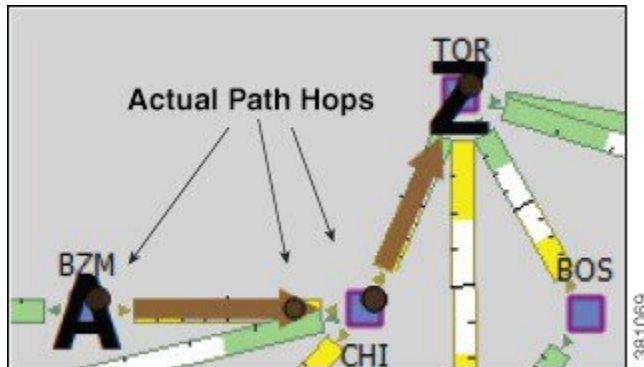
When an LSP path is routed, the actual route is attempted first. If this routing fails due to CSPF constraints, such as not enough reservable bandwidth or affinity restrictions, WAE Design reverts to the standard CSPF routing algorithm.

Example: WAE Design routes all LSPs in its simulation according to the actual paths read from the network, and the routings completely match the network. If a circuit fails, only the LSPs through this circuit are rerouted. Because they cannot be established along their actual paths, standard CSPF is used. The result is a prediction of incremental routing changes that would occur on the current network if such a failure were to occur.

It might be desirable to simulate a network without taking actual paths into account, for example, for long-term planning. To achieve this, actual paths should simply be deleted. To delete an actual path, right-click an LSP or LSP path and choose **Delete Actual Paths**.

An LSP or an LSP path might have a corresponding actual path. An LSP path with no actual path inherits its LSP actual path, if available.

Actual paths are represented as a series of interface hops in the Actual Path Hops table. A hop is shown in the network plot as a brown circle on the interface. To view an entire actual path of an LSP, choose **View > LSPs > Actual Paths**; all hops are displayed when the LSP is selected.



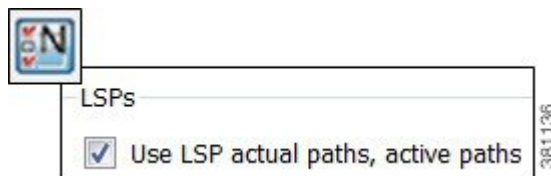
Two columns in the LSP table are useful to see the result of actual paths on LSP routing.

Column	Description
Actual	<p>The simulation can only use actual paths if they are resolved. If the network discovery was incomplete, this might not be possible.</p> <ul style="list-style-type: none"> • T = true; the simulation converted the actual path hops into a complete path from source to destination of the LSP • F = false; the simulation did not convert the actual path hops into a complete path • na = not applicable; no actual path was available.

Column	Description
Route	<ul style="list-style-type: none"> • Actual = The LSP follows an actual path. • Simulated = The LSP does not follow an actual path. • Unrouted = Routing was not possible.

Deactivating Actual Paths for Simulations

WAE Design uses network state in its MPLS simulation, routing LSPs on actual paths where possible and using LSP active path settings. For planning purposes, when state is not relevant, you can change this behavior to disregard actual paths.



-
- Step 1** Choose **Edit > Network Options** or click the **Network Options** icon in the Visualization toolbar.
- Step 2** Click the **Simulation** tab.
- Step 3** To use or disregard actual paths and active paths in MPLS simulation, check or uncheck **Use LSP actual paths, active paths**.
- Step 4** Click **OK**.
-

Affinities

Affinities provide a mechanism for implementing LSP path diversity. By assigning affinities to physical circuits and associating LSPs with affinities, you can implement a variety of routing policies. For example, you can use affinities to restrict certain traffic to specific topological regions. Or you can force primary and backup LSP paths onto different routes so they do not simultaneously fail.

WAE Design supports an unlimited number of 64-bit affinities, each of which is defined by a number and an optional name. Named affinities are sometimes called *administrative groups* or *link coloring*.

The default plan files have 0-31 unnamed and unassigned affinities.

Workflow:

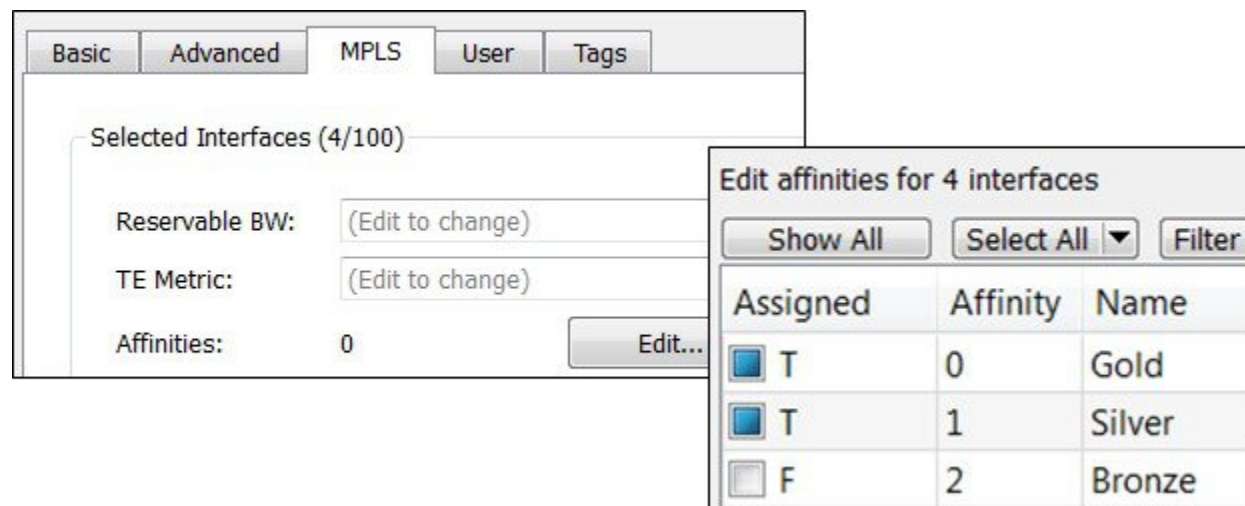
1. [Creating and Editing Affinities](#), on page 145.
2. [Assigning Affinities to Interfaces](#), on page 145.
3. [Associating LSPs with Affinities](#), on page 146.

Creating and Editing Affinities

- Step 1** Choose **Edit > Admin Groups**.
- Step 2** Modify an existing affinity number and name, or click **New** to add the number and name.
Affinity numbers must be unique. Affinity names are optional and must also be unique.
- Step 3** Click **OK**.

Assigning Affinities to Interfaces

To use affinities, you must assign them to interfaces in a way that encourages favorable routes and discourages others. For example, continental paths might have one affinity and international paths have another.



- Step 1** From the Interfaces table, choose one or more interfaces to which you are assigning the affinities.
- Step 2** Double-click the selection or choose **Edit > Properties**.
- Step 3** Click the **MPLS** tab.
- Click the **Affinities Edit** button.
 - In the Include column, check the boxes for the affinities that you want to associate with the selected circuits. T = true; associate the affinity with the circuit. F = false; do not associate the affinity.
 - Click **OK**.
- Step 4** Click **OK**.

You can follow this procedure to assign affinities to interfaces through a circuit Properties dialog box. However, you must choose and assign the affinity twice: once to each interface.

Associating LSPs with Affinities

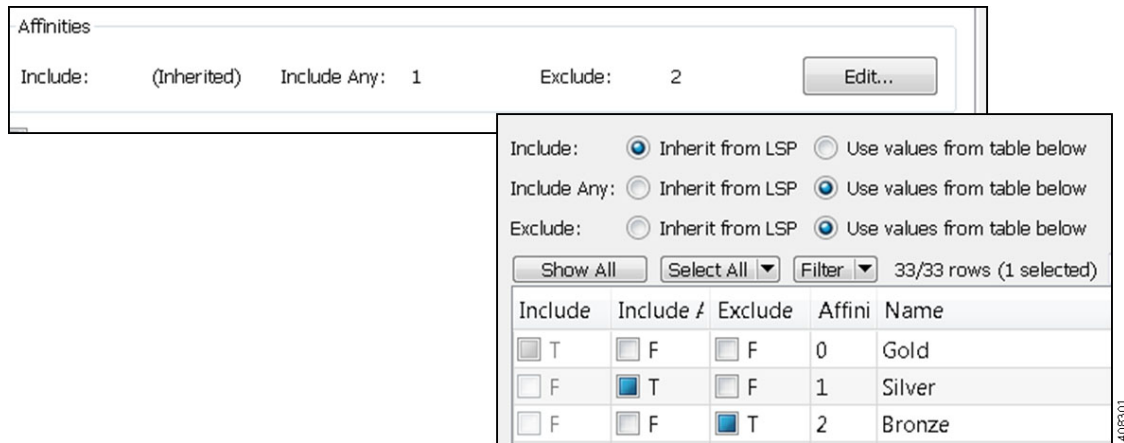
For affinities to influence LSP routing, you must associate LSPs with the affinities assigned to interfaces that you want to include or exclude in the routing process. If no affinity is specified, the LSP can be routed through any path. Explicit routes can also have affinities for hops with a Loose relationship. (See [Named Paths and Explicit LSP Routing](#), on page 140.)

Inclusion and Exclusion Rules

- Include—Only use interfaces that include all affinities.
- Include Any—Use interfaces that include at least one affinity.
- Exclude—Do not use interfaces with this affinity.

LSP paths inherit the LSP affinities. However, you can edit LSP path affinities to have priority over the LSP affinities.

Figure 46: LSP Path Affinities

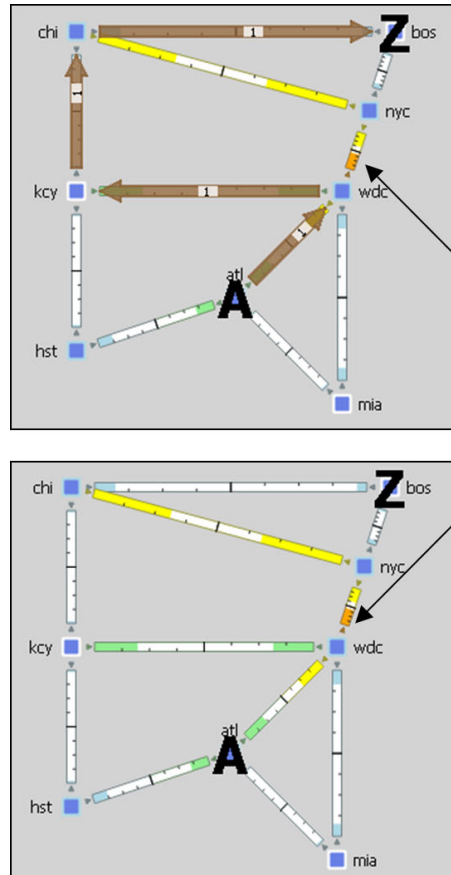


Example

This example (Figure 47: Example Affinities, on page 147) demonstrates how affinities affect the routing of two LSPs with the same source and destination nodes.

- The to_cr1.nyc interface between wdc and nyc is assigned to both affinities 0 (Gold) and 1 (Silver). No other interfaces are assigned affinities.
- LSP A is configured to exclude all interfaces assigned to affinity 0. While it cannot take the shortest path using the to_cr1.nyc interface, it can route around it.
- LSP B is not associated with any affinities. Its LSP path is configured to include any interfaces assigned to affinity 1 (Silver) and to exclude interfaces assigned to affinity 2 (Bronze). Because the to_cr1.nyc interface is assigned to both of these affinities, LSP B cannot route.

Figure 47: Example Affinities



LSP A

Include	Include Any	Exclude	Affinity	Name
<input type="checkbox"/> F	<input type="checkbox"/> F	<input type="checkbox"/> F	0	Gold
<input type="checkbox"/> F	<input type="checkbox"/> F	<input type="checkbox"/> F	1	Silver
<input type="checkbox"/> F	<input type="checkbox"/> F	<input checked="" type="checkbox"/> T	2	Bronze

Interfaces		
Node	Interface	Affinities
cr2.wdc	to_cr1.nyc	Silver;Bronze

LSP B

Include	Include Any	Exclude	Affinity	Name
<input type="checkbox"/> F	<input type="checkbox"/> F	<input type="checkbox"/> F	0	Gold
<input type="checkbox"/> F	<input type="checkbox"/> F	<input type="checkbox"/> F	1	Silver
<input type="checkbox"/> F	<input type="checkbox"/> F	<input type="checkbox"/> F	2	Bronze

LSP Path for LSP B

Include	Include Any	Exclude	Affinity	Name
<input type="checkbox"/> F	<input type="checkbox"/> F	<input type="checkbox"/> F	0	Gold
<input type="checkbox"/> F	<input checked="" type="checkbox"/> T	<input type="checkbox"/> F	1	Silver
<input type="checkbox"/> F	<input type="checkbox"/> F	<input checked="" type="checkbox"/> T	2	Bronze

408295

Assigning LSPs to Affinities

- Step 1** From the LSPs table, choose one or more LSPs to which you are associating the affinities.
- Step 2** Double-click the selection or choose **Edit > Properties**.
- Step 3** Click the **Affinities** tab.
- Step 4** Choose the inclusion or exclusion rule for each affinity you are associating with the selected LSPs.
- Step 5** Click **OK**.

Assigning LSP Paths to Affinities

- Step 1** From the LSP Paths table, choose one or more LSP paths to which you are associating the affinities.
- Step 2** Double-click the selection or choose **Edit > Properties**.
 - a) Click the **Affinities Edit** button. For each rule (Include, Include Any, Exclude), choose whether the LSP path should inherit the LSP affinity rule or whether it should be based on rules defined in the table below these options.
 - b) If you chose to use the table, choose the rule for each affinity you are associating with the LSP path.

c) Click **OK**.

Step 3 Click **OK** to save and edit.

Assigning Affinities When Creating LSP Meshes

You can associate affinities with LSPs when creating a new LSP mesh.

Step 1 Choose **Insert > LSPs > LSP Mesh**, or right-click in the plot and choose **New > LSPs > LSP Mesh**.

Step 2 Click the **Affinities Edit** button. Choose the inclusion or exclusion rule for each affinity you are associating with all LSPs in the mesh, and click **OK**.

Step 3 Click **OK** to save and exit.

Simulation Modes

WAE Design lets you set several global parameters that affect how LSPs are routed or rerouted. To access these options, either choose **Edit > Network Options** or click the **Network Options** icon in the Visualization toolbar. Then click the **Simulation** tab.



WAE Design supports four simulation modes for RSVP-TE LSPs:

- A *Fast Reroute* (FRR) mode using FRR LSPs, which is a common failure restoration mechanism used by RSVP-TE LSPs. In a live network, FRR restoration typically occurs within milliseconds. See [Fast Reroute Simulations, on page 149](#).
- By default, WAE Design simulates the state of the network once it has fully responded to a failure. Specifically, this is the network state after LSPs have re-established new routes around the failure, and the IGP has fully reconverged. This is called the *IGP and LSP Reconvergence* simulation mode. The Optimization tools only work in IGP and LSP Reconvergence mode.
- In the Autobandwidth Convergence mode, the network is simulated after the traffic (Traff Sim) has been rerouted to other LSPs, but before the Setup BW Sim values have been reset. See [Autobandwidth Simulations, on page 159](#).
- In the Autobandwidth Convergence Including Failures mode, the network is simulated after the Setup BW Sim values have been reset. See [Autobandwidth Simulations, on page 159](#).

The status bar in the lower, left of the GUI identifies which simulation modes you are currently using. Unless specified otherwise, the documentation describes behavior in the IGP and LSP Reconvergence mode.

Fast Reroute Simulations

WAE Design simulates a Fast Reroute convergence mode using FRR LSPs. If a failure occurs in the route of a protected RSVP-TE LSP, a presignaled FRR LSP (or bypass tunnel) forwards traffic locally around the point of failure, typically from the node just before the failure to a node downstream. This restoration mechanism gives the source node (head-end) of the FRR LSP time to re-establish an alternate route around the failure. This period is sometimes described as the *50 millisecond* behavior of the network because FRR restoration ideally becomes effective within approximately 50 milliseconds of a failure.

Routing of LSPs and demands differ in FRR simulations compared to full convergence simulations, as follows:

- The source node of the protected LSP does not reroute the traffic. If a failure occurs on a node or circuit (link) within the LSP path, the following routing occurs:
 - If an FRR LSP exists to protect a failed node or circuit, the protected LSP continues to be routed, but its new path includes the FRR LSP route that redirects the traffic around the failure. This traffic enters the FRR LSP's source node just prior to the failure and rejoins the protected LSP path after the failure at the FRR LSP's destination node.
 - If the LSP is not protected by an FRR LSP, then the LSP traffic is not routed.
- Because there is no IGP reconvergence, demands through unrouted LSPs are unrouted. Additionally, demands through failures outside of these LSPs are not routed.

FRR Fundamentals

This section summarizes key properties and terms, and how FRR LSPs are visualized in the network plot. More details about these are further explained throughout the following sections. Note that FRR LSPs appear in the same table as do regular LSPs and are identified in the Metric Type column as FRR.

Objects and Properties

Object	Type	Property	Description
LSP	FRR LSP	FRR Interface	Designated interface to be avoided by the FRR LSP.
		FRR Type	Type of FRR protection: Link or Node.
	Autoroute Forwarding Adjacency	FRR Enabled	Marks an LSP to be protected by FRR LSPs.
Interface	na	FRR Enabled	Marks an interface to be protected by FRR LSPs. This property is required if an interface is to be used when running the FRR LSPs initializer.

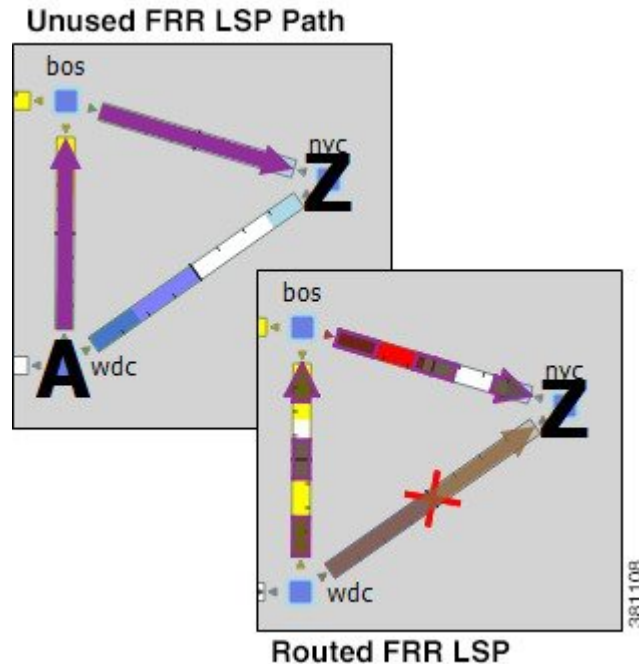
Terms

- Protected interface—An interface that is avoided when using FRR LSPs.
- Protected LSP—An LSP that would be rerouted through FRR LSPs in the event of failures. In WAE Design, this requires the Fast Reroute simulation mode. To turn this mode on, access the Network Options dialog box. For more information, see [Running Fast Reroute Simulation, on page 158](#).

- Protected SRLG—An SRLG that contains a protected interface.

Visualization

An unused FRR LSP path highlights in magenta when it is selected from the LSPs table. When routed, FRR LSPs appear as a dashed magenta line.

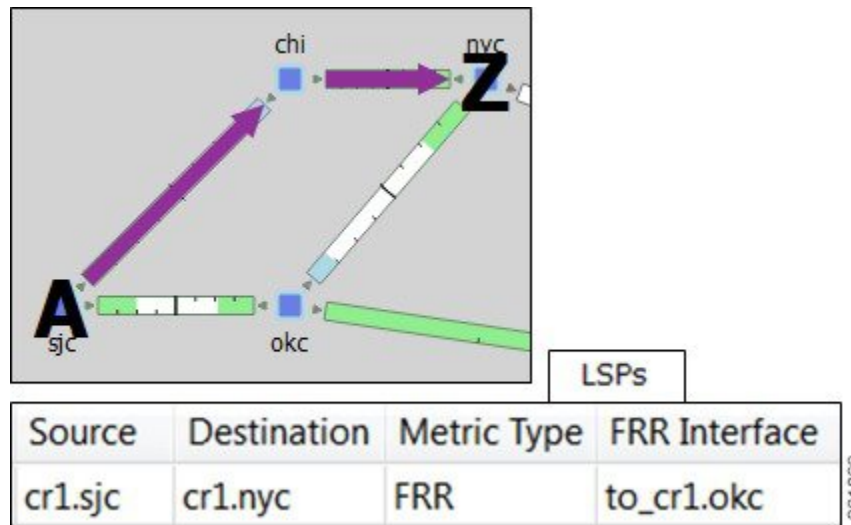


FRR LSP Routing

The source and destination of an FRR LSP can be any pair of nodes, although in practice the source and destination nodes are usually one hop (link protection) or two hops (node protection) apart. The source node for the FRR LSP has a designated interface that is configured so that it is avoided during Fast Reroute simulations. (This is the *protected interface*.)

This is configured using the FRR Interface property, which is set in the LSP Properties dialog box when manually creating the FRR LSP, or it is automatically created when running the FRR LSPs initializer. The initializer derives it from interface FRR Enabled property.

Example: This figure shows an FRR LSP sourced at sjc, destined for nyc, and whose protected interface (FRR Interface) is to_cr1.okc. Therefore, the FRR LSP path routes towards chi and away from okc. If this FRR Interface property were to_cr1.chi instead, the FRR LSP path would route sjc-okc-nyc, away from chi.



Like other LSPs, FRR LSPs can use actual and named paths. Unlike other LSPs, they do not use setup bandwidth when routing and they only use one LSP path.

Link and Node Protection

WAE Design supports both link (circuit) and node protection FRR LSPs. The type of protection shows in the FRR Type column of the LSPs table.

Metric Type	FRR Type
FRR	Link
FRR	Node

- Link-protection LSPs—Protect an LSP from a circuit failure by routing an FRR LSP around the failed circuit. The source node of the FRR LSP is the local node of the failed circuit, and if configured correctly, the destination of the FRR LSP is the remote node of the failed circuit. [Figure 48: Example Full Convergence Versus Fast Reroute with Link Protection, on page 152](#) shows an example of an LSP from sjc to nyc with a failure on the circuit between slc and hst. The top portion shows the Full Convergence mode where the source node of the LSP is signaled to reroute the traffic. The lower portion shows that a link-protection FRR LSP was set up for the slc-hst interface, and that the simulation is operating in Fast Reroute mode. Here, slc reroutes the LSP traffic around the failed circuit.
- Node-protection LSPs—Protect an LSP from a node failure by routing an FRR LSP from the node one hop prior to the failed node in the protected LSP path to the node one hop after the failed node. [Figure 49: Example Full Convergence Versus Fast Reroute with Node Protection, on page 152](#) shows an example of an LSP from sjc to nyc with a failure on the hst node. The top portion shows the Full Convergence mode where the source node of the LSP is signaled to reroute the traffic. The lower portion shows that a node-protection FRR LSP was set up to protect the hst node, and that the simulation is operating in Fast Reroute mode. Here, the slc node reroutes the LSP traffic around the node failure.

Figure 48: Example Full Convergence Versus Fast Reroute with Link Protection

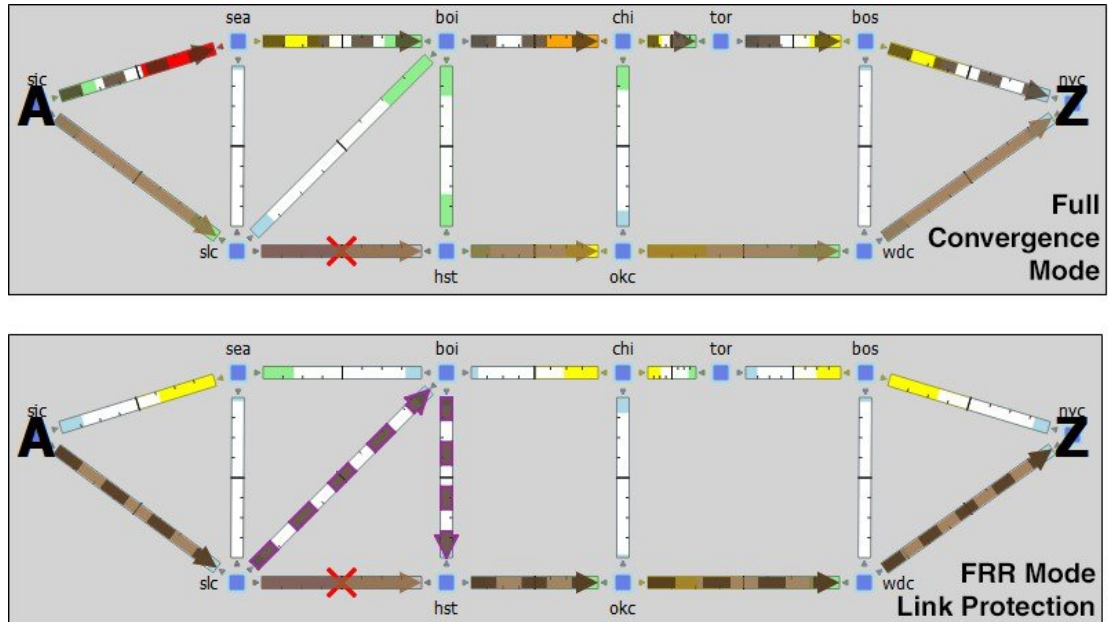
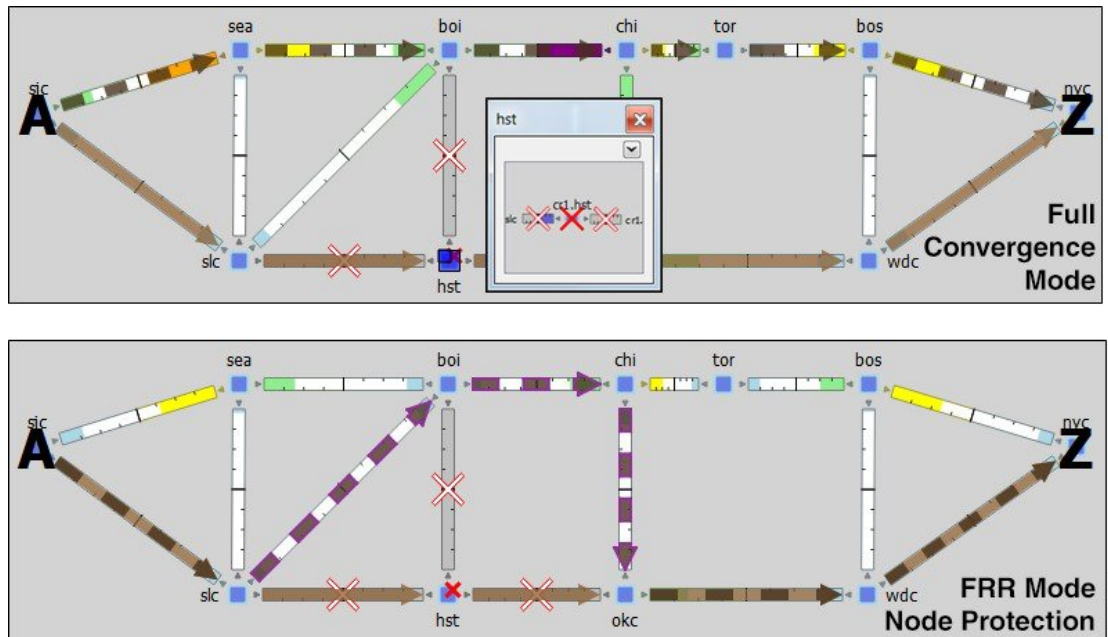


Figure 49: Example Full Convergence Versus Fast Reroute with Node Protection



SRLG Protection

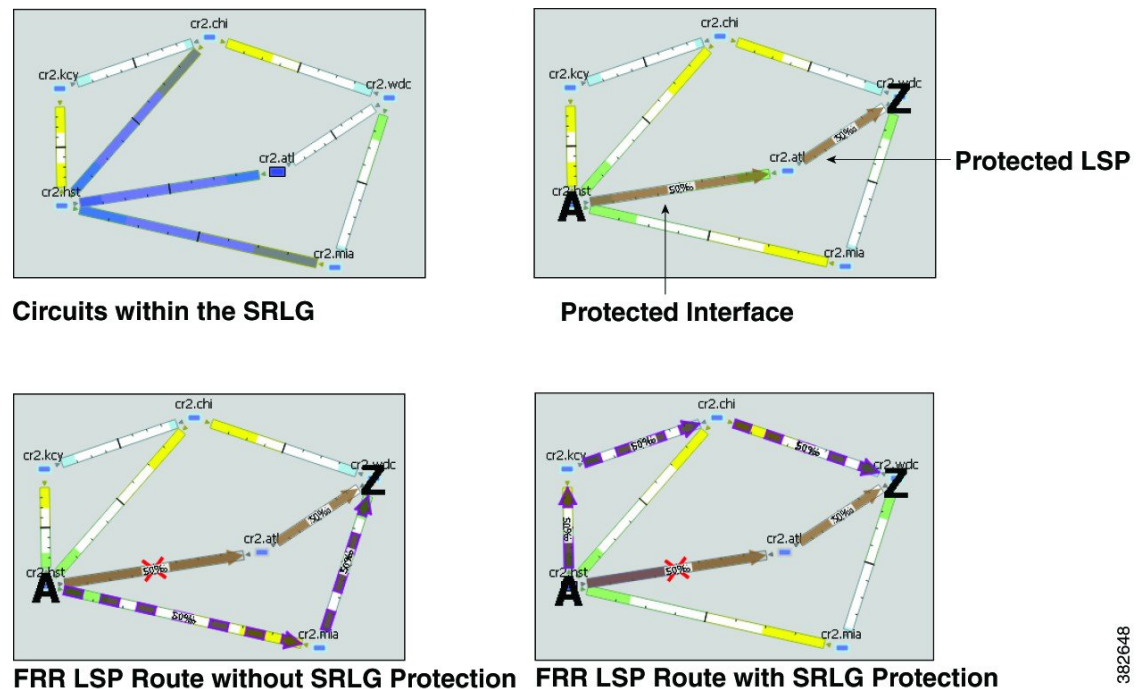
Figure 50: Example Fast Reroute without and with SRLG Protection, on page 153 In a network, if interfaces are sourced from the same node (router) and are configured as an SRLG, an FRR LSP protecting one of these interfaces can be configured to reroute around that interface. If possible, the FRR LSP avoids all other interfaces in the SRLG. In WAE Design, FRR LSPs can be configured to avoid all objects defined in the SRLG, not just interfaces on the source node of the FRR LSP.

Example: shows an example that uses the following setup:

- A protected LSP from cr2.hst to cr2.wdc.
- The protected interface is to_cr2.atl, and its source node is cr2.hst.
- An SRLG containing three circuits: cr2.hst_cr2.chi, cr2.hst_cr2.atl, and cr2.hst_cr2.mia.
- This SRLG has been configured on the cr2.hst node.

When the cr2.hst_cr2.atl circuit fails and there is no SRLG protection, circuits within the SRGL can be used to reroute the LSP. When there is SRLG protection in place, the LSP is rerouted to avoid all circuits within the SRLG.

Figure 50: Example Fast Reroute without and with SRLG Protection



382648

Routing Around Failures

Assuming one or more failures occurred, the following decisions determine which FRR LSP to take.

Prerequisites:

- LSPs must be configured to be protected by FRR LSPs. For information on setting up FRR LSPs, see [Setting Up Fast Reroute Simulations, on page 154](#).
- Any interfaces along the FRR LSP path must be configured to be protected by FRR LSPs. For information on protecting interfaces, see [Identifying LSPs for Protection, on page 154](#) or the [FRR LSPs Initializer, on page 155](#) (which automates the process of setting these).

Step 1

For each such interface on the LSP path, WAE Design checks for an FRR LSP protecting that interface whose destination is a node further down the path of the LSP.

- Step 2** If more than one of these FRR LSPs exist, WAE Design chooses the one that is farthest down that path of the protected LSP, thus simulating standard FRR behavior that prefers node protection over link protection.
- If more than one eligible FRR LSP has the same destination farthest down the protected LSP path, the FRR LSP is chosen arbitrarily among them.
- Step 3** If more failed circuits or nodes occur after this destination, further FRR LSPs are chosen in the same manner.

Setting Up Fast Reroute Simulations

- Step 1** Set appropriate properties on LSPs to mark them for protection. See [Identifying LSPs for Protection, on page 154](#).
- Step 2** If protecting SRLGs, associate a node with the SRLG in order to establish the protected interface within it. See [Identifying SRLGs for Protection, on page 154](#).
- Step 3** If you are using the initializer to create FRR LSPs, identify which interfaces to protect. See [FRR LSPs Initializer, on page 155](#).
- Step 4** Create the FRR LSPs using the FRR LSPs initializer or manually. See [FRR LSPs Initializer, on page 155](#) or [Manually Creating FRR LSPs, on page 157](#).

Identifying LSPs for Protection

To be marked for Fast Reroute protection, LSPs must have two properties set in the LSP Properties dialog box.

- Routing type must be Autoroute or Forwarding Adjacency. The type appears as “autoroute” or “FA” in the Metric Type column of the LSPs table.
- FRR Enabled must be checked. This appears as T (true) in the FRR Enabled column of the LSPs table. If this is F (false), the LSP is not protected.

Identifying SRLGs for Protection

To simulate a router’s “exclude protect” configuration, associate the source node of the protected circuit with the SRLG. If that circuit fails, the FRR LSP reroutes around all circuits within that SRLG.

Included	SRLG
<input checked="" type="checkbox"/> T	Midwest_SRLG
<input type="checkbox"/> F	South_SRLG
<input type="checkbox"/> F	North_SRLG

- Step 1** Right-click the source node of a circuit within an SRLG that you want to protect and choose **Properties**.

Step 2 Click the **SRLGs** tab.

Step 3 Choose one or more SRLGs with which you want to associate this node, and click **OK**.

FRR LSPs Initializer

The FRR LSPs initializer automatically creates link-protection or node-protection FRR LSPs provided two conditions are met:

- One or more LSPs have their FRR Enabled property set. (See [Identifying LSPs for Protection](#), on page 154.)
- One or more interfaces on the path of these LSPs have their FRR Enabled property set. This tells the FRR LSPs initializer how to create the FRR LSPs so that in the event of a failure, the interface would be avoided when the LSP is being rerouted.
 - This property is set in the MPLS tab of the interface Properties dialog box. The Interfaces table > FRR Enabled column shows T (true) if the interface is included or F (false) if it is not.
 - After FRR LSPs are created, the interface is listed in the LSPs table > FRR Interface column.

Interface Properties

Basic | **Advanced** | MPLS | User

Selected Interfaces (2/32)

Reservable BW:

Reservable BW (%):

TE Metric:

Affinities:

TE Enabled FRR Enabled

Interfaces

Interface	FRR Enabled
to_cr1.wdc	T
to_cr1.chi	T

FRR LSPs Initializer Uses this Interface
FRR Simulations Route Away from this Interface

38118

The initializer creates the FRR LSPs based on these FRR Enabled properties and based on whether you choose to create link or node protection.

- Link protection—WAE Design creates FRR LSPs for each pair of next-hop nodes (two connected nodes) in the protected LSP path where the egress interface of the first hop has its FRR Enabled property set ([Figure 51: Example Link-Protection FRR LSPs Automatically Created](#), on page 156).
- Node protection—WAE Design creates FRR LSPs between each set of next next-hop nodes (three connected nodes) in the protected LSP path where the egress interface by of the first hop has its FRR

Enabled property set (Figure 52: Example Node-Protection FRR LSPs Automatically Created, on page 157).

These new FRR LSPs are named `FRR_<source>_<destination>_<postfix>`, where postfix is optional and set in the initializer's dialog box.

Figure 51: Example Link-Protection FRR LSPs Automatically Created

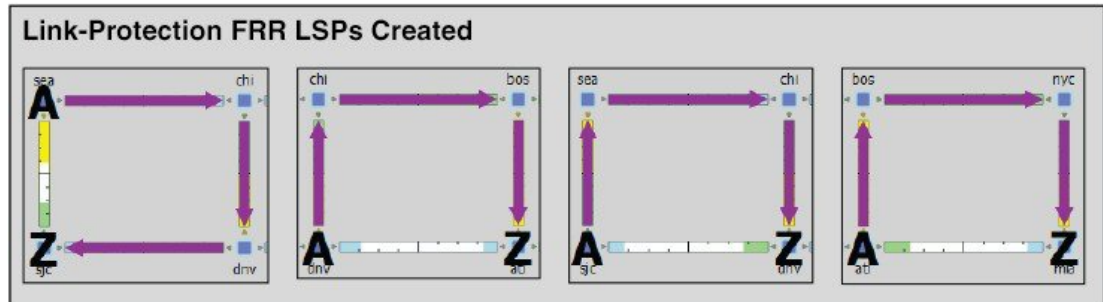
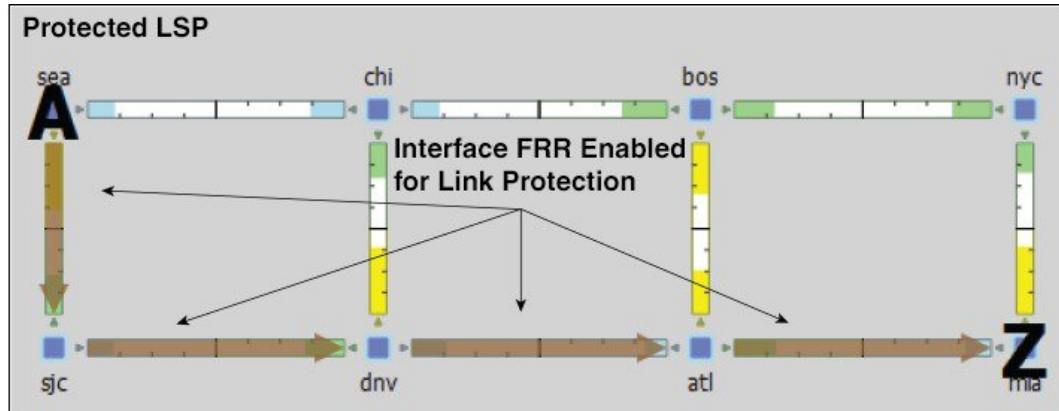
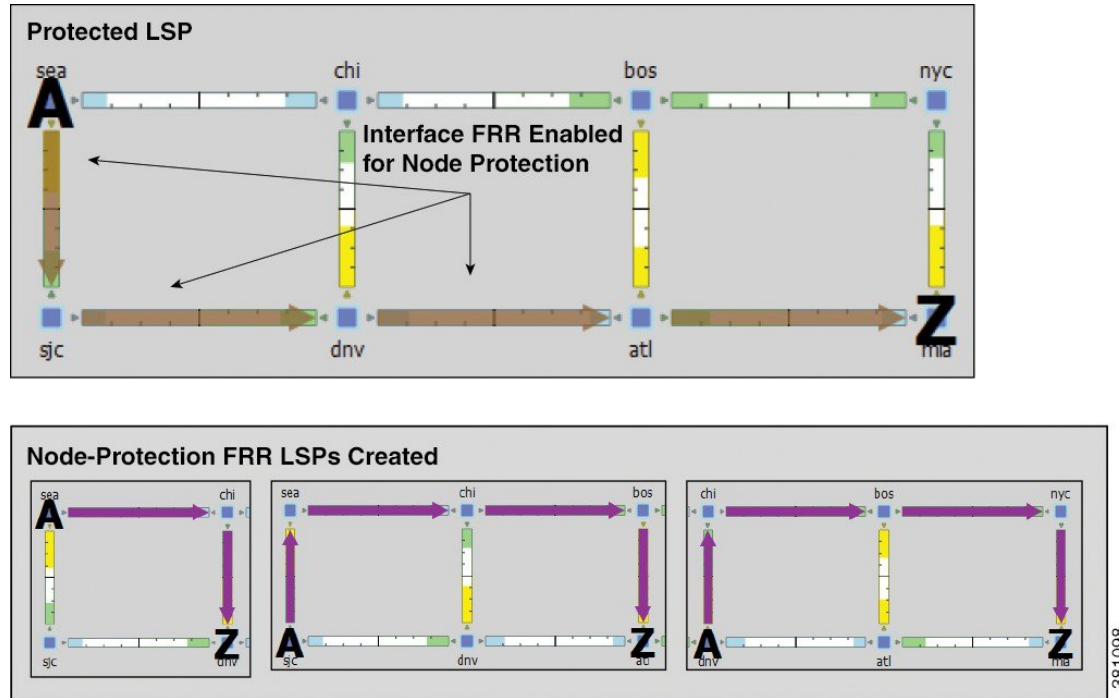


Figure 52: Example Node-Protection FRR LSPs Automatically Created



381098

Running the FRR LSPs Initializer

Step 1 Choose one or more nodes. If you do not choose any nodes, the initializer uses all nodes in the plan file.

Link Protection
 Node Protection
 Name (FRR_source_destination_postfix)
 Postfix:
 Delete current FRR LSPs

381110

Step 2 Choose **Initializers > FRR LSPs**.

Step 3 Choose whether to create FRR LSPs that protect links (circuits), nodes, or both.

Step 4 (Optional) Enter a postfix to add to the end of the names of the newly created FRR LSPs.

Step 5 Click **OK**.

Manually Creating FRR LSPs

This section describes the required properties for an FRR LSP. It does not describe all the options available when creating an LSP.

- Step 1** Choose **Insert > LSPs > LSP**, or right-click in an empty area and choose **New > LSPs > LSP**. Alternatively, you can choose an existing LSP to modify to be an FRR LSP. If so, right-click the LSP and choose **Properties**.

Source
Site: chi
Node: cr1.chi

Destination
 Site: okc
Node: cr1.okc
NetInt Destination:

- Step 2** Choose the source site and node of the FRR LSP.

- Step 3** Choose the destination site and node, or enter a NetInt destination IP address. For information on NetInt destination IP addresses, see [Unresolved LSP Destinations and Hops](#), on page 176.

Routing
Fast Reroute
Metric value: Load Share:
Active Path: Not set
FRR Interface: to_cr1.okc
NetInt FRRInterface:

to_cr1.bos
to_cr1.okc
to_cr1.sea

Interface to Route Away from in FRR Routing

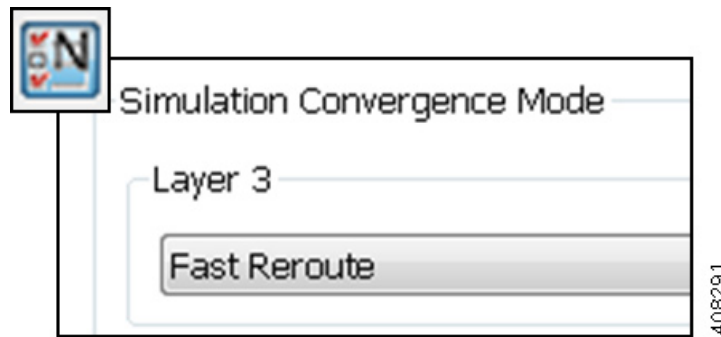
- Step 4** In the Routing area, choose **Fast Reroute** as the LSP type.

- Step 5** Specify the interface that is to be avoided during routing by choosing it from the FRR Interface drop-down list or by entering its IP address in the NetInt FRR Interface field.

- Step 6** Click **OK**.

Running Fast Reroute Simulation

FRR LSPs are routed only when simulating failures in the Fast Reroute simulation mode. To enable the Fast Reroute simulation mode, follow these steps.



-
- Step 1** Choose **Edit > Network Options** or click the **Network Options** icon in the Visualization toolbar.
- Step 2** Click the **Simulation** tab.
- Step 3** Choose **Layer 3 Fast Reroute** and click **OK**.
-

Autobandwidth Simulations

In a network, autobandwidth-enabled LSPs reset their setup bandwidth periodically based on the configured autobandwidth timers, and establish new routes if necessary. In networks with constrained bandwidth, changes in LSP routes can make other LSPs unroutable, which can then affect the amount of traffic in each LSP, the setup bandwidth settings, and the routes taken, without ever converging.

While WAE Design does not simulate this instability, it does simulate autobandwidth-enabled LSPs. In this simulation mode, internally WAE Design first routes the autobandwidth-enabled LSPs with a setup bandwidth of zero. Then, demands are routed to determine the simulated traffic (Traff Sim) through each LSP. The result in the plan file and GUI is that the Traff Sim value is copied to the Setup BW Sim for each autobandwidth-enabled LSP, and these LSPs are then routed using their new Setup BW Sim value. Note that if some LSPs cannot be routed, the Traff Sim and Setup BW Sim values resulting from this process might not match one another for all autobandwidth-enabled LSPs.

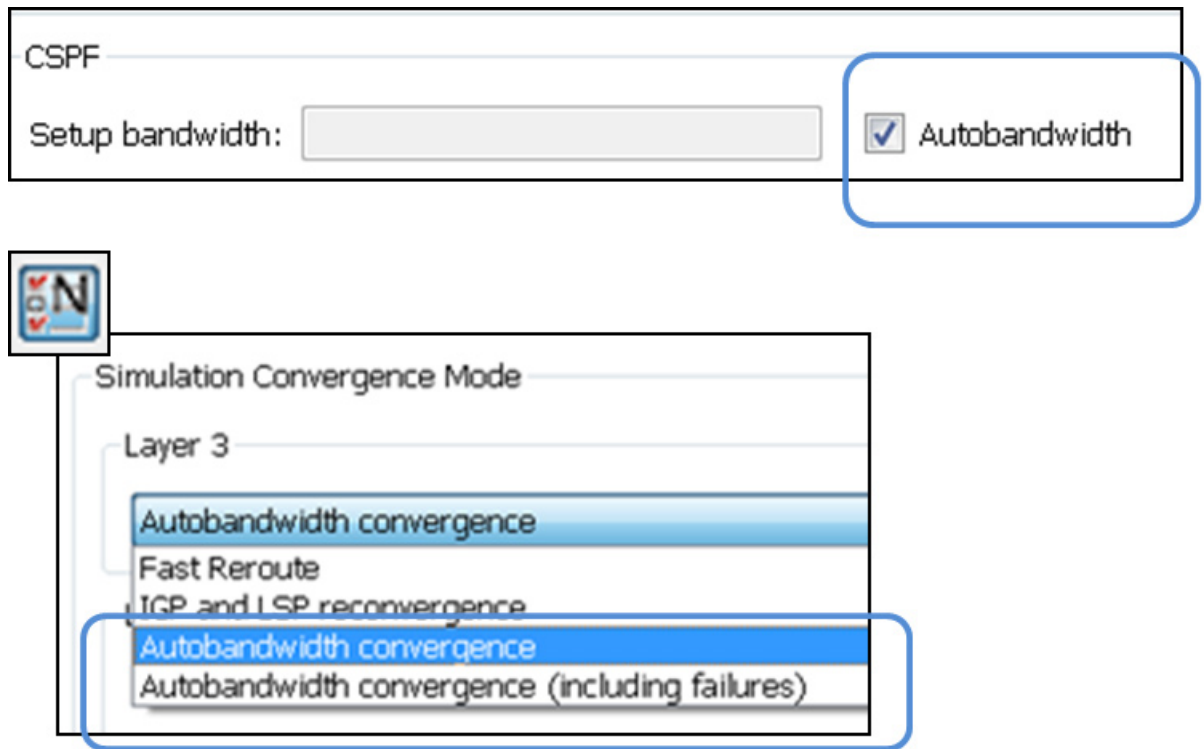
There are two autobandwidth simulation modes available. Before simulating failures or running Simulation Analysis on autobandwidth-enabled LSPs, choose the appropriate mode depending on which state you need to simulate.

- In the Autobandwidth Convergence mode, the network is simulated after the traffic (Traff Sim) has been rerouted to other LSPs, but before the Setup BW Sim values have been reset. This simulates the immediate, less optimal response to failures.
- In the Autobandwidth Convergence Including Failures mode, the network is simulated after the Setup BW Sim values have been reset. This simulates the longer, more optimal response to failures.



Note The Autobandwidth Convergence and Autobandwidth Convergence Including Failures simulation modes are only available in plan files with a single traffic level.

These autobandwidth simulations require that the LSPs have their Autobandwidth property set to T (True), and that one of the two autobandwidth modes be selected from the Network Options dialog box.

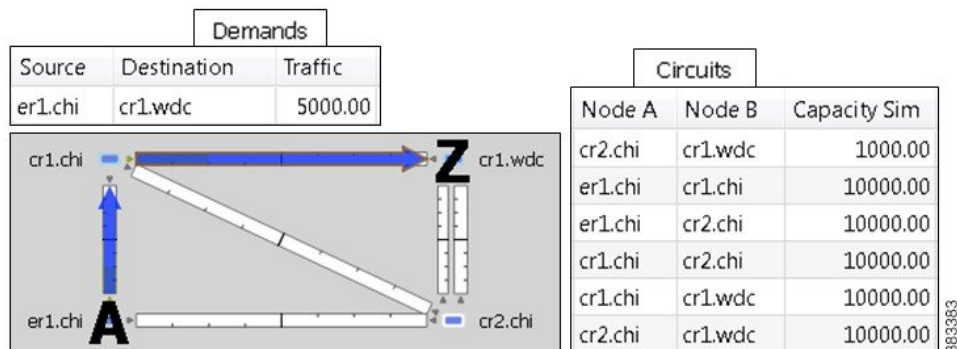


408290

Example Autobandwidth Convergence Without and with Failures

The network in this example has the following parameters:

- One demand from er1.chi to cr1.wdc.
- Two LSPs: LSP-A from cr1.chi to cr1.wdc and LSP-B from cr2.chi to cr1.wdc.
- Neither of these LSPs has a Setup BW.
- All circuits have a 10,000 Mbps capacity except for one (between cr2.chi and cr1.wdc), which is only 1000 Mbps.



- [Figure 53: Example LSP Routing Using IGP and LSP Reconvergence and Using Autobandwidth Convergence, on page 161](#) shows that in the IGP and LSP Reconvergence mode, the Setup BW Sim value for both LSPs is 0.

- In the Autobandwidth Convergence mode, the Setup BW Sim value for LSP-A is copied from its Traff Sim value, which is 5000.
- There are no failures, and each LSP follows the same route in both convergence modes.
- [Figure 54: Example Autobandwidth Convergence with Failures, on page 162](#) shows the Autobandwidth Convergence mode when there is a failure.
 - Failing the circuit between cr1.chi and cr1.wdc causes LSP-A to reroute through cr2.chi.
 - The shortest path from er1.chi through cr1.chi is now longer than the shortest path from er1.chi through cr2.chi, and so the demand moves over to LSP-B. This is shown in the LSPs table where the traffic (Traff Sim) of LSP-A is moved to LSP-B.
 - Because the smaller of the two circuits is congested, LSP-A uses the larger circuit, which has enough capacity to carry the traffic.
 - LSP-B continues to route on the lower capacity circuit, thus causing congestion.
 - The Setup BW Sim value is not updated for either LSP because the Autobandwidth Convergence mode does not update setup bandwidth as a result of traffic shifts due to failure.
- [Figure 55: Example Autobandwidth Convergence Including Failures, on page 162](#) shows the same failure in Autobandwidth Convergence Including Failures mode.
 - The LSP-B Setup BW Sim is updated to the Traff Sim value, forcing it to take the larger circuit to reach its destination. Note that the plot shows both the normal (solid) and failure (dashed) route for LSP-B, indicating that it has rerouted under the failure, even though the failure did not interrupt its path.
 - The demand continues to route on LSP-B, but because the LSP has found a path with sufficient capacity, there is no more congestion.

Figure 53: Example LSP Routing Using IGP and LSP Reconvergence and Using Autobandwidth Convergence

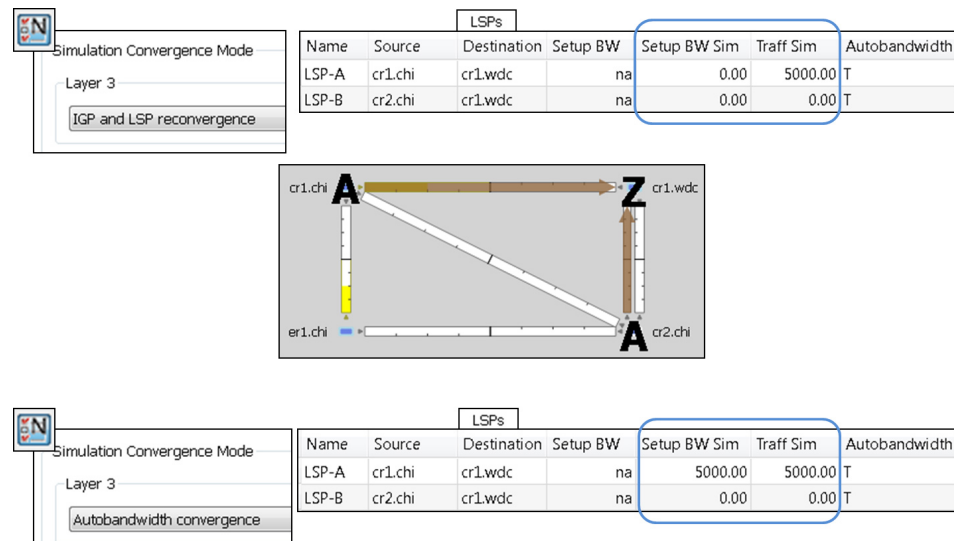


Figure 54: Example Autobandwidth Convergence with Failures

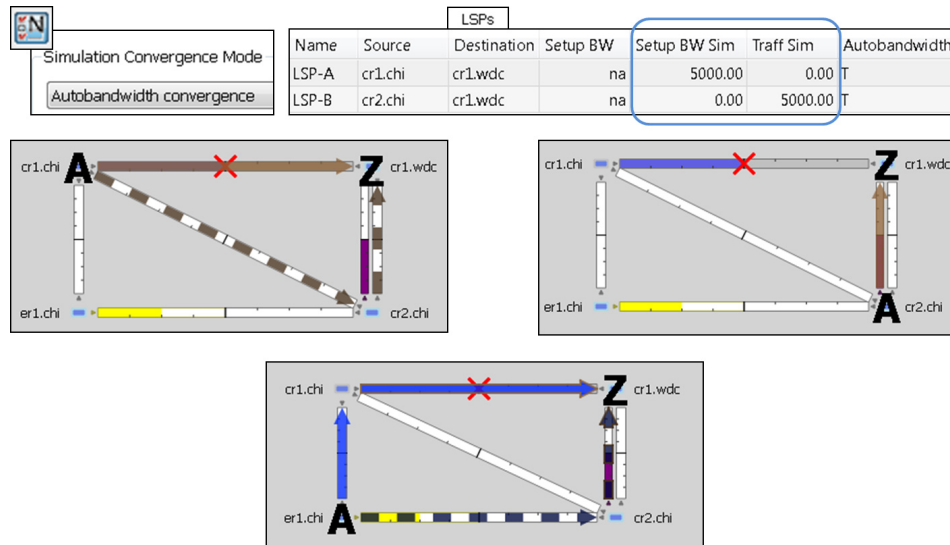
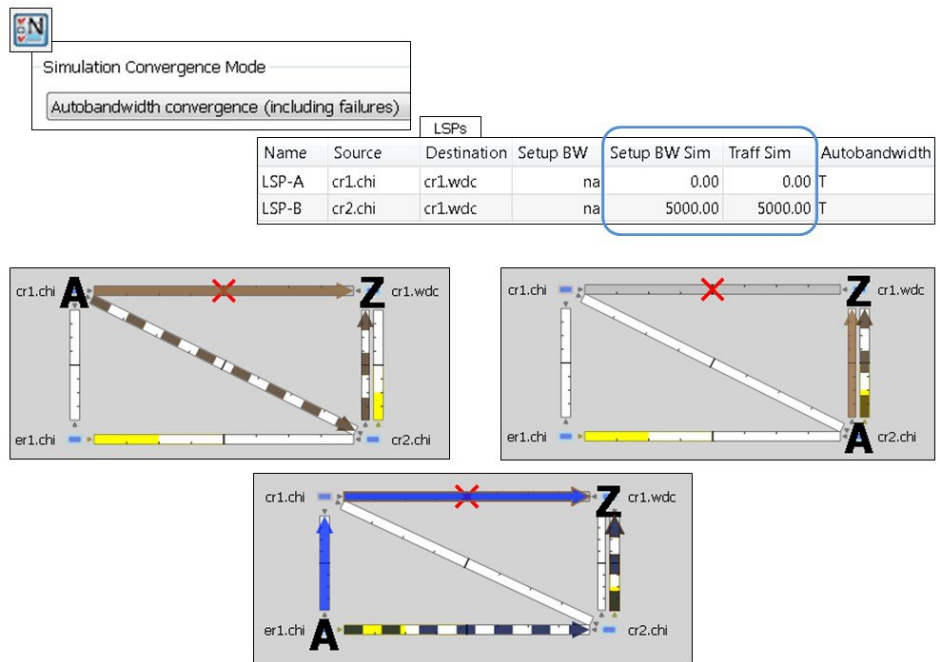


Figure 55: Example Autobandwidth Convergence Including Failures



Advanced RSVP-TE LSP Simulations

Ignoring Reservable Bandwidths for Capacity Planning

When using WAE Design for capacity planning, it is common to increase the demand traffic to a level expected at a future date. Various simulations are then performed, such as failures and worst-case analysis, to determine which circuits will likely experience over-utilization.

In MPLS networks using RSVP-TE LSPs, a setup bandwidth is configured for each LSP. If the demand traffic is increased, LSP setup bandwidths are usually increased correspondingly (for example, by using the LSP Setup Bandwidth initializer). When these LSP setup bandwidths become too large, some LSPs might not be routable, thus rendering an improbable view of the future network. When this occurs, you have the option to temporarily remove the reservable bandwidth constraints for a more realistic view into future capacity requirements. While this is not a routing method that actually exists on routers, it is a convenient WAE Design simulation that can assist you when planning networks.

When you enable this routing mode, WAE Design routes LSPs in the order they appear in the plan file LSPs table, as described in the following steps. Note that static order of the LSPs table in the plan file likely differs from the order of LSPs in the GUI, which changes frequently due to user interaction. For information on tables in plans files, see the *Cisco WAE Design Integration and Development Guide*.

This operates only on LSPs that do not have autobandwidth enabled.

Step 1 Route LSPs as usual. (For information, see [Dynamic LSP Routing and CSPF, on page 135.](#))

Step 2 If an LSP is not routable (call it “LSP-A”), try to route it with zero setup bandwidth.

- If LSP-A cannot be routed, leave it unrouted and try to route the next LSP in the LSPs table. If this LSP cannot be routed, try to route the next LSP listed in the LSPs table in the plan file.
- If LSP-A can be routed, then there will be one or more interfaces on its route whose reservable bandwidth is exceeded. These interfaces are marked as having unlimited reservable bandwidth during the simulation of all subsequent LSPs, thus allowing for the routing of LSP-A and other LSPs that would otherwise exceed the reservable bandwidth on these interfaces. Note that actual Resv BW and Resv BW Sim values in the Interfaces table do not change.

On completion of the simulation, a number of interfaces will have their reserved bandwidth greater than their configured reservable bandwidth. The circuits containing these interfaces are candidates for capacity expansion.

Example

In this example, there are four LSPs, each with a setup bandwidth (Setup BW) of 600 Mbps. All interfaces have a reservable bandwidth (Resv BW) of 1000 Mbps.

- LSPs
 - LSP-1, LSP-2, and LSP-3 go from LON to FRA.
 - LSP-4 goes from LON to ROM.
- IGP metrics

- Interfaces between PAR and FRA, and interfaces between FRA and ROM have an IGP metric of 2.
- All other interfaces have an IGP metric of 1.

Figure 56: Comparison of Reservable Bandwidth Being Used vs Not Being Used, on page 164 shows the LSP Reservations view of this simple network in both simulations modes: the usual where reservable bandwidths are observed and the capacity planning version wherein they are ignored. In this latter mode, the two purple circuits indicate they are being over utilized and likely need their capacity increased. (For information on the LSP Reservations view, see [Viewing LSP Reservations, on page 137.](#))

Figure 56: Comparison of Reservable Bandwidth Being Used vs Not Being Used

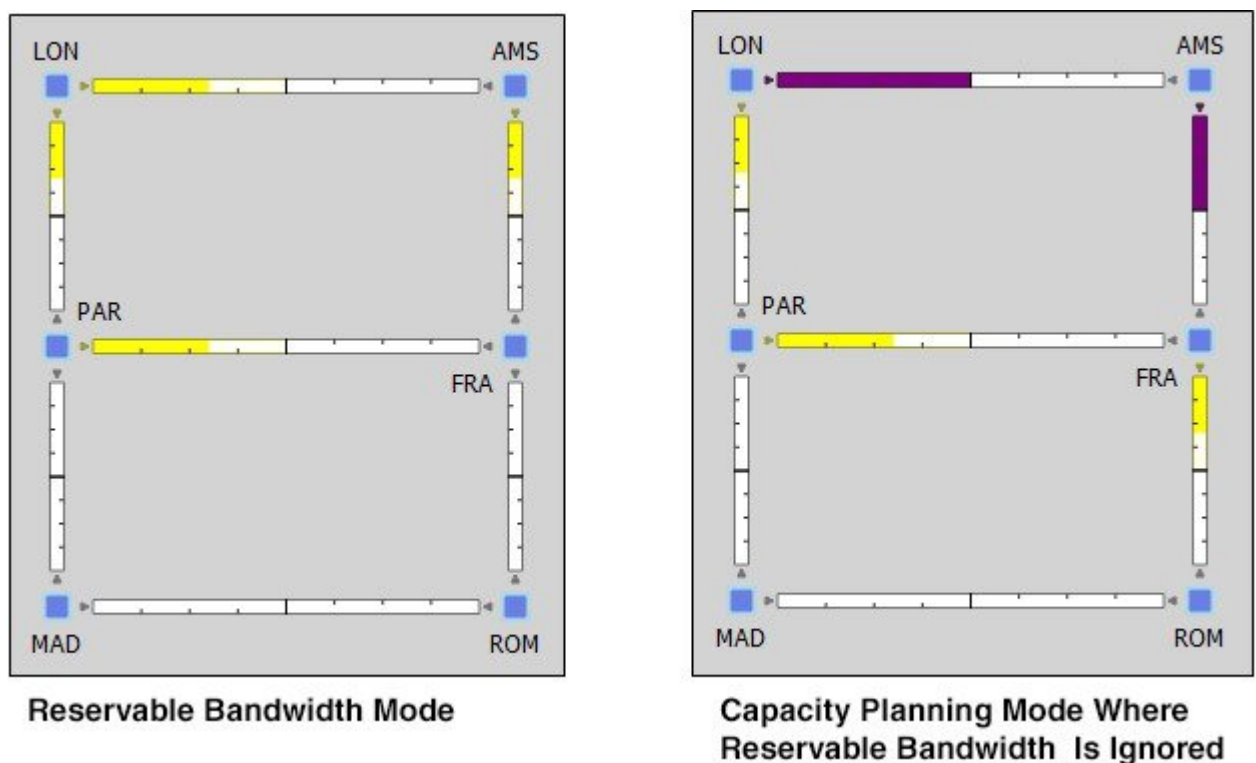
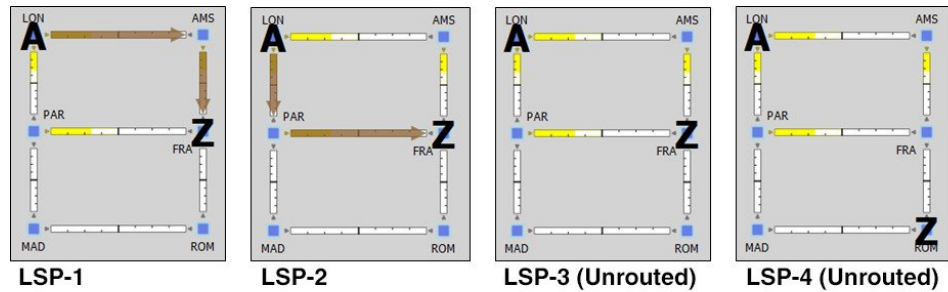


Figure 57: Simulated Routing Using Reservable Bandwidth Constraints, on page 165 demonstrates routing simulations where reservable bandwidth constraints are applied, which is the default behavior.

- LSP-1 routes across LON-AMS-FRA, leaving circuits LON-AMS and AMS-FRA each with only 400 Mbps of available bandwidth.
- LSP-2 routes across LON-PAR-FRA, leaving circuits LON-PAR and PAR-FRA each with only 400 Mbps of available bandwidth.
- LSP-3 cannot be routed because there is not enough available bandwidth on LON-AMS or LON-PAR. Its Setup BW Sim value in the LSPs table is set to “na” and the Routed column is set to “Unrouted.”
- LSP-4 cannot be routed, again because there is not enough available bandwidth. Its Setup BW Sim value in the LSPs table is set to “na” and the Routed column is set to “Unrouted.”

Figure 57: Simulated Routing Using Reservable Bandwidth Constraints



Interfaces				
Node	Interface	Resv BW	Resv BW Sim	Remote Node
cr1.lon	to_cr1.par	1000.00	1000.00	cr1.par
cr1.ams	to_cr1.fra	1000.00	1000.00	cr1.fra
cr1.par	to_cr1.fra	1000.00	1000.00	cr1.fra
cr1.lon	to_cr1.ams	1000.00	1000.00	cr1.ams

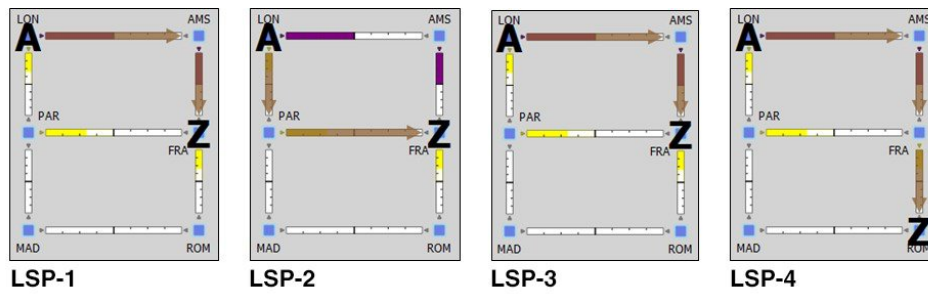
LSPs					
Name	Source	Destination	Setup BW	Setup BW Sim	Routed
LSP-1	cr1.lon	cr1.fra	600.00	600.00	Simulated
LSP-2	cr1.lon	cr1.fra	600.00	600.00	Simulated
LSP-3	cr1.lon	cr1.fra	600.00	na	Unrouted
LSP-4	cr1.lon	cr1.rom	600.00	na	Unrouted

381107

In [Figure 58: Simulated Routing Without Using Reservable Bandwidth Constraints, on page 166](#), the use of reservable bandwidth constraints has been disabled. The LSP routes are now as follows.

- As with the default behavior, LSP-1 routes across LON-AMS-FRA and LSP-2 routes across LON-PAR-FRA.
- Once it is determined that there is not enough reservable bandwidth for LSP-3 to route, it takes the shortest path, LON-AMS-FRA. The reservable bandwidth on interfaces LON-to-AMS and AMS-to-FRA is ignored for the remainder of this simulation. LSP-3's Setup BW Sim value in the LSPs table is set to 600.
- LSP-4 routes across LON-AMS-FRA-ROM because these reservable bandwidth constraints have been removed and because it is the only route available. Even though LON-PAR-MAD-ROM is the shortest path, there is not enough reservable bandwidth on LON-to-PAR or PAR-to-FRA. LSP-4's Setup BW Sim value in the LSPs table is set to 600.

Figure 58: Simulated Routing Without Using Reservable Bandwidth Constraints

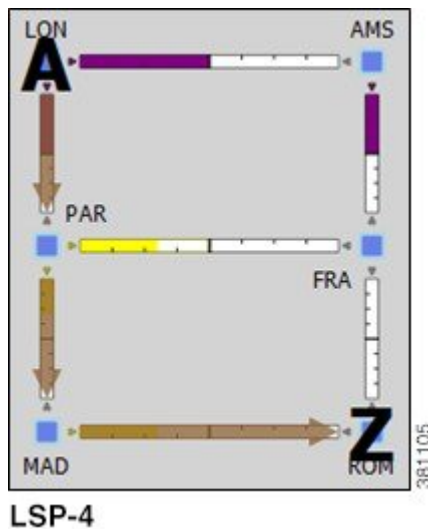


Interfaces				
Node	Interface	Resv BW	Resv BW Sim	Remote Node
cr1.lon	to_cr1.par	1000.00	1000.00	cr1.par
cr1.ams	to_cr1.fra	1000.00	1000.00	cr1.fra
cr1.par	to_cr1.fra	1000.00	1000.00	cr1.fra
cr1.lon	to_cr1.ams	1000.00	1000.00	cr1.ams

LSPs					
Name	Source	Destination	Setup BW	Setup BW Sim	Routed
LSP-1	cr1.lon	cr1.fra	600.00	600.00	Simulated
LSP-2	cr1.lon	cr1.fra	600.00	600.00	Simulated
LSP-3	cr1.lon	cr1.fra	600.00	600.00	Simulated
LSP-4	cr1.lon	cr1.rom	600.00	600.00	Simulated

381106

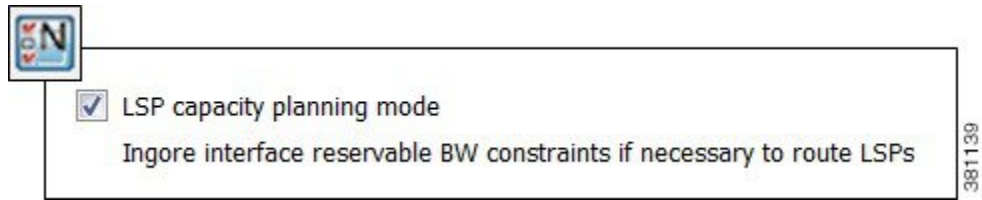
If, however, LSP-4 were routed before LSP-3, then LSP-4 would route across its shortest path of LON-PAR-MAD-ROM, and the LON-to-PAR interface would have its reservable bandwidth constraint ignored. With this routing, LSP-3 would still route LON-AMS-FRA, and the interfaces LON-to-AMS and AMS-to-FRA would still be set to remove their reservable bandwidth constraint.



381105

Enabling Capacity Planning Mode for LSPs

To ignore reservable bandwidth constraints in routing simulations, follow these steps.



- Step 1** Choose **Edit > Network Options**, or click the **Network Options** icon in the Visualization toolbar.
- Step 2** Click the **Simulation** tab.
- Step 3** Check **LSP capacity planning mode**, thus ignoring reservable bandwidth constraints, and click **OK**.

P2MP LSPs

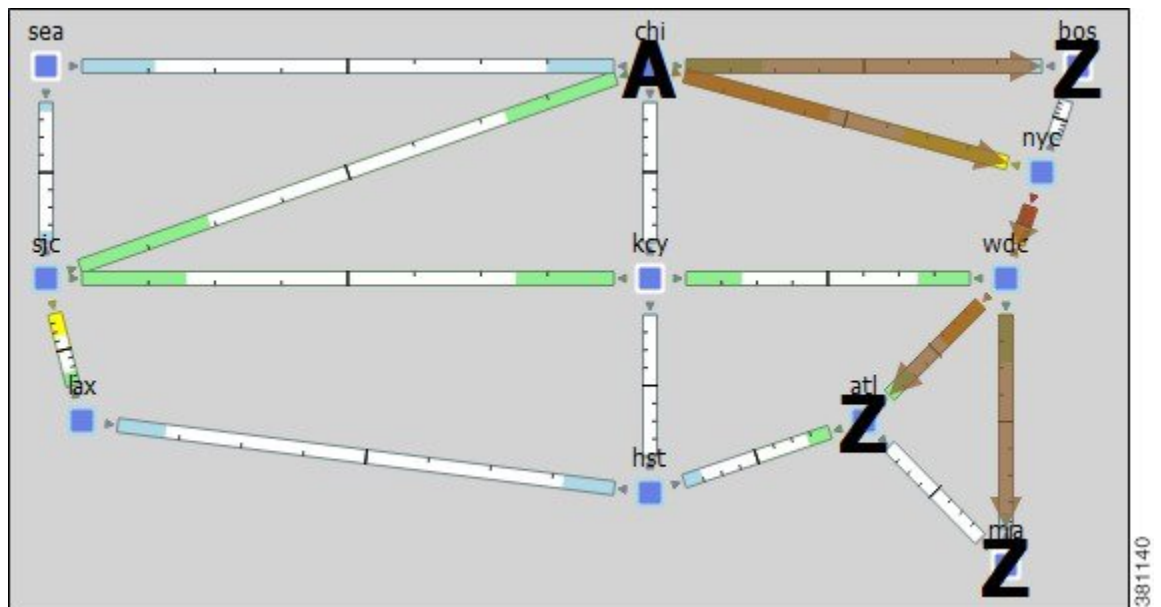
Point-to-multipoint (P2MP) LSPs offer an MPLS alternative to IP multicast for setting up multiple paths to multiple locations from a single source. In live networks, the signal is sent once across the network and packets are replicated only at the relevant or designated branching MPLS nodes.

A P2MP LSP consists of two or more sub-LSPs that have the same source node. Most of WAE Design refers to sub-LSPs as LSPs, and there is no distinction between the two.

To view the P2MP LSPs or their associated sub-LSPs, choose them from the P2MP LSPs and LSPs tables.

- To determine if an LSP belongs to a P2MP LSP, you can show the P2MP LSP column in the LSPs table. You can also right-click an LSP and choose **Filter to P2MP LSPs**.
- To determine which LSPs belong to a P2MP LSP, you can visually see them by choosing the P2MP LSP from the P2MP LSPs table. You can also right-click a P2MP LSP and choose **Filter to sub LSPs**.

Figure 59: P2MP LSP with chi as the Source for Three Destinations



Plotting P2MP LSPs

- Step 1** In the P2MP LSPs table, choose one or more P2MP LSPs.
- Step 2** Right-click and choose **Plot P2MP LSPs**.
- Step 3** To move the plot horizontally or vertically, use the left/right arrows and up/down arrows.
- Step 4** To choose all P2MP LSPs within the plot, click **Select P2MP LSPs**. To deselect all of these P2MP LSPs, click in an empty plot area.
- Step 5** To visualize multilayer L1 paths when plotting P2MP LSPs, use the following drop-down options:
- **L3**—Displays the path in terms of interfaces.
 - **L1**—Displays the path in terms of L1 links.
To better align the L1 links, WAE Design checks for L3-L1 links. If they do not exist, WAE Design checks for collocated L3/L1 nodes within sites.
 - **L3+L1**—Displays the L3 plot above and the L1 plot below.
WAE Design does not perform alignment between L3 and L1 nodes. When you click an L3 interface, the associated L1 links are highlighted automatically.
-

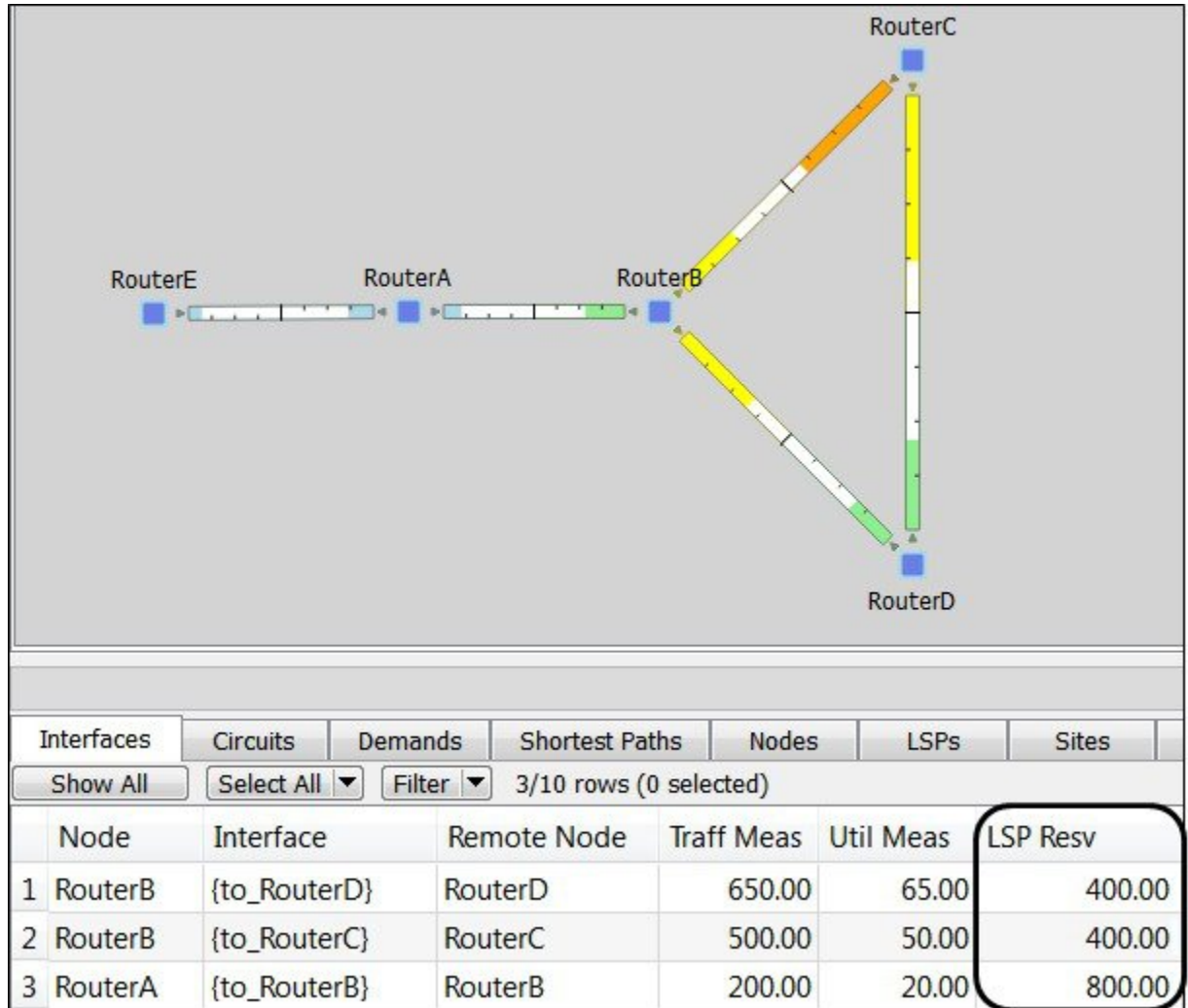
P2MP LSP Bandwidth

All sub-LSPs within a given P2MP LSP share bandwidth and therefore, common circuits between these sub-LSPs have the same bandwidth, rather than an aggregate. Common practice is to set all sub-LSPs within a P2MP LSP to the same bandwidth. Otherwise, the bandwidth for the shared circuit is the highest bandwidth set.

To see the reserved LSP bandwidth, show the LSP Resv column in the Interfaces table. You can also view the associated traffic by choosing **LSP Reservations** in the Network Plot menu in the Visualization toolbar. For more information, see [Viewing LSP Reservations, on page 137](#).

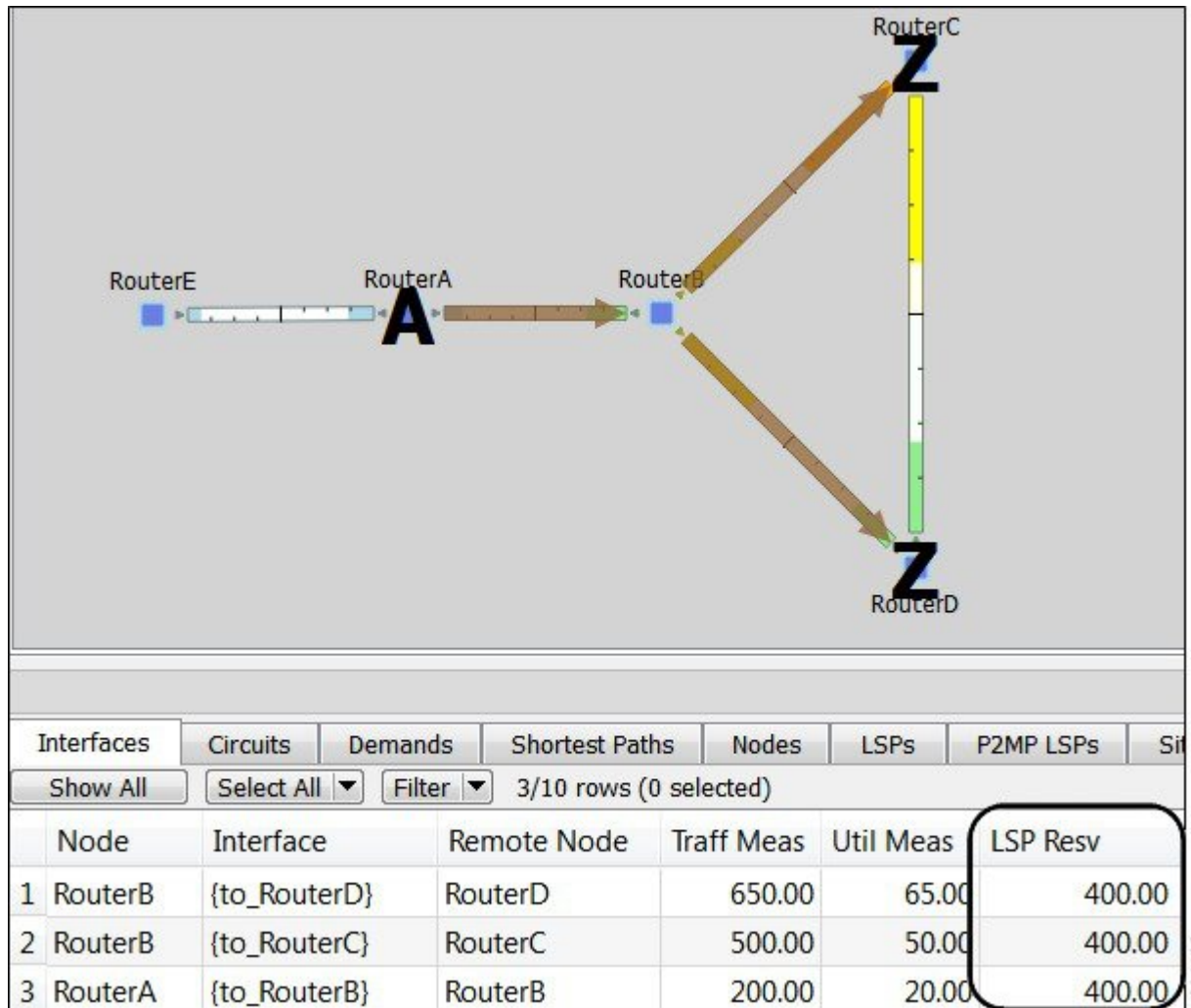
Example: If A-C LSP and A-D LSP each have a bandwidth 400 Mbps, the A-B circuit has an aggregate bandwidth of 800 Mbps ([Figure 60: Bandwidth Behavior Without a P2MP LSP, on page 169](#)). However, if A-C and A-D are sub-LSPs within a P2MP LSP, the A-B circuit has a bandwidth of 400 Mbps ([Figure 61: Bandwidth Behavior with a P2MP LSP, on page 170](#)).

Figure 60: Bandwidth Behavior Without a P2MP LSP



381122

Figure 61: Bandwidth Behavior with a P2MP LSP



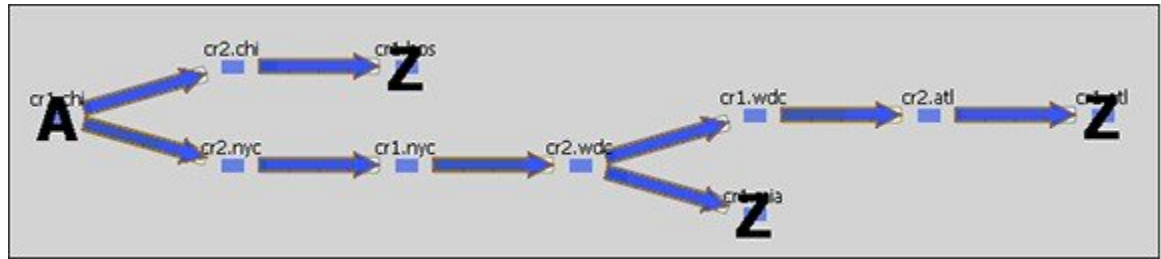
P2MP LSP Demands

Demands routed through P2MP LSPs must have the same source as the P2MP LSP; the demand then terminates at each of the P2MP LSP destinations. No other demands can go through the P2MP LSP except for the one created for it. If the P2MP LSP fails, its demand cannot be routed. (See [Creating Demands for P2MP LSPs, on page 175](#) for information on creating these demands.)

Demands must be private to the P2MP LSP. This privacy is automatically created when you insert a demand for a P2MP LSP.

For complex P2MP LSP demands, filter to the demand, right-click the demand, and choose **Plot Demand**. [Figure 62: Example Demand Plot for P2MP LSP, on page 171](#) shows an example of a demand plot for the P2MP LSP shown in [Figure 59: P2MP LSP with chi as the Source for Three Destinations, on page 167](#).

Figure 62: Example Demand Plot for P2MP LSP

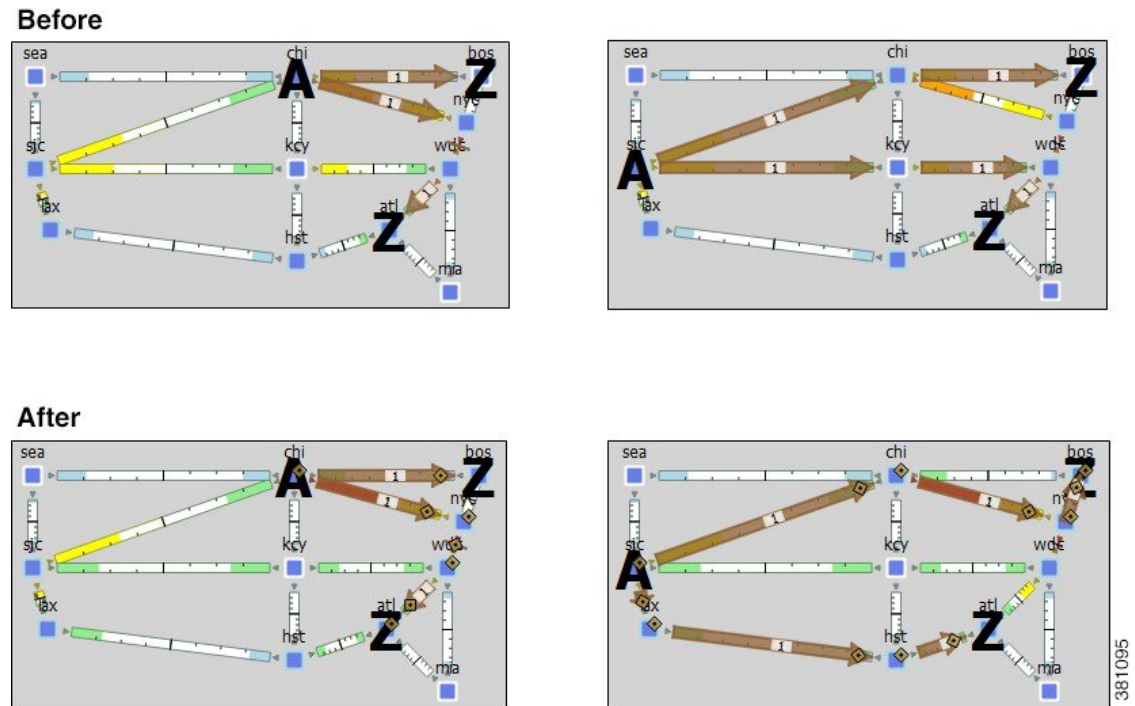


Explicit P2MP LSP Path Initializer

The Explicit P2MP LSP Paths Initializer automates the process of creating explicitly routed sub-LSP paths, as follows:

- Convert a set of dynamically routed sub-LSPs to explicitly routed sub-LSPs along their current paths.
 - Create disjoint paths. Two PSMP LSP paths are *disjoint* if they do not route over common objects. These objects are configurable and can be circuits, nodes, sites, SRLGs, or L1 links.
 - Create disjoint paths between sub-LSPs that have the same destination and whose P2MP LSPs belong to the same disjoint P2MP LSP group. This is useful for creating an alternate path if one of the sub-LSP paths becomes unavailable.
- Example: [Figure 63: Example Disjoint Paths for P2MP LSPs, on page 172](#) shows two P2MP LSPs (Region A and Region B) that are in the same disjoint group. Their sub-LSPs share two common destinations: bos and atl. Upon choosing this option, the sub-LSPs in Region B are rerouted to arrive at the common destinations using different explicit paths.
- Create disjoint primary and secondary paths for selected sub-LSPs.

Figure 63: Example Disjoint Paths for P2MP LSPs



Note Explicit P2MP LSP Paths Initializer supports Inter-Area functionality.

WAE Design creates named paths and named path hops for the sub-LSP paths. After running the initializer, you can verify the named paths and their hops in the Named Paths and Named Path Hops tables.

- If you choose a disjoint option that creates secondary paths, the secondary named path follows the shortest alternate sub-LSP path that is disjoint from the primary path, according to rules that you specify.
- Orphaned named paths are deleted.
- Explicit paths are named *<Sub-LSP Name> _<Path Option>*.

Prerequisites:

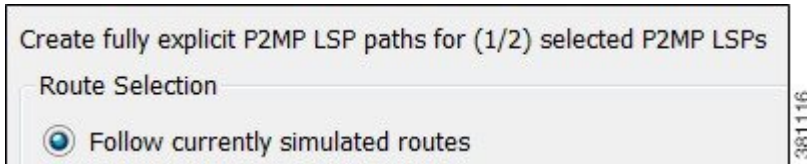
- The plan file must already contain the LSP paths.
- If creating disjoint paths between P2MP LSPs in the same disjoint group, the P2MP LSPs must first be added to the disjoint group in the P2MP LSP Properties dialog box. Here, you can also assign priorities to P2MP LSPs within these groups. Higher priority P2MP LSPs are assigned shorter routes based on the TE metric. The higher the number, the lower the priority.

Step 1 From the P2MP LSPs table, choose the P2MP LSPs to initialize with explicit named paths. If you do not choose any P2MP LSPs, all sub-LSP paths are used for creating the named paths.

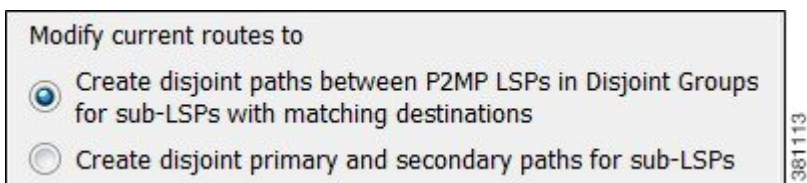
Step 2 Choose **Initializers > Explicit P2MP LSP Paths**, or right-click a P2MP LSP and choose **Explicit Path Initializer**.

Step 3 Specify how to create the explicitly routed named paths, and then click **OK**.

- To create named paths for all currently simulated primary or standby sub-LSP paths, click **Follow currently simulated routes**.

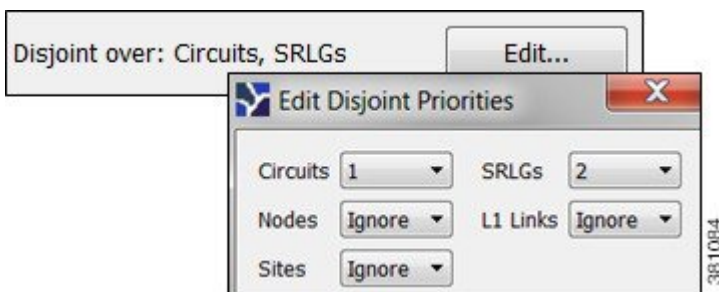


- To create disjoint paths, choose whether to create them between disjoint PSMP LSPs that have sub-LSPs with the same destination, or between disjoint primary and secondary sub-LSP paths.



- To modify the disjoint priorities that determine which objects are considered in paths, click **Edit**. Choose **1**, **2**, **3**, or **Ignore**, where the lower the number, the higher the priority. Click **OK**.

Example: If you give circuits a priority of 1, SRLGs a priority of 2, and ignore the other objects, when the disjoint paths are created, WAE Design first creates paths with disjoint circuits. Then, if there are common SRLGs in those paths, it creates paths so the SRLGs are disjoint. The remaining objects are not considered in the disjoint path creating.



Creating P2MP LSPs and Sub-LSPs

Rules and guidelines:

- All sub-LSPs within a given P2MP LSP must have the same source site and source node.
- Because sub-LSPs within a P2MP LSP share setup bandwidth, typically bandwidth is set to the same value for each sub-LSP.

Creating P2MP LSPs

- Step 1** Create an empty P2MP LSP.
- Choose **Insert > LSPs > P2MP LSP**, or right-click in the plot and choose **New > LSPs > P2MP LSP**.
 - Enter a unique P2MP LSP name. If you do not enter a name, WAE Design creates one using the format `p2mp_lsp_<source_node>`.
 - Choose the source site and node. This selection is the root for all sub-LSPs contained in this P2MP LSP.
 - (Optional) Enter or choose a disjoint group and disjoint priority for WAE Design to consider when using the Explicit P2MP LSP Path Initializer.

The screenshot shows a configuration dialog for a P2MP LSP. The 'Name' field contains 'East_Region'. Under the 'Source' section, the 'Site' dropdown is set to 'atl' and the 'Node' dropdown is set to 'cr1.atl'. Below this, the 'Disjoint Group' dropdown is set to 'Voice', and the 'Disjoint Priority' field is empty. A vertical ID number '381141' is visible on the right side of the dialog box.

- Step 2** If you have not done so, create sub-LSPs for the P2MP LSP.
- Step 3** Associate the sub-LSPs to the P2MP LSP. You can follow these steps individually, or you can choose multiple sub-LSPs and edit them collectively.
- Double-click the sub-LSPs from the LSP table.
 - In the Sub-LSP of P2MP LSP list (at the bottom), choose the P2MP LSP to which you want these LSPs to belong. Then click **OK**.

Creating Sub-LSPs

- Step 1** Choose **Insert > LSPs > LSP**, or right-click in the plot and choose **New > LSPs > LSP**.
- Step 2** Choose the site and node that will be the source of the P2MP LSP. All sub-LSPs in a P2MP LSP must have the same source site and source node.
- Step 3** Choose the site and node that are the final destination for the sub-LSP.
- Step 4** Complete the remaining optional fields and tabs, as needed. Because sub-LSPs share bandwidth, typically the CSPF Setup Bandwidth field is set to the same value for each sub-LSP in the P2MP LSP.
- Step 5** If you have already created the P2MP LSP to which this sub-LSP will belong, choose that P2MP LSP from the Sub-LSP of P2MP LSP list (at the bottom). Then click **OK**.

Creating a Mesh of Sub-LSPs

- Step 1** For each node that will be the egress node of the P2MP LSP, create the same tag name.

- a) In the Nodes table, choose and then double-click all nodes that will be included in the P2MP LSP.
- b) Click the **Tags** tab, then click **New Tag**.
- c) Enter a name that collectively identifies the sub-LSPs you are creating, and then click **OK**.
- d) Click **OK** to save and exit.

- Step 2** In the Nodes table, choose the node you will be designating as the source for the P2MP LSP.
- Step 3** Choose **Insert > LSPs > LSP Mesh**, or right-click in the plot and choose **New > LSPs > LSP Mesh**.
- Step 4** Uncheck these check boxes:
- **Same Destinations as Sources**
 - **Also create LSPs from destination to source**
- Step 5** In the Source Nodes drop-down list, choose **Selected in table (###)**. All sub-LSPs in a P2MP LSP must have the same source node. A P2MP LSP cannot have multiple source nodes.
- Step 6** In the Destination Nodes drop-down list, choose the newly created tag name.
- Step 7** To set the bandwidth to be the same for all sub-LSPs in this mesh, enter that value in the Fixed Value ___ Mbps field.
- To use the autobandwidth feature so that these LSPs have their Setup BW Sim value automatically calculated and used in autobandwidth simulations, click **Autobandwidth**.
- Step 8** Complete the remaining optional fields as needed, and then click **OK**.

Creating Demands for P2MP LSPs

To simulate traffic routing through a P2MP LSP, create a demand for it.

- Step 1** If you want to create demands for P2MP LSPs, choose one or more P2MP LSPs from the P2MP LSP table.
- Step 2** Choose **Insert > LSPs > Demands for P2MP LSPs**, or right-click in the plot and choose **New > LSPs > Demands for P2MP LSPs**.
- a) Choose the P2MP LSPs through which you are routing traffic.
 - b) (Optional) Choose a service class for the newly created demand.
 - c) Choose the desired bandwidth: Minimum sub-LSP Setup BW or Zero. Then click **OK**.

Deleting P2MP LSPs and Sub-LSPs

Deleting a P2MP LSP also deletes all sub-LSPs within it. If you do not want all the sub-LSPs deleted, disassociate them first.

Disassociating Sub-LSPs from Their P2MP LSPs

- Step 1** In the LSPs table, choose one or more sub-LSPs.
- Step 2** Double-click to open the Properties dialog box.
- Step 3** In the Sub-LSP of P2MP LSP list, do one of the following:
- Choose the empty row to disassociate the sub-LSP and change it to an LSP.

- Choose a different P2MP LSP to associate the sub-LSP with a different P2MP LSP.

Step 4 Click **OK**.

Deleting Sub-LSPs or P2MPs

Delete Sub-LSPs but Keep P2MP LSP	Delete P2MP LSP and Associated Sub-LSPs
Right-click one or more sub-LSPs in the LSPs table. Click Delete . Click Yes to continue.	Right-click one or more P2MP LSPs in the P2MP LSP table. Click Delete . Click Yes to continue.

Unresolved LSP Destinations and Hops

RSVP-TE LSP configurations and state are read using network discovery from the source node of the LSP, through SNMP, the Parse Configs tool, or other methods. These configurations might reference nodes or interfaces that do not exist in the plan file, so they are referred to as being unresolved. For example, the LSP destination node, read from the configuration on the source, might not be in the plan file. There are several reasons for these differences:

- The plan file was modified to contain fewer nodes than are in the IGP.
- WAE cannot read all the nodes, or cannot obtain the IP addresses of all the nodes.
- The LSPs themselves are not configured correctly.

References that might not be resolved are as follows:

- Destination nodes of the LSPs.
- Hops in named paths configured on the source node.
- Hops in the actual paths read from the source node.

WAE tries to resolve as many of these references as possible. Tables and columns are updated as follows:

- If the LSP destinations are resolved, they are updated in the Destination column of the LSPs table. If they are not resolved, the column remains empty. The original IP address remains in the NetInt Destination column regardless of whether the destination is resolved or not.
- If the hops are resolved, they are updated in the Node and Interface columns in the Named Path Hops and Actual Path Hops tables. If they are not resolved, the columns remain empty. The original IP address remains in the NetIntHop column regardless of whether the destination is resolved.

WAE Design does not make further attempts to resolve these references unless you explicitly make the request. There are special cases in which this might be useful. For example, if additional nodes are added to a network from a second network discovery procedure and if LSPs in the original network have unresolved references to these nodes, it is useful to resolve the LSPs again.

Resolving Plan File to Objects

If plan files have configurations that are not matched or refer to objects outside the modeled network, there are several ways to resolve these differences:

- Resolve LSP destinations, hops in named paths, hops in actual paths, or segment list hops by choosing **Initializers > Resolve Plan**.
- Resolve named path hops in the named paths Properties dialog box.
- Resolve named path hops in the named path hops Properties dialog box.



CHAPTER 11

Segment Routing Simulation

By default, WAE Design creates and routes RSVP LSPs. However, you can create Segment Routing (SR) LSPs by changing the LSP Type property to SR. These SR LSPs use autoroute, forwarding adjacencies, and class-based forwarding, just like RSVP LSPs. Once the packet enters the tunnel, the different routing mechanisms determine the path. An RSVP LSP is first established along a path, with each node on the path accepting the reservation request and maintaining the reservation state. In contrast, SR LSPs rely on a segment list that is created at the LSP's source node (head-end). This segment list, which is stored in the packet (demand), directs the traffic over a series of configured node segments and interface segments. SR LSPs use IP routing (and thus, potentially ECMP) between segments.



Note WAE Design uses the terms *SR LSPs* that use *segment lists*, which contain *segment list hops*. In Cisco routing terminology, these terms are *SR tunnels* that use *segment ID (SID) lists*, which contain *SIDs*.

This section contains the following topics:

- [SR LSP Segment Types, on page 179](#)
- [SR LSP Paths, on page 184](#)
- [SR LSP Routing, on page 185](#)
- [Inter-AS SR LSP Routing, on page 185](#)
- [Creating SR LSPs and Their LSP Paths, on page 187](#)
- [Creating Segment Lists, on page 188](#)
- [SR-TE Protection, on page 189](#)

SR LSP Segment Types

The segment list of an SR LSP contains one or more segment list hops, which can be nodes, interfaces, SR LSPs, or anycast groups.

To view a segment list, right-click an SR LSP and choose **Filter to Segment Lists**. You can then right-click the segment list and choose **Filter to Segment List Hops**.

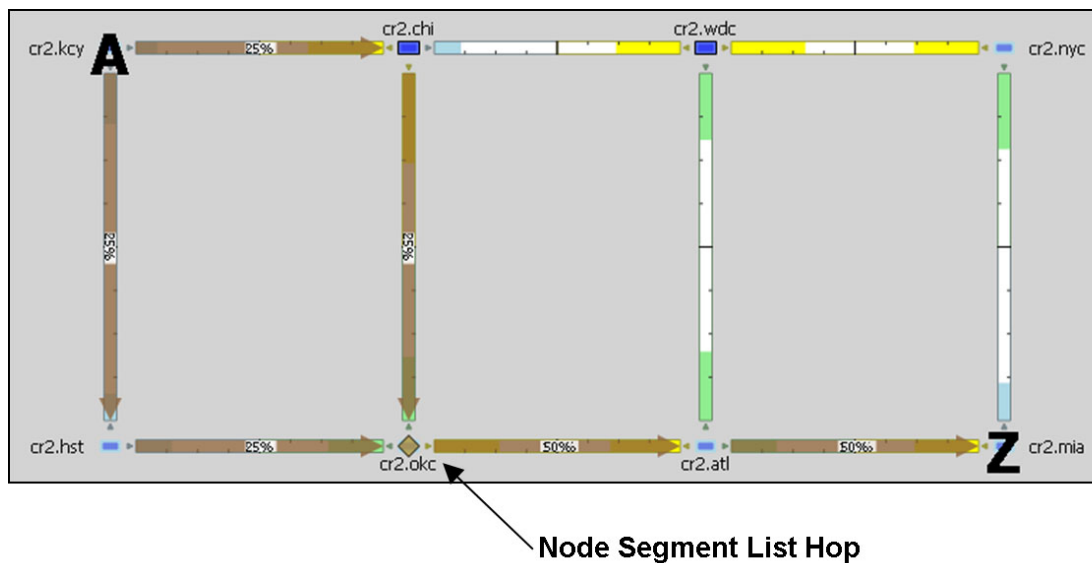


Note The process of creating and optimizing segment lists and their segment list hops can be automated using the SR-TE Optimization tool. For information, see [SR-TE Optimization, on page 341](#). You can also create SR LSPs and their hops by optimizing bandwidth. For information, see [SR-TE Bandwidth Optimization, on page 347](#).

Node Segment List Hops

When using node segment list hops, the SR LSP takes the shortest path to the specified node. [Figure 64: Example Node Segment List Hop, on page 180](#) shows an example SR LSP from cr2.kcy node to cr2.mia node using one node segment list hop, which is cr2.okc. Because the IGP metrics are equal, the traffic is routed using ECMP to reach okc.

Figure 64: Example Node Segment List Hop



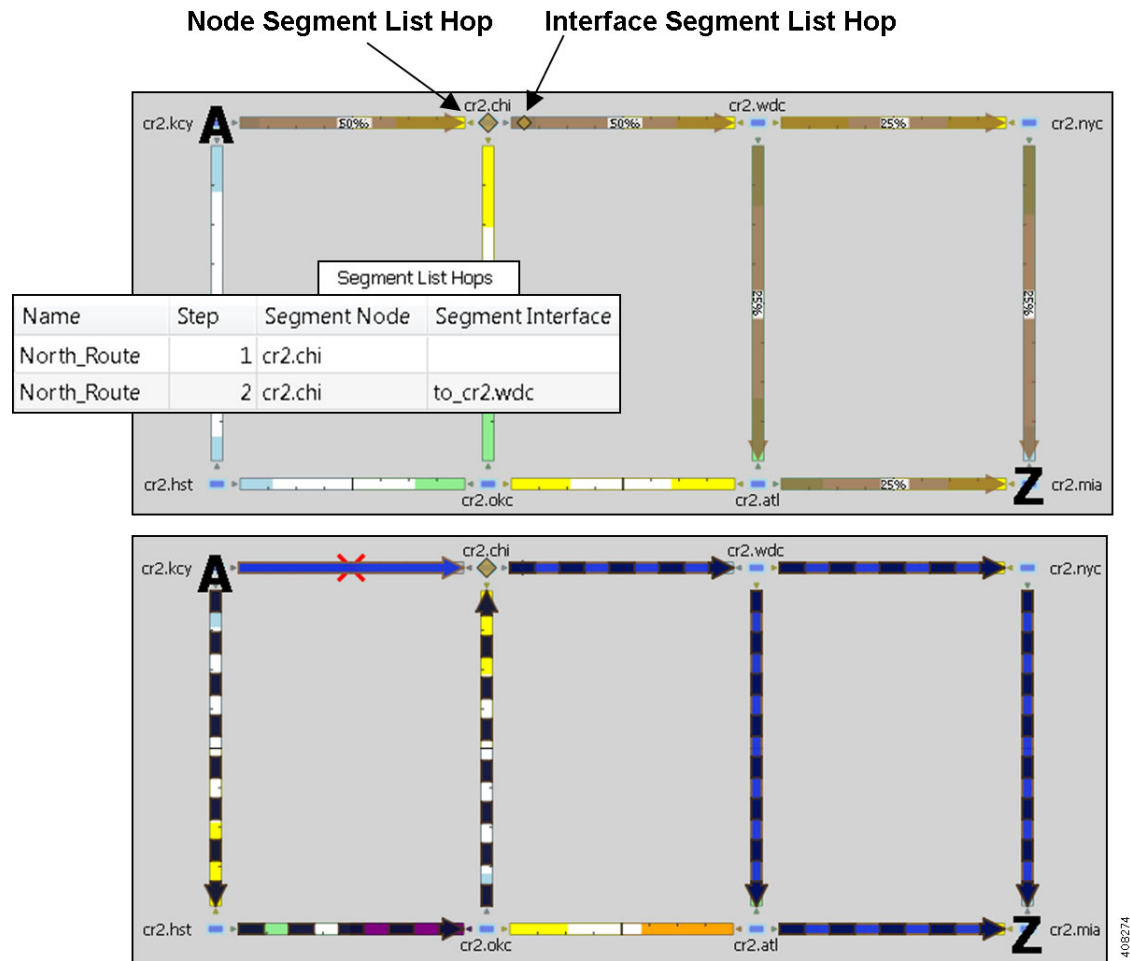
4/08/273

Interface Segment List Hops

On routers, interfaces can only be used in segment lists if the previous segment list hop is the node containing the interface, or if the interface is the egress of the source node. In WAE Design, there is no restriction that an interface segment list hop be local. There is no requirement that the segment list contain a preceding segment to the node containing the interface.

[Figure 65: Example Interface Segment List Hop, on page 181](#) shows an example of an interface segment list hop from cr2.chi to cr2.wdc. To ensure the path reaches cr2.chi (the node the packet must reach to use this interface segment list hop), a node segment list hop is configured first to reach chi. This figure also shows how a demand through an SR LSP reroutes when a node fails. The demand uses IGP to route around the failure and back to the segment, if possible.

Figure 65: Example Interface Segment List Hop



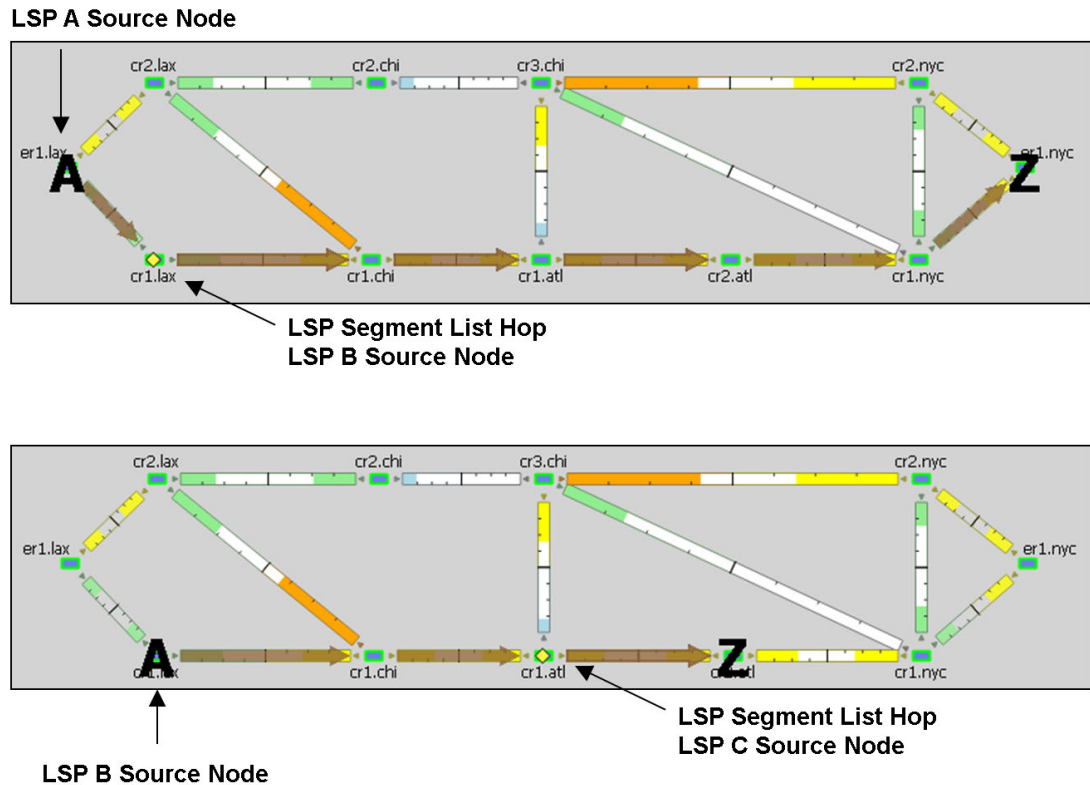
LSP Segment List Hops

SR LSPs can use other SR LSPs as segment list hops, and in turn, these SR LSPs segment list hops can contain other SR LSPs as segment list hops. Two rules apply:

- The LSP segment list hop must be an SR LSP, not RSVP LSP.
- An LSP segment list hop cannot reference another SR LSP that directly or indirectly references it. For example, if the segment for SR LSP A contains SR LSP B as a segment list hop, SR LSP B cannot contain SR LSP A as a segment list hop.

If a selected SR LSP contains an LSP segment list hop that contains yet another LSP segment list hop, only the top-level LSP segment list hop is identified by the yellow diamond icon. The destination of an LSP segment list hop is not identified. For example, [Figure 66: Example LSP Segment List Hop, on page 182](#) shows that LSP A (from er1.lax to er1.nyc) contains an LSP B segment list hop, whose source node is cr1.lax. By deselecting LSP A and then selecting LSP B, you see that LSP B also contains an LSP segment list hop, which is LSP C whose source node is cr1.atl.

Figure 66: Example LSP Segment List Hop



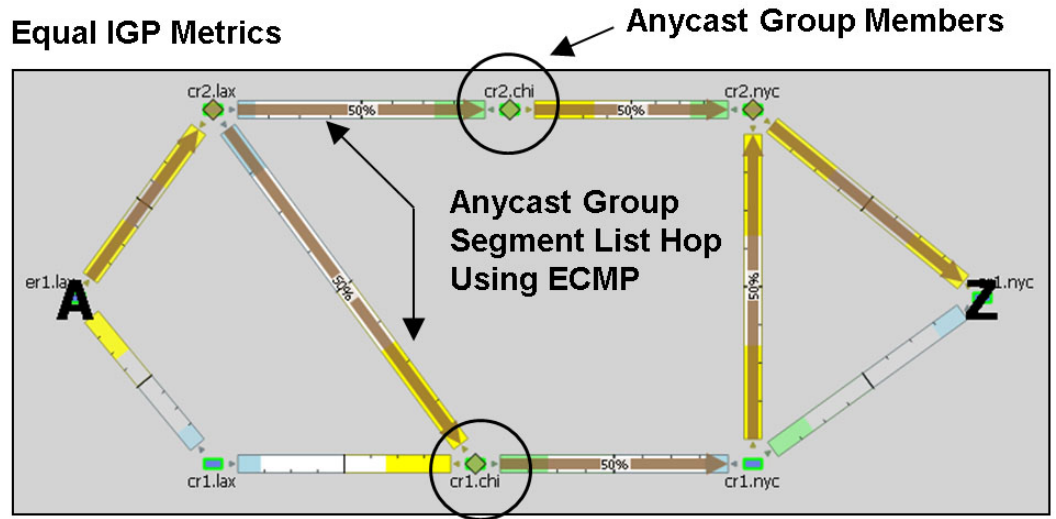
40/29/26

Anycast Group Segment List Hops

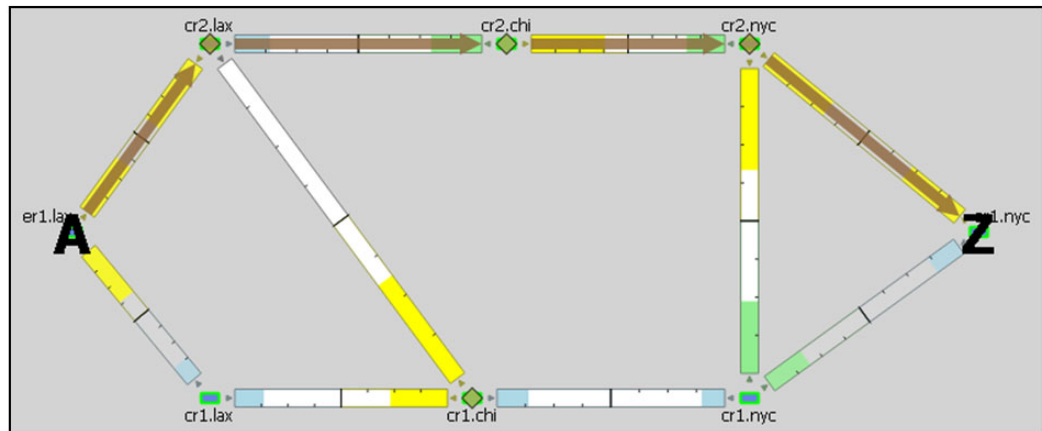
An SR LSP routes through the node in an anycast group segment list hop that has the shortest path to it. In case of a tie between multiple nodes in an anycast group, ECMP is applied. This mechanism lets you impose routing restrictions in terms of potential intermediate segment destinations. It also lets the SR LSP choose among the possible next hops (next segment list hops), potentially reducing latency and improving load balancing.

Figure 67: Example Anycast Group Segment List Hop, on page 183 shows an example of anycast group nodes cr2.chi and cr1.chi. In one instance, the IGP metrics to those nodes have equal IGP metrics, so the SR LSP uses ECMP to route through both of them. When the IGP metric increases from cr2.lax to cr1.chi, the SR LSP routes only through cr2.chi because it has the shortest path. However, as Figure 68: Example Anycast Group Rerouting Around Failure, on page 184 shows, if a failure prevents the shortest path from being used, the SR LSP can still route through the anycast group using the next-highest cost path.

Figure 67: Example Anycast Group Segment List Hop

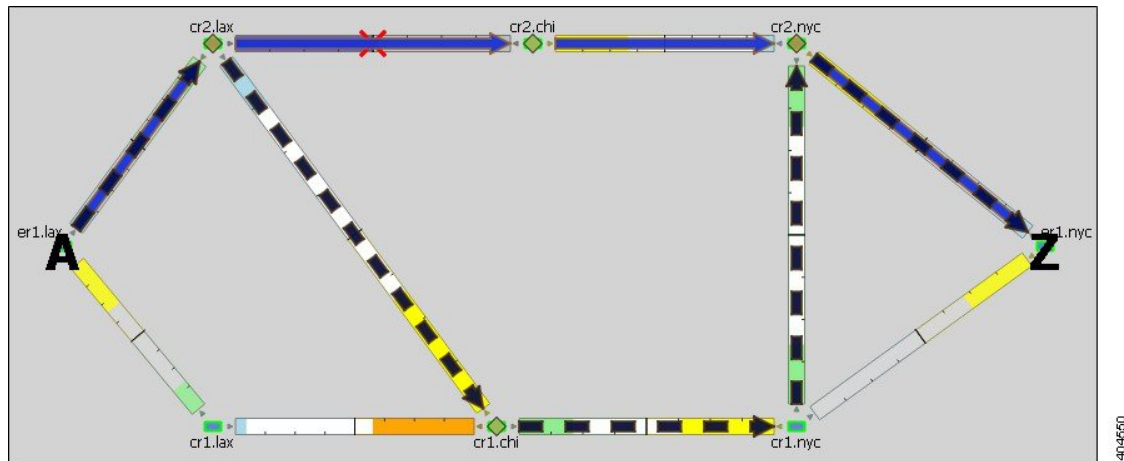


SR LSP Route Based on Anycast Group Node with Shortest Path



408275

Figure 68: Example Anycast Group Rerouting Around Failure



SR LSP Paths

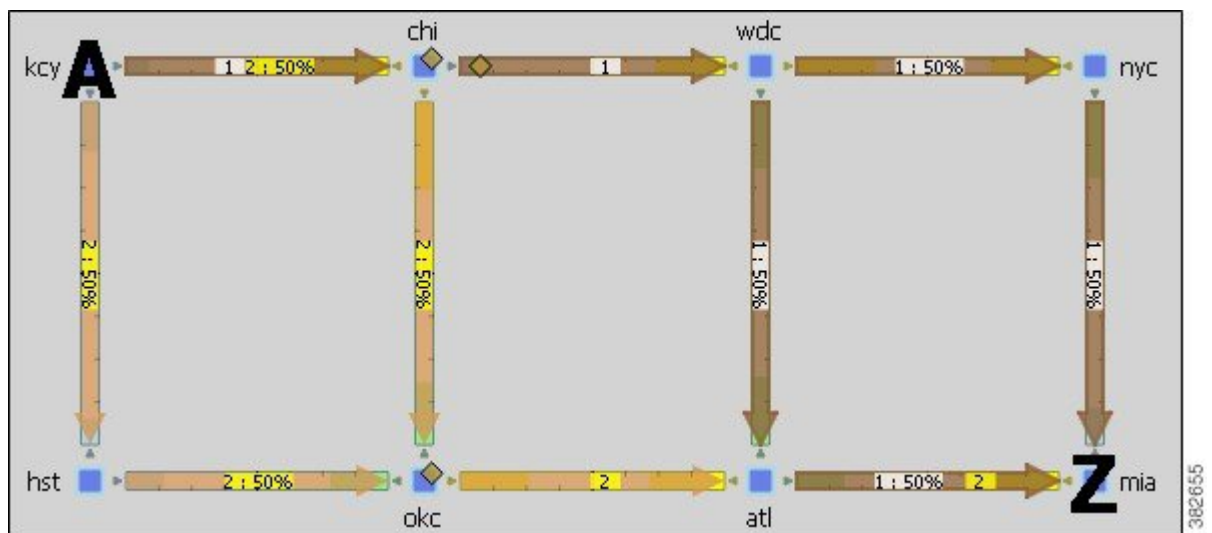
SR LSPs can have multiple LSP paths, where each LSP path has its own segment list. There is no distinction between standby and non-standby LSP paths for SR LSPs.



Note If both the SR LSP and its SR LSP paths have segments, the segments on the LSP paths override those on the SR LSP.

Figure 69: Example SR LSP Paths, on page 184 shows an SR LSP from key to mia that has a primary and a secondary path. The primary path is configured with a chi node segment list hop and chi-to-wdc interface segment list hop, and the secondary path is configured with a node segment list hop in okc. This can be seen by selecting the LSP paths individually from the LSP Paths table.

Figure 69: Example SR LSP Paths



SR LSP Routing

Assuming the common LSP mechanisms such as autoroute and forwarding adjacencies have determined that a demand will enter an SR LSP, the LSP traffic is routed as follows. If the destination node or segment list hops are not reachable, the SR LSP is not established.



Note The process of creating final segment list hops can be automated using the [Traffic Demand Modeling, on page 43](#) tool.

- If the SR LSP does not have a segment list, the demand is routed via IGP to the destination node.
If the SR LSP has a segment list, the demand is routed using the segment list hops within the segment list in the sequential order in which they appear.
- If it is using a node segment list hop, the demand takes the shortest path to that node.
If it is using an interface segment list hop, the demand takes the shortest path of the source node of that interface, and then routes to that interface's destination node.
If it is using an anycast group segment list hop, the demand routes through the node with the shortest path. In case of a tie between multiple nodes in an anycast group, the SR LSP uses ECMP to route.
- If you select both a node and an anycast group containing that node when creating the segment list hop, the node segment has precedence.
- If the SR LSP contains LSP paths, the demand is routed on the LSP path with the lowest path option for which the demand can be routed from source to destination. Demands for SR LSP paths route in the same manner as described above for SR LSPs.
- If the SR LSP contains multiple SR Candidate paths, the demand is routed on the LSP path with the highest preference option for which the demand can be routed from source to destination. Demands for SR LSP paths route in the same manner as described above for SR LSPs.
- If the demand cannot be routed on the segments defined in the SR LSP or its LSP paths (for example, because of a failure in a node segment list hop), the demand is routed to the destination using the IGP shortest path.
- If a hop (nodes, interfaces, LSPs, or anycast groups) contains a segment ID (SID), the segment list hop inherits that SID.

Inter-AS SR LSP Routing

SR LSPs with source and destination nodes in different ASes can route provided the following hop configurations exist:

- There is a node segment list hop on the border of the AS that the LSP is exiting. This node is the exit node.
- One of these hops exists:

- A node segment list hop exists on the first node traversed in another AS. This node is the entry node. If there are multiple peering interfaces, their metrics determine which one the LSP uses.
- An interface segment list hop connects the exit node of one AS to an entry node in another AS.

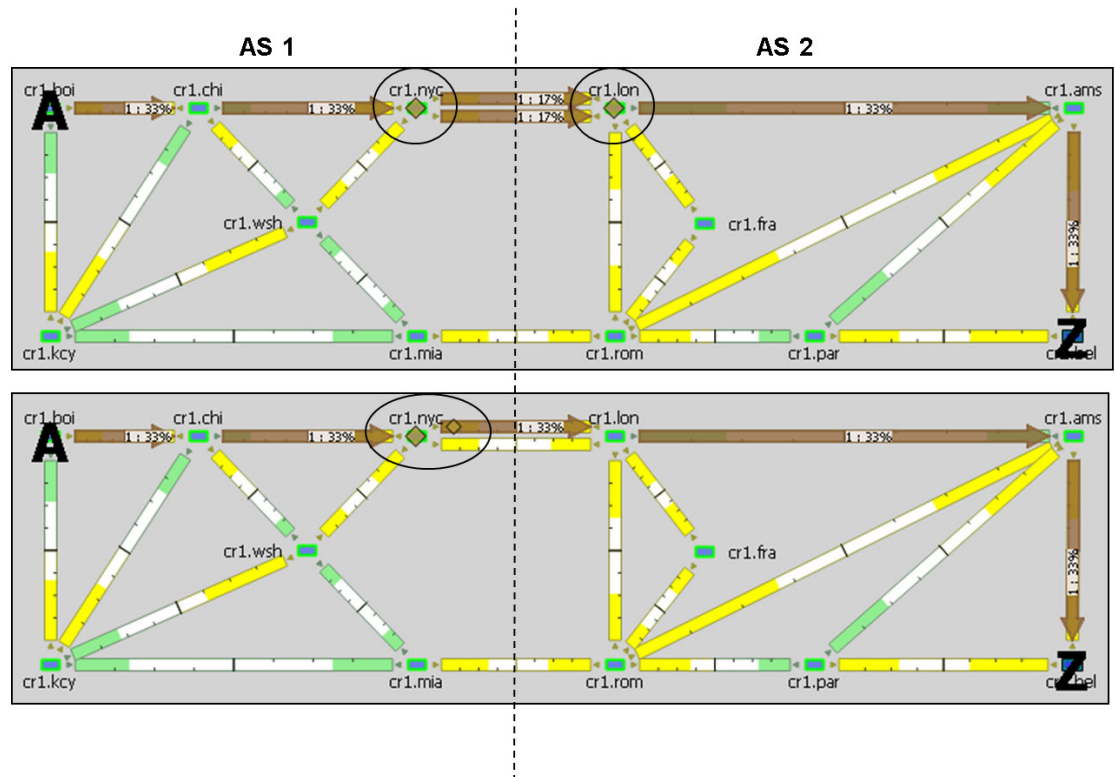
Figure 70: Segment List Hops Define Exit and Entry Points Across AS, on page 186 shows an example of each of the preceding configurations. Both networks show two ASes, and both show a node segment list hop on cr1.nyc, which is the node from which the LSP is exiting AS 1. The top half shows the cr1.lon node, the entry node to AS 2, as the next segment list hop. The lower half shows the cr1.nyc-to-cr1.lon interface, which connects the exit node in AS 1 to an entry node in AS 2, as the next segment list hop.

Without these conditions being met, SR LSPs cannot route across AS boundaries. For example, Figure 71: Inter-AS SR LSP Routing Requires Correct AS Exit and Entry Hops, on page 187 shows an SR LSP with the proper node segment list hop set on cr1.mia (exit node of AS 1), but it does not route because the next segment list hop (cr1.par) is not the first node in AS 2 that this LSP would traverse. By adding a node segment list hop on cr1.rom, the SR LSP can route across the ASes.

Using demands with inter-AS SR LSPs requires that two conditions be met:

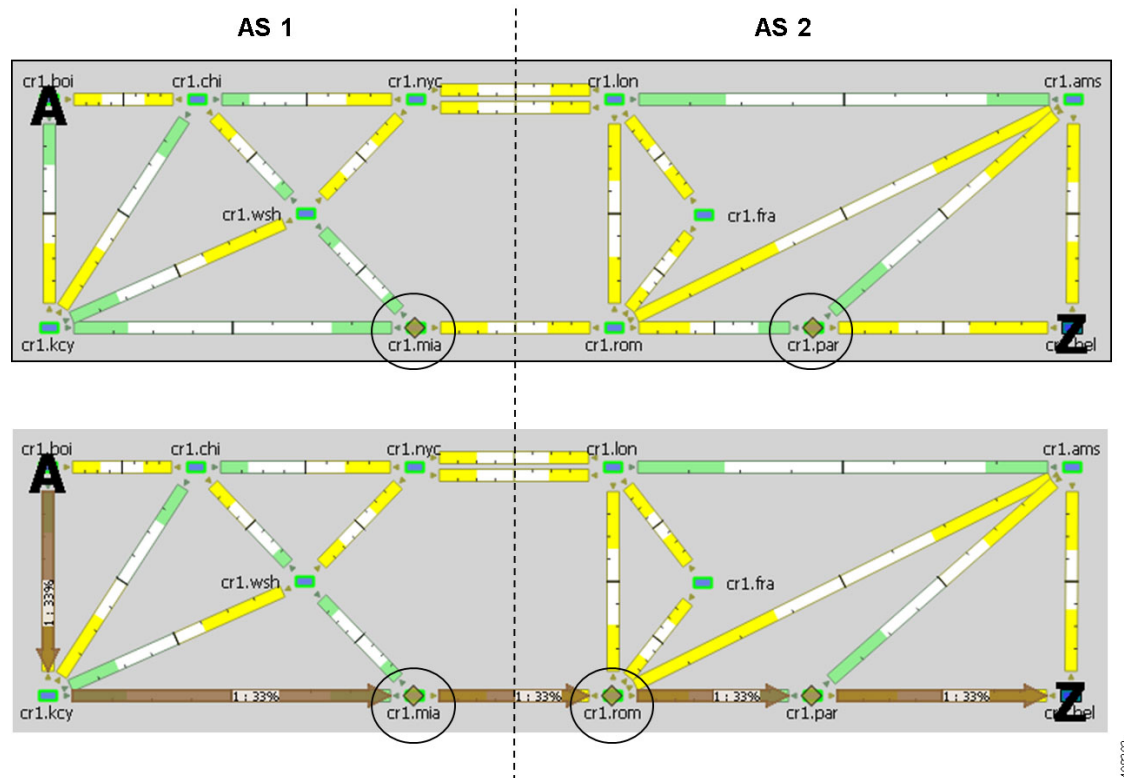
- The demand must be private to the SR LSP. For information on creating private LSPs, see [MPLS Simulation, on page 113](#).
- The demand must have the same source and destination as the SR LSP.

Figure 70: Segment List Hops Define Exit and Entry Points Across AS



406304

Figure 71: Inter-AS SR LSP Routing Requires Correct AS Exit and Entry Hops



408303

Creating SR LSPs and Their LSP Paths

SR LSPs and mesh SR LSPs are created the same way RSVP-TE LSPs are created. The difference is that you must set the Type property to SR.



The SR LSP has a color property which is a 32-bit numerical value.

Once created, the next step is to either create the segment list or create LSP paths and their segment lists.

Likewise, LSP paths are created in the same way as RSVP-TE LSP paths. Because you are creating them from SR LSPs, you can edit the properties Path Option and Active. Set the property Protocol Origin to identify the component or protocol that originates the SR LSP path. Choose from Local, PCE Initiated, and BGP Initiated.

For information on creating LSPs and LSP paths, see [MPLS Simulation, on page 113](#).

Creating Segment Lists

You can create segment lists and destination segment lists hops automatically using the Explicit LSP Paths initializer. This initializer creates as few segment list node and interface hops as possible to route according to the options selected. The destination node of the LSP is added as the final segment list hop unless the final hop is a segment list interface hop that terminates on the destination node.

Step 1 Choose **Insert > LSPs > Segment List**, or right-click in the plot and choose **New > LSPs > Segment List**.

Step 2 Add the segment list hops in the order in which they are to be followed.

- To add a new segment list hop, select an existing hop (if applicable), and then click **Insert Before** or **Insert After** to determine where the new hop goes sequentially. Go to Step 2.
- To edit an existing segment list hop, select it from the list and then click **Edit**. Go to Step 2.
- To delete an existing segment list hop, select it from the list, click **Delete**, and then click **OK**. You need not continue.

Step	Node	Interface
1	cr2.sjc	
2	cr2.sjc	to_cr2.lax
3	cr1.hst	
4	cr1.hst	to_cr1.atl

Hop Specified by:

Site:

Node:

Interface:

LSP:

Anycast Group:

408256

Step 3 To continue creating or editing a segment, use the following options and then click **OK**. As needed, choose the site, node, interface, or anycast group.

Note Interface segment list hops must be the egress interface of a node segment list hop.

- If the node or source node (for interfaces and LSPs) is within a site, first choose the site.
- For node segment list hops, do not choose an interface.
- For interfaces and LSPs segment list hops, first choose the source node.
- For anycast group segment list hops, do not choose a site, node, or interface.

After the segment list is created, another way to manage segments is to double-click one from the Segment List Hops table. From there you can change the site, node, and interface, but not the sequence.

Creating Anycast Groups

-
- Step 1** Choose **Insert > Anycast Group**, or right-click in the plot and choose **New > Anycast Group**.
- Step 2** Enter a unique name to identify the anycast group.
- Step 3** For each node you want to include in or exclude from the anycast group, check the **T** (true) or **F** (false) check box. Then click **OK**.
-

SR-TE Protection

If the network contains a configurable topology-independent loop-free alternate (TI-LFA), you can simulate SR-TE tunnels before and after convergence.



Note To see the simulation before convergence, you must be in Fast Reroute mode (choose **Edit > Network Options**; in the Simulation Convergence Mode area, choose **Fast Reroute**). In IGP and LSP Reconvergence mode, you can only see the after-convergence routes.

SR-TE Tunnel Simulation Before Convergence

For each SR tunnel in a plan file, and for each SID in the path, WAE Design determines the no-failure route.

In a *normal path state*, WAE Design determines the no-failure route from segment list hop to segment list hop. Between segment list hops, WAE Design follows the IGP shortest path to the next segment list hop. At each interface along the path, if the connected circuit is down, WAE Design determines whether the next segment list hop is an interface or a node, and whether the SR tunnel is routable.

- If SR FRR is enabled for the interface in the Interfaces table, WAE Design configures the *protect path state*:
 - WAE Design derives the post-convergence path from the local node to the next SID hop with the subject circuit removed from the topology.
 - The next SID hop depends on the SID hop type. For an interface SID, the next hop is the remote node of that interface. For a node SID, the next hop is that node.
 - WAE Design follows the derived after-convergence path hop by hop until it reaches the next SID hop. (If a failure occurs along the way, the SR FRR and the SR LSP do not route.)
 - When WAE Design reaches the next SID hop, it returns to the normal path state and resumes following the no-fail path to the SID hop.
- If SR FRR is not enabled for the interface in the Interfaces table, the SR tunnel does not route.

SR-TE Tunnel Simulation After Convergence

WAE Design routes to each segment list hop through the IGP shortest path under all failure conditions.



Note An SR LSP might be routable before convergence through a TI-LFA, but unroutable after convergence.

Constraints

- In the Interfaces table, if SR FRR Enabled is T (true) for a given interface, all SR LSPs with a node or interface SID as the next hop are protected over that interface. If the SRR FRR Enabled is F (false), all SR LSP SIDs are unprotected over that interface.
- Protected versus unprotected adjacency SIDs are not supported. (All adjacency SIDs are considered protected.)
- WAE Design does not impose a limit on the TI-LFA path's label stack depth.
- WAE Design does not explicitly derive P and Q nodes, but instead derives a protect path that aligns with the after-convergence shortest path from the PLR to the next segment list hop.
- If any part of the TI-LFA path encounters a failure before reaching the next segment list hop, the TI-LFA path fails and the SR LSP does not route.



CHAPTER 12

Layer 1 Simulation

Although WAE Design focuses on IP/MPLS Layer 3 (L3) network views and traffic, you can also create and view Layer 1 (L1) topologies for a more complete view of the network. Using the L3 and L1 topologies together, you can specify how L3 circuits route over L1 links. These Layer 3-to-Layer 1 associations let you simulate individual L1 link and L1 node failures, or include L1 links and L1 node failures in a Simulation Analysis to determine the effects on L3 traffic. Additionally, you can include L1 links and L1 nodes in SRLGs. With this interaction of layers, you can better design the network to meet performance requirements.



Note Aside from this chapter, throughout this guide the term *circuit* means a circuit with L3 properties. In this chapter, for purposes of clarity, *L3 circuit* and *L1 circuit* refer to objects defined in the Circuits and L1 Circuits tables, respectively, whereas *circuit* refers to the actual network element.

This section contains the following topics:

- [L1 and L3 Views, on page 191](#)
- [L1 Topology, on page 193](#)
- [L1 Simulations, on page 195](#)
- [Relationships Between L1 and L3, on page 208](#)
- [Delay and Distance, on page 212](#)
- [L1 Failures, on page 214](#)
- [Inactive State, on page 219](#)
- [L1 Simulation Analysis, on page 220](#)
- [Creating L1 Topology, on page 222](#)
- [Creating L1 and L3 Mappings, on page 230](#)
- [Visualizing the L1 Network, on page 235](#)

L1 and L3 Views

[Figure 72: Comparison of L1 and L3 Views, on page 192](#) shows an example L1 view in comparison to an L3 view. Notice the different uses of circuits. In the L3 view, L3 circuits are relevant to IP/MPLS routing and show traffic. The L1 view shows the physical topology with no traffic; if selected, it also shows the L1 circuit paths taken across the L1 links. L3 is the default view unless the plan file does not contain L3 nodes.

To more easily find relationships between L3 and L1 objects, right-click the object and choose the appropriate filter.

Alternate Views

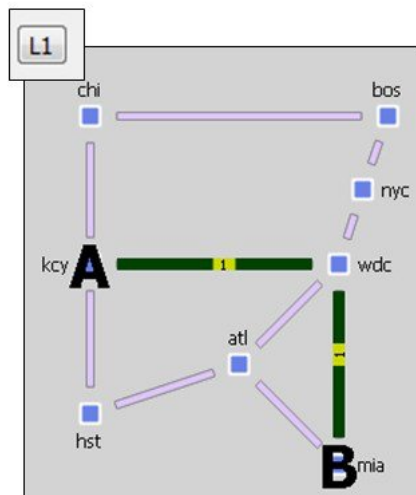
To alternate between L1 and L3 views, use one of these methods:

- Click the L1/L3 toggle button in the top left of the Visualization toolbar.
- Choose **View > Layer 1 View** or **View > Layer 3 View**.
- Use keyboard shortcuts **Ctrl-1** for L1 and **Ctrl-3** for L3. (On a Mac, use **Alt-1** and **Alt-3**.)

Figure 72: Comparison of L1 and L3 Views

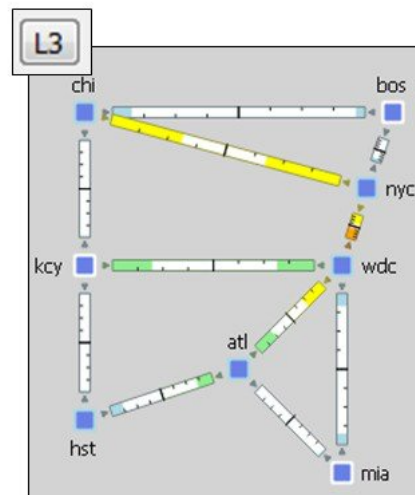
Layer 1 View

Shows physical topology and L1 links, but does not show traffic. A-B shows a selected L1 circuit path.



Layer 3 View

Shows IP/MPLS topology and routing, including measured and simulated traffic on L3 circuits.



380921

Multi-Layer Views

To show relationships between the L1 and L3 views, you can simultaneously view both of them in either view (Figure 73: Multi-Layer (Combined Layer 1 and Layer 3) Views, on page 193). If you are in the L1 view, L3 objects appear in the background. If you are in the L3 view, L1 objects appear in the background. Additionally, the L3-L1 links, which model the physical relationship between L3 nodes and L1 nodes, are visualized (Figure 74: L1-L3 Link Visualization, on page 193).

In the Plot Options dialog box, click the **Layer 1** tab, and then choose options to determine whether to show Layer 1 objects in the Layer 3 view. For more information, see [Editing L1 Plot Options, on page 236](#).

The ability to move, select, and edit properties for an object in the background is toggled on and off with **View > Preferences > Enable Selection of Background Objects**.

Figure 73: Multi-Layer (Combined Layer 1 and Layer 3) Views

A-B shows a selected L1 circuit path in both views

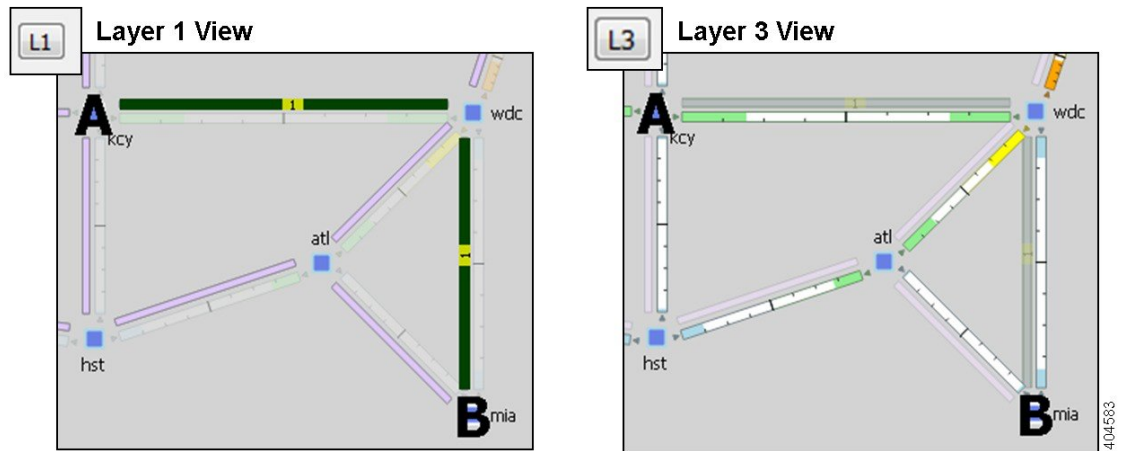
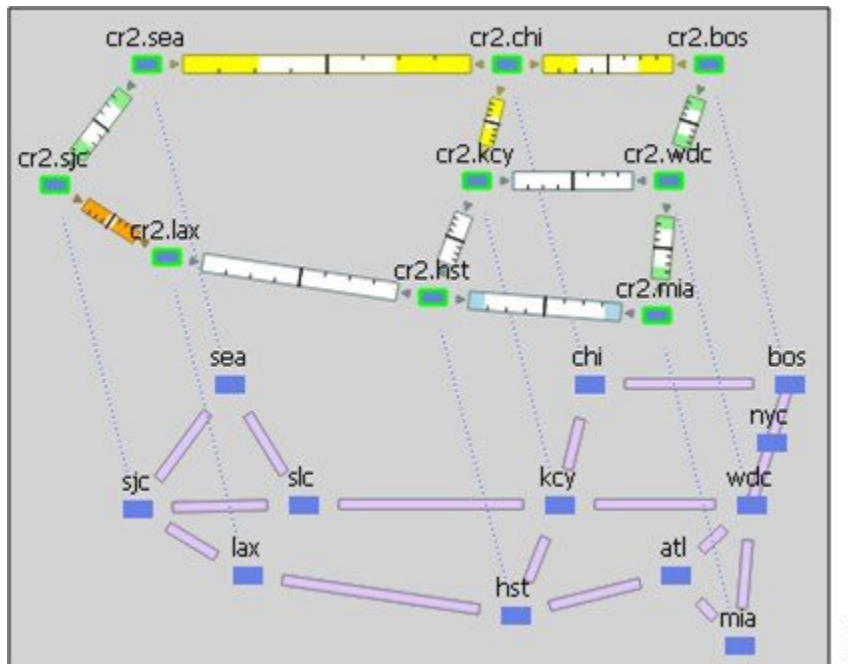


Figure 74: L1-L3 Link Visualization



L1 Topology

An L1 topology consists of a unique set of objects (Table 10: L1 Objects , on page 194) that are distinct from IP/MPLS L3 topology and objects. However, two of the objects, L1 circuits and L3 circuits, represent the same circuit.

- An L3 circuit contains L3 properties of the circuit.

- An L1 circuit contains L1 properties, specifically L1 routing, of the circuit.

You can create the objects and generate an L1 topology in WAE Design using these methods:

- Import an L1 topology from another plan file. This method creates the same L1 topology and the same L1 link, L1 circuit, L1 port, and L3 mappings as in the source file. You can also copy an L1 topology from a plan file. To import an L1 topology, choose **File > Import > Layer 1**.
- Construct an L1 topology based on all or part of an existing L3 topology in the same plan file. This method creates an L1 topology that mirrors the L3 one, and automatically generates the mappings between the L1 and L3 objects. For more information, see [Creating L1 Topology from L3 Topology, on page 222](#).
- Individually create the L1 ports, L1 nodes, L1 links, L1 circuits, L1 circuit paths, and L1 circuit path hops. You can add both L1 nodes and L1 links to SRLGs for simulation purposes. This is typically done to represent ducts containing multiple fiber segments, where a break in the duct could cause multiple L1 link failures, and thus potentially multiple L3 circuit failures for each of these L1 links. The end of this chapter contains several sections on manually creating L1 objects.
- Create the L1 topology directly in the tables in the plan file, such as by editing the L1 tables (**Edit > Plan Tables as Database**) or by creating the table in the plan file using a text editor.

L1 Objects

Table 10: L1 Objects

L1 Object	Description
L1 port	Models L1 ports and is listed in the L1 Ports table. The source and destination for L1 circuits are L1 ports. Each L1 port can belong to only one L1 circuit. An L1 port can be associated with an L3 port.
L1 node	Models L1 devices and is listed in the L1 Nodes table. When selected, they become a darker shade of blue.
L3-L1 link	Models the physical relationship between an L3 node and an L1 node. If showing both L1 and L3 objects, these appear as light blue dotted lines. If selected, they are solid black lines.
L1 link	Models the fiber segments connecting the L1 nodes. These are listed in the L1 Links table and are highlighted in blue when selected. L1 circuits are routed over L1 links, which is how these L1 links can be associated with the IP/MPLS topology in the L3 view.
L1 link waypoint	Models network devices (that are not L1 nodes) that reside in the L1 link path. Each waypoint has coordinates that identify a point on an L1 link. They are used for visualization purposes and for calculations executed by the Latency and Distance initializer. They are not used in simulations. They appear as dots on the L1 link, and they become darker when selected.
L1 circuit	Models the L1 properties of a circuit, such as which L1 ports are connected to it and the physical path and distance. These are listed in the L1 Circuits table and are highlighted in green when selected. Each L1 circuit can be associated with an L3 circuit directly or indirectly.

L1 Object	Description
L1 circuit path	Models the paths available for the L1 circuits. These are listed in the L1 Circuit Paths table and are highlighted in green when selected. The number on the highlighted path indicates the Path Option number.
L1 circuit path hop	Models the L1 node or L1 links in the L1 circuit path. These are listed in the L1 Circuit Path Hops table.
Actual L1 circuit path hop	Discovered L1 circuit path hops.

L1 links, L1 nodes, and L1 ports can be failed, set to an inactive state, and included in Simulation Analysis to calculate a worst-case failure analysis. Additionally, L1 links and L1 nodes can both be included in SRLGs, which in turn can also be failed and included in Simulation Analysis. These capabilities let you assess the effect of L1 outages on the IP/MPLS network, thus improving your ability to design around and plan for potential service disruptions.

L1 Waypoints

Waypoints are only viewable in the geographic view, and by default, they are visible in the network plot. There is an option to show or hide them in the Layer 1 tab of the Plot Options dialog box.



- The Order property represents the proximity to L1 node A, where 1 is the closest waypoint to the L1 node A and each subsequent number is farther away.
- Default Longitude and Latitude properties are based on the L1 link on which they are created, though you can change them.

Sites

All sites are listed in the Sites table whether they contain L1 nodes, L3 nodes, or both. To determine how many L1 nodes, if any, are in a site, show the L1 Node Count column. By default, sites containing only L1 nodes appear only in the L1 view, and sites containing only L3 nodes appear only in the L3 view. You can change this default in the Plot Options dialog box. Empty sites appear in both views.

L1 Simulations

When reading the L1 tables, the derived Established and Routed properties indicate simulation capabilities.

- Established—Identifies whether an L1 circuit or L1 circuit path is routed when there are no failures. Note that while an L1 circuit or L1 circuit path has Established equals F (false), it might be routable under failures.
- Routed—Identifies whether and how an L1 circuit or L1 circuit path is routed:
 - Unrouted
 - Actual (if being routed through actual L1 circuit path hops)

- Simulated (if being routed and not using actual L1 circuit path hops)



Note You can configure the number of attempted routes for L1 circuit paths. Increasing the number of attempted routes allows for more paths to be explored, thereby potentially reducing the number of unrouted L1 circuits. This increases the overall computation time. To configure the number of route attempts, choose **Edit > Network Options** and set the **Number of Attempts to Route L1 Circuit Paths** parameter.

Active L1 Path Selection

Because multiple L1 circuit paths can be routed for a single L1 circuit, WAE Design uses the Active Path Sim property to determine which L1 circuit path to use to set the derived properties for each L1 circuit, such as Distance Sim and Delay Sim. The Active Path Sim property is copied from the user-specified Active Path property if it has been set and if it is routable.

Standby Paths and Path Options

The user-specified L1 circuit path's Standby property sets the path to standby (T) or non-standby (F). The user-configurable Path Option property is an integer that identifies the order in which the routing of L1 circuit paths is explored. The lower the value, the higher the priority to route the L1 circuit path.

If other L1 circuit paths must be explored, the standby L1 circuit path with the lowest Path Option value is used provided it is routable. If it is not routable, attempts are made for each Path Option with the next lowest value (next highest priority). If none of the standby L1 circuit paths are routable, WAE Design then follows the same procedure with non-standby L1 circuit paths, starting with the one with the highest priority Path Option.



Note Non-standby L1 circuit paths reroute under failure only you check the **Network Options > Layer 1 rerouting convergence under failures** check box. For more information, see [Resiliency Under L1 Failures, on page 216](#).

L1 Circuit Path Routing Order

- WAE Design always tries to route L1 circuit paths that have their Standby property set to T (true), regardless of whether they are set as the active path. If they are routable, they are routed in the order of this priority:
 - Actual L1 circuit path hops have first priority. Note that while these are not discovered by WAE, if they are discovered through another means, you can manually add them using the Actual tab of the L1 circuit paths Properties dialog box or by adding an <ActualL1CircuitPathHops> table to the .txt plan file.
 - L1 circuits paths with explicit central frequency IDs or central frequency IDs defined through L1 circuit path hops are then routed. For information on central frequency IDs, see [Central Frequency ID Assignments, on page 204](#).

- Thereafter, L1 circuit paths that do not have central frequency ID properties set explicitly or through L1 circuit path hops are routed.
- After attempting to route standby L1 circuit paths, WAE Design determines whether and how to route non-standby L1 circuit paths (Standby property set to F).
 - WAE Design determines if the non-standby L1 circuit path should route by checking the following L1 circuit properties in this order.

If the L1 circuit Active Path property is set to this L1 circuit path, WAE Design tries to route the non-standby L1 circuit path.

If the number of routed L1 circuit paths is less than the associated L1 circuit's Min Routed Paths property, WAE Design tries to route the non-standby L1 circuit paths.

- Non-standby L1 circuit paths are routed based on their Path Option setting (lower numbers are routed first). For each of these paths, the route order is based on the same criteria as described above for standby L1 circuit paths (actual L1 circuit path hops, then central frequency ID values, and then L1 circuit path hops that do not have central frequency ID values).

Within each of the above routing instances, WAE Design routes the L1 circuit path based on the following criteria regardless of whether the L1 circuit path is standby or non-standby.

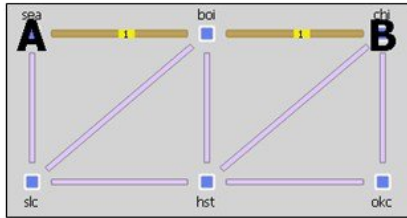
- Routes are based on the lowest Setup Priority value and then deterministically. For example, among L1 circuit paths with actual L1 circuit path hops, those with higher Setup Priority are routed first. See .
- L1 circuit paths that do not require automatic wavelength assignments to route have higher priority than those that do. For information on the Auto Spectrum Assignment property, see [Central Frequency ID Assignments, on page 204](#).

Viewing L1 Circuit Paths

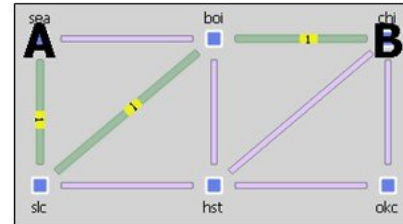
To view the actual paths or simulated paths of L1 circuits, choose **View > L1 Circuits**. Note that given the priority of routing, if the actual L1 circuit path hops path is valid, the actual and simulated paths are the same. You can also choose to view the shortest latency path from this menu.

Rerouted simulated L1 circuit paths show automatically when they are rerouted due to failures, regardless of whether you are showing simulated paths.

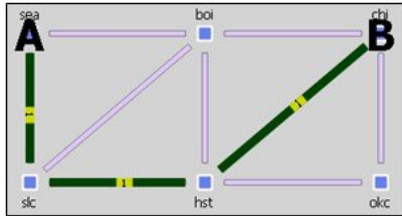
Actual L1 Circuit Path



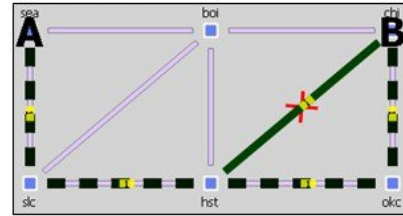
Shortest Latency L1 Circuit Path



Simulated L1 Circuit Path



Simulated L1 Circuit Path Rerouted Under Failure



383000

L1 Circuit Path Hops

You can force an L1 circuit path to choose a different route by setting one or more node hops or link hops.

Node Hops	Link Hops
<ul style="list-style-type: none"> • Loose—The path must pass through the node, but the node does not have to be connected directly to the previous hop. • Strict—The node must be directly connected to the previous node. • Exclude—This node cannot be in the path. 	<ul style="list-style-type: none"> • Strict—The L1 link must be in the path in the specified order. • Exclude—This L1 link cannot be in the path.

Generally, node hops are used for simplicity, but link hops are useful for when you need to distinguish between two parallel L1 links. To do this, set a loose hop on the L1 node preceding the link hop, and then set a strict hop on the parallel L1 link over which you want to route this L1 circuit.

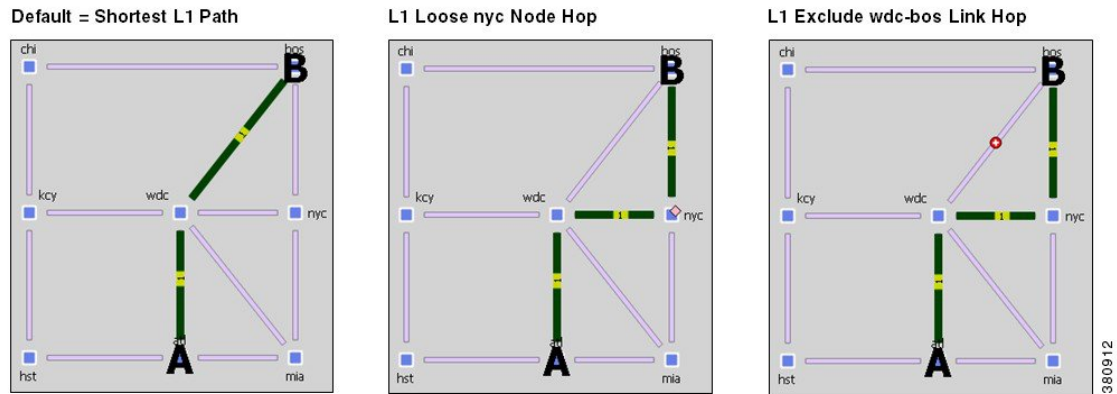
You can set L1 circuit path hops in an L1 circuit path Properties dialog box. Because WAE Design always uses the shortest path, remember to set as many hops as needed to achieve the desired path, and put them in the order of the desired path. The L1 Circuits Path Hops table identifies the path and the steps.



Note Because L1 circuits paths are bidirectional, to simulate L1 circuit path hops WAE Design automatically adds the initial L1 node in the path as a strict hop.

Example: [Figure 75: Shortest L1 Circuit Paths Versus Paths Determined by L1 Circuit Path Hops, on page 199](#) shows an example of a default shortest path being used, as well as an alternate circuit path created using a loose nyc node hop. It also demonstrates that you could achieve the same path by setting only an exclude link hop on wdc-bos.

Figure 75: Shortest L1 Circuit Paths Versus Paths Determined by L1 Circuit Path Hops



For More Information...	See...
Automate the process of setting up L1 circuit hops	Explicit L1 Circuit Paths Initializer, on page 199
Set L1 circuit hops per L1 node or L1 link	Creating Individual L1 Circuit Path Hops, on page 227

Explicit L1 Circuit Paths Initializer

The Explicit L1 Circuit Paths Initializer automates the process of creating explicitly routed L1 circuit paths.

- Step 1** Choose one or more L1 circuit paths. If you select none, the default is to create L1 circuit path hops for all L1 circuit paths.
- Step 2** Choose **Initializers > Explicit L1 Circuit Paths**, or right-click an L1 circuit path and choose **Explicit L1 Circuit Path Initializer**.
- Step 3** Choose whether to use all L1 circuit paths, those selected in the table, or those with specific tags.
- Step 4** Click **OK**.

L1 Circuit Route Reports

You can create reports on L1 circuits based on the L1 circuit path identified as the Active Path Sim for each L1 circuit. Additionally, the report contains all other columns in an L1 Circuits table, though they are initially hidden.

- Step 1** Choose **Tools > Reports > Layer 1 Circuit Routes**.
 - Step 2** Click **OK**.
- After closing the report, you can access it again by choosing **Window > Reports**.
- To export the report, right-click the **Routes** report name in the Reports navigation pane, and choose **Export**.

L1 Link Metrics

When L1 circuit paths are routed over L1 links, by default they are established along the shortest path between the L1 nodes. This shortest path is defined as the path with the lowest metric, and the metric of an L1 circuit path is defined as the sum of the L1 link metrics in the path. To move L1 circuit paths away from an L1 link, raise the metric on that L1 link.

Because you can simultaneously edit multiple L1 link Metric properties, this method is useful for altering numerous L1 circuit paths at a time. For exact path specification per L1 circuit path, however, you must set explicit L1 circuit path hops.

L1 Link



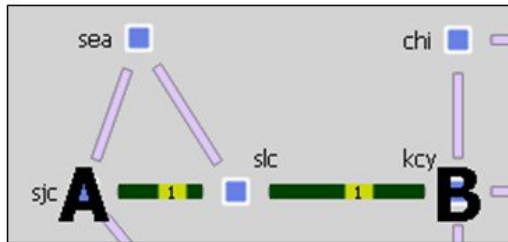
If the two shortest L1 circuit paths have the same metric, WAE Design chooses one arbitrarily provided no other properties impact the route.

Note that in contrast to an interface IGP metric, this L1 metric does not represent an actual configurable quantity in the network.

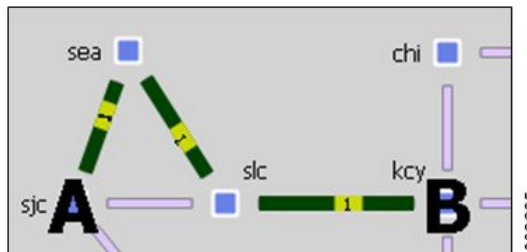
Example: [Figure 76: Example Effect of L1 Link Metric on Routing L1 Circuits, on page 200](#) shows an example L1 circuit from sjc to kcy. When the sjc-slz L1 link has a metric of 2.7, the shortest L1 circuit path is sjc-slz-kcy. By raising the sjc-slz metric, the shortest L1 circuit path becomes sjc-sea-slz-kcy.

Figure 76: Example Effect of L1 Link Metric on Routing L1 Circuits

L1 Links		
L1 Node A	L1 Node B	Metric
kcy	slc	12.000
sea	sjc	7.500
sjc	slc	2.700
sea	slc	8.000



L1 Links		
L1 Node A	L1 Node B	Metric
kcy	slc	12.000
sea	sjc	7.500
sjc	slc	20.700
sea	slc	8.000



L1 Link Metric Initializer

The L1 Link Metric Initializer sets L1 metrics based on L1 link distances. WAE Design calculates the metric by multiplying each L1 link's Distance value by the multiplier you specify. You also set a minimum value for the resulting metric. WAE Design then uses the higher of the two values.

Example: If the L1 Distance value is 7, the multiplier value is 3, and the minimum value is 1, the resulting L1 metric is 21 (Distance * Multiplier). However, if the minimum value were 25, the resulting L1 metric would be 25.

Step 1

Make your selections.

- Choose one or more L1 links.
- If you make no selections, metrics are calculated for all L1 links.

Step 2

Choose **Initializers** > **Layer 1 Metric**, or right-click an L1 link and choose **Initialize L1 Metric**.

Step 3

Enter a multiplier and minimum value, and click **OK**.

L1 Circuit Path Setup Priority

The Setup Priority property identifies the order in which L1 circuit paths are routed when they share the same L1 link. The lower the number, the higher the priority. [Figure 78: Example Spectrum Blocking, on page 207](#) shows an example of rerouting L1 circuit paths based on Setup Priority.

L1 Circuit Path

To set the Setup Priority, right-click one or more L1 circuit paths and choose **Properties**. This value appears in the L1 Circuit Paths table in the Setup Priority column.

L1 Circuit Wavelengths

The L1 circuit central frequency ID (wavelength) properties include those that identify transmission feasibility and central frequency ID assignments.

Using these properties, you can answer questions like:

- Can I add an L3 circuit that uses a specific L1 circuit?
- How will my network choose a new L1 circuit path under L1 failure?
- Can I modify an existing L1 circuit for an L3 circuit?
- How many available wavelengths are there in my L1 topology?
- What happens to the network when there are L1 failures? Does optical restoration select a new L1 circuit path, and if so, which path?

Transmission Feasibility

In a real network, the quality of an L1 circuit deteriorates as it passes through L1 links. Using feasibility properties, WAE Design lets you simulate this weakening of L1 circuits to determine if they have degraded to the point of being unroutable.

Feasibility Property	Description
L1 Link	
Noise	Identifies unidirectional properties for L1 Links. <ul style="list-style-type: none"> NoiseAtoB (default is undefined): The noise introduced by L1 Link from L1 Node A to L1 Node B in dB. NoiseBtoA (default is undefined): The noise introduced by L1 Link from L1 Node B to L1 Node A in dB.
NoiseSigma	Identifies unidirectional properties for L1 Links. <ul style="list-style-type: none"> NoiseSigmaAtoB (default is undefined): The noise sigma introduced by L1 Link from L1 Node A to L1 Node B in dB. NoiseSigmaBtoA (default is undefined): The noise sigma introduced by L1 Link from L1 Node B to L1 Node A in dB.
L1 Circuit	
Feasibility Metric Sim	This is copied from the Feasibility Metric Sim of the active L1 circuit path.
Feasible Transmission	Identifies whether the L1 circuit can be transmitted. This is copied from the Feasible Transmission of the active L1 circuit path. If all L1 circuit paths have a higher Feasibility Metric Sim than Feasibility Limit, this value is F (false), and the L1 circuit cannot route.
L1 Circuit Path	
Feasibility Metric Sim	The simulated Feasibility Metric of the L1 Circuit Path in dB.
Feasibility Limit	Identifies the acceptable quality of the L1 circuit path.
Feasibility Acceptance Threshold	Identifies the acceptance threshold used in calculating the L1 circuit path feasibility metric. Lower thresholds increase the risk that the L1 circuit path is not feasible in the network. The default is 3.
Feasible Transmission	Identifies whether the L1 circuit path can be transmitted. If the L1 circuit path's Feasibility Limit is more than the sum of the Feasibility Metrics for the L1 links that the L1 circuit path traverses, this value is T (true) and the L1 circuit path can route. <p>Note This property does not apply if an L1 circuit uses actual L1 circuit path hops.</p>

Routing Based on Feasibility Properties

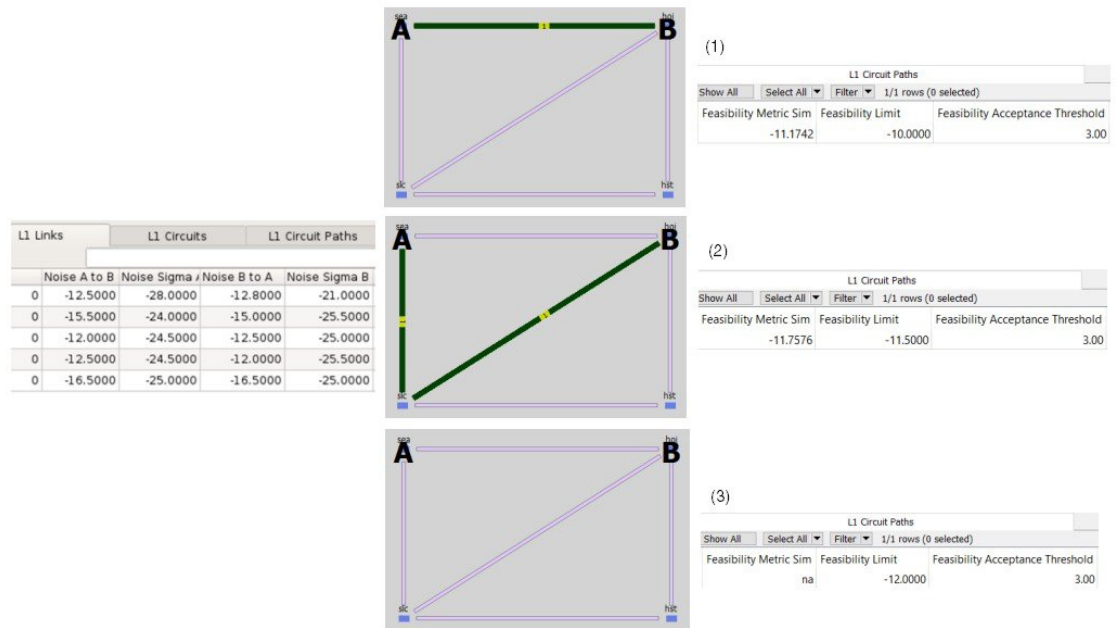
Feasibility properties affect routing as follows:

- If there is no Feasibility Limit, the shortest path is based on the L1 link Metric property, and Feasible Transmission is set to “na.”
- The shortest L1 circuit path is based on the L1 links that result in a L1 circuit path’s Feasibility Metric Sim value that is lower than the L1 circuit path’s Feasibility Limit. The Feasible Transmission of the L1 circuit path is set to T (true).
- If the L1 circuit path Feasibility Metric Sim is higher than the Feasibility Limit, it is not routed, and the Feasible Transmission is set to F (false) except when actual L1 circuit path hops exist.

Example: [Figure 77: Example Effects of L1 Link Feasibility on Routing L1 Circuit Paths, on page 203](#) shows a series of changes to the Feasibility Limit on an L1 circuit path. The L1 Links table—which shows the amount of noise, in decibels—does not change in any of the following scenarios.

- Row 1 shows that the Feasibility Limit of the L1 circuit path is set to –10 dB. Because the simulated feasibility metric (Feasibility Metric Sim) is lower than the Feasibility Limit, the L1 circuit path follows the shortest path, from sea to boi.
- Row 2 shows that the Feasibility Limit of the L1 circuit path decreases to –11.5 dB, and the path follows a longer route (sea-slc-boi) to stay under the limit.
- Row 3 shows that the Feasibility Limit of the L1 circuit path decreases to –12 dB, thus making it impossible for the path to route. The Feasibility Metric Sim changes to “na.”

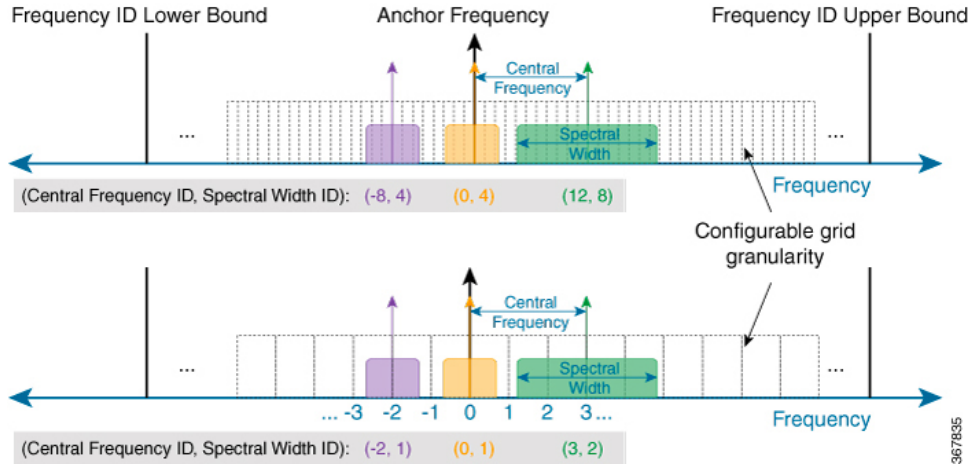
Figure 77: Example Effects of L1 Link Feasibility on Routing L1 Circuit Paths



Spectrum Modeling

L1 Circuits can be assigned spectrum of variable sizes.

Each Spectrum Grid is identified by Anchor Frequency, Central frequency granularity, and Central Frequency ID Blocked list. The Frequency ID Lower Bound and Frequency ID Upper Bound represent the lower and upper bounds on available central frequency IDs.



To define the spectrum grids, choose **Edit > Network Options**. On the Edit Network Options dialog box, click **Advanced** tab and update the values under Spectrum Grid section.

Central Frequency ID Assignments

Because multiple L1 Circuit Paths are transmitted in the same fiber, it is required that they each occupy a different part of the spectrum as they cross an L1 link. The portion of the spectrum that an L1 Circuit Path occupies is determined by its Central Frequency ID and Spectral Width ID. The Central Frequency ID identifies the central frequency of the L1 circuit path in terms of its distance from the Anchor Frequency of the defined grid. The Spectral Width ID defines the size of the L1 Circuit Path around this central frequency. This is expressed in integer increments corresponding to double the Central Frequency Granularity.

In WAE Design, the Central Frequency ID property identifies the L1 circuit path wavelength. To simulate changes in the Central Frequency ID values (regenerators) along an L1 circuit path, you can manually set Central Frequency ID values for each L1 circuit path hop.

- Each L1 circuit path traversing an L1 link must have a unique Central Frequency ID while on that L1 link. If more than one L1 circuit path has the same Central Frequency ID as they cross the L1 link, *Spectrum blocking* occurs. That is, the L1 circuit path with the highest Setup Priority property (lowest number) is routed, and the others are blocked.

To determine which L1 links are causing the spectrum blocking, right-click any L1 circuit path whose Spectrum Blocking value is T (True), and choose **Filter to Spectrum Blocking L1 Links**.

- Each L1 circuit path must use the same Central Frequency ID value on all L1 links that it traverses unless Central Frequency ID values are set on the L1 circuit path hops. This requirement is called the *spectrum continuity constraint*.

Feasibility Property	Description
L1 Circuit	

Feasibility Property	Description
Central Frequency ID Sim	<p>Simulated ID of the initial central frequency, which is copied from the Central Frequency ID Sim of the active L1 circuit path.</p> <p>This value is set to “na” if no L1 circuit path can be routed.</p>
Spectrum Blocking	<p>True (T) means the L1 circuit cannot be routed because there is spectrum blocking on all of its L1 circuit paths.</p>
Prefer Lower Frequency IDs	<p>Specifies if the spectrum assignment algorithm prefers lower Frequency IDs. The default is True (T).</p>
L1 Circuit Path	
Central Frequency ID	<p>The ID of the initial central frequency of the L1 Circuit Path..</p> <ul style="list-style-type: none"> • If there is no Central Frequency ID value entered, the central frequency ID value of the most recently used L1 circuit path hop is copied here. • If there are no L1 circuit path hops or if there are no Central Frequency ID assigned to L1 circuit path hops, this value is set to “na.”
Spectral Width ID	<p>The spectral width of the L1 Circuit Path in integer increments corresponding to double the Central Frequency Granularity.</p> <p>This value is set to 1 if it is not explicitly set.</p>
Central Frequency ID Sim	<p>Simulated ID of the initial central frequency , which is copied from one of these values in this order of priority:</p> <ul style="list-style-type: none"> • Central Frequency ID Sim value of the first actual L1 circuit path hop. • Central Frequency ID Sim value of the first L1 circuit path hop. • Central Frequency ID value of the L1 circuit path. <p>If Auto Spectrum Assignment is invoked, WAE Design changes the Central Frequency ID Sim in order to route the L1 circuit path.</p> <p>This value is set to “na” if the L1 circuit path cannot be routed.</p>
Central Frequency Sim	<p>The simulated initial central wavelength of the L1 Circuit Path.</p>
Spectral Width	<p>The spectral width of the L1 Circuit Path in GHz.</p>
Spectrum Blocking	<p>True (T) means this L1 circuit path cannot be routed either because it has the same central frequency ID value as another L1 circuit path going over an L1 link.</p>
Auto Spectrum Assigment	<p>If spectrum blocking occurs and if Auto Spectrum Assigment is T (true), the L1 circuit path tries to use another central frequency if it is available.</p> <p>This property is useful for simulating optical restoration when there are L1 failures.</p>
L1 Circuit Path Hop	

Feasibility Property	Description
Central Frequency ID	<p>The user-specified wavelength of the L1 circuit hop.</p> <p>The central frequency is used for all of the subsequent hops if no other change is specified.</p> <p>This value is set to “na” if it is not explicitly set.</p>
Central Frequency ID Sim	<p>Simulated wavelength number, which is copied from one of these values in this order of priority:</p> <ul style="list-style-type: none"> • The hop’s Central Frequency ID value. • The Central Frequency ID of the first preceding L1 circuit path hop that has a Central Frequency ID value. • The Central Frequency ID property of its L1 circuit path. <p>When you manually create L1 circuit path hops, WAE Design automatically creates the first L1 circuit path hop to be the first L1 node in the path. It also copies the L1 circuit path’s Central Frequency ID Sim value to this hop.</p> <p>If Auto Spectrum Assignment is used, this value is set to the Central Frequency ID Sim value of the L1 circuit path.</p> <p>This value is set to “na” if no Central Frequency ID values have been specified for this L1 circuit path hop or if the L1 circuit path cannot be routed.</p>

Routing Based on Central Frequency IDs

Central Frequency ID properties affect L1 circuit path routing as follows:

- If the Central Frequency ID is not set and cannot be derived, or if there is no spectrum blocking, the L1 circuit path is routed.
- If there is spectrum blocking, WAE Design checks the Auto Spectrum Assignment property for each L1 circuit path.
 - If F (false), the L1 circuit path is not routed.
 - If T (true), WAE Design tries to route the L1 circuit path on the first central frequency that is continuously available on all L1 links over which it is routed. If no such central frequency is found, the L1 circuit path is not routed.
- Spectral Width ID properties affect L1 Circuit path routing by defining how much spectrum must be available around the Central Frequency.

Note that the Export Spectrum Utilization tool lets you create and export a file that identifies all of the L1 circuit Central Frequency ID Sim values being used on each L1 link. Choose **File > Export > Spectrum Utilization**.

Example: [Figure 78: Example Spectrum Blocking, on page 207](#) shows two L1 circuit paths attempting to route over a common L1 link (chi-nyc).

- Path Option 1 uses sea-boi-chi-nyc.
- Path Option 2 uses sea-slc-hst-okc-chi-nyc.

Both have a Central Frequency ID value of 7. Both have Auto Spectrum Assignment set to F (false). Path Option 1 can route due to its lower Setup Priority value. Path Option 2 cannot route due to Spectrum Blocking being T (true). Notice that the Established column is F (false) and the Routed column is Unrouted.

Example: [Figure 79: Example Auto Spectrum Assignment, on page 207](#) shows that Path Option 2 can route after its Auto Spectrum Assignment property is set to T (true). As a result, its Central Frequency ID Sim automatically changes to an unused Central Frequency ID value so that there is no spectrum blocking. The Spectrum Blocking column changes to F (false), the Established to T (true), and the Routed to Simulated. The reason Path Option 1 can still route, even though its Auto Spectrum Assignment is still set to F, is because it has a lower Setup Priority value. Thus, WAE Design uses Central Frequency ID 7 for this path.

Figure 78: Example Spectrum Blocking

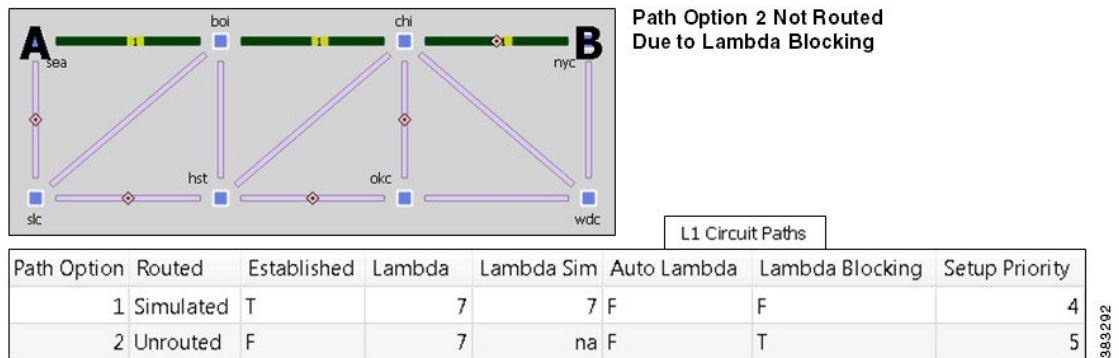
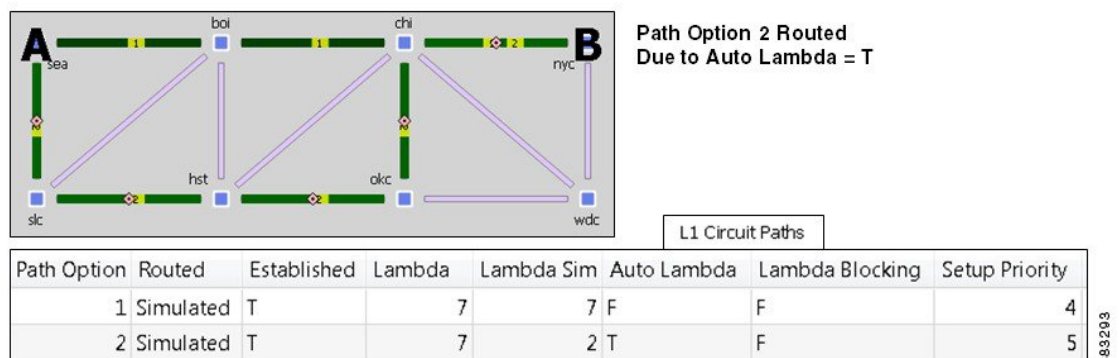


Figure 79: Example Auto Spectrum Assignment



Central Frequency ID Blocked List

Central Frequency Blocked list indicates the list of frequency IDs that may not serve as Central Frequency IDs for L1 Circuit Paths. Central Frequency ID Blocked lists may be set globally for the entire plan file or locally for each L1 Link. Local settings override global settings.

To set a global blocked list choose **Edit > Network Options**. On the Edit Network Options dialog box, click **Advanced** tab and select a list from the **Central Frequency ID Blacklist** dropdown.

To add a blocked frequency list for an L1 link, right-click one or more L1 links and choose **Properties**. To create a new blocked list, see [Creating Central Frequency ID Blocked List, on page 229](#).

L1 Diversity Constraints

L1 Circuits can be configured to be disjoint from other L1 Circuits. This helps in speedy recovery when network failures occur.

WAE Design only supports mutual diversity between two L1 circuits. This means, each L1 circuit can only have one diversity constraint defined. For example, L1 Circuit A and L1 Circuit B can be specified to be diverse from each other. But WAE design does not support setting of L1 Circuit A to be diverse from L1 Circuit B and L1 Circuit B diverse from L1 Circuit C.

You can specify different levels of diversity. L1 Circuit Path can be set to be diverse in terms of L1 Links, L1 Nodes, and SRLGs.

The diversity constraints apply to all L1 Circuit Paths of L1 Circuits. For example, if L1 Circuit A and L1 Circuit B are specified to be diverse from each other with each L1 Circuit having two L1 Circuit Paths, then both L1 Circuit Paths of Circuit A will be diverse from both Circuit Paths of Circuit B.



Note If a fully disjoint path is not found, the L1 Circuit Path is not routed.

Relationships Between L1 and L3

Relationships between L1 and L3 objects can be created in multiple ways. [Figure 80: Configurable L1 and L3 Relationships, on page 209](#) shows the relationships that can be directly configured.

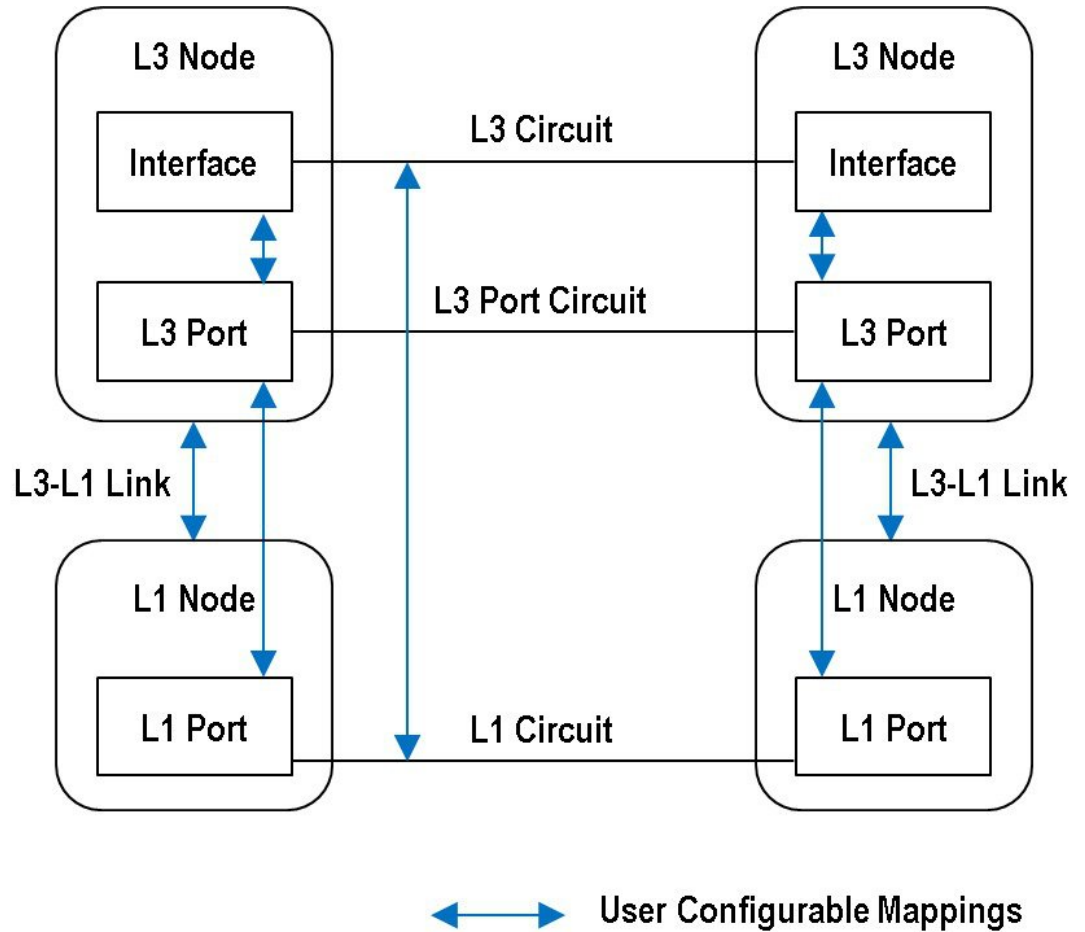
- You can create L3-L1 links to model the physical relationship between L3 nodes and L1 nodes as shown in [Figure 74: L1-L3 Link Visualization, on page 193](#).
- You can directly map L1 ports to L3 ports.
- You can directly map L1 circuits to L3 circuits, or these associations can be formed indirectly. Once associated with L1 circuits, L3 circuits inherit properties, such as Delay and Distance, from L1 circuits. Additionally, these associations let you see the effect that failed and inactive L1 links, L1 nodes, and L1 ports have on the L3 network.

The basis of indirect L1-L3 relationships is the fact that every L1 circuit must have two L1 ports and that you can optionally map these L1 ports to L3 ports.

- When L1 ports are mapped to L3 ports that are associated with L3 circuits, L1 circuits become indirectly associated with those same L3 circuits.
- Similarly, when L1 ports are mapped to L3 ports that are associated with L3 port circuits, L1 circuits become indirectly associated with those same L3 port circuits.
- Two L3 ports are implicitly connected (1) if they are connected to two interfaces on the same L3 circuit and (2) if they are mapped to L1 ports that are connected to an L1 circuit.

Note that you can enhance the visualization of L1 and L3 relationships by viewing both layers simultaneously.

Figure 80: Configurable L1 and L3 Relationships



407030

For More Information...	See...
Directly map L1 circuits to L3 circuits	Mapping an L1 Circuit to an L3 Circuit, on page 234
Directly map L1 ports to L3 ports	Mapping an L1 Circuit to an L3 Port Circuit, on page 235
Create L3-L1 links	Creating L3-L1 Links Between Nodes, on page 230
View L1 and L3 layers simultaneously	Multi-Layer Visualization, on page 236

Example Relations Between L1 Link and L3 Circuit

This example demonstrates the dynamics between an L1 link and L3 circuits. [Figure 81: Relationship Between L1 Links and L1 Circuits, on page 211](#) shows an L1 link named key-slc that is filtered to its associated L1 circuits. Each L1 circuit path has one L1 circuit path (labeled “1” on the path) where the end L1 nodes are labeled A and B. [Figure 82: Relationship Between L1 Circuits and L3 Circuits, on page 212](#) shows the association of these L1 circuits with L3 circuits.

Example Relations Between L1 Link and L3 Circuit

This L1 Link...	Is Used By These L1 Circuits...	Which Are Mapped to These L3 Circuits
key-slc	chi-sea	cr1.chi-cr2.sea
	chi-sjc	cr2.chi-cr1.sjc
	kcy-sjc	cr1.sjc-cr1.kcy

Figure 81: Relationship Between L1 Links and L1 Circuits

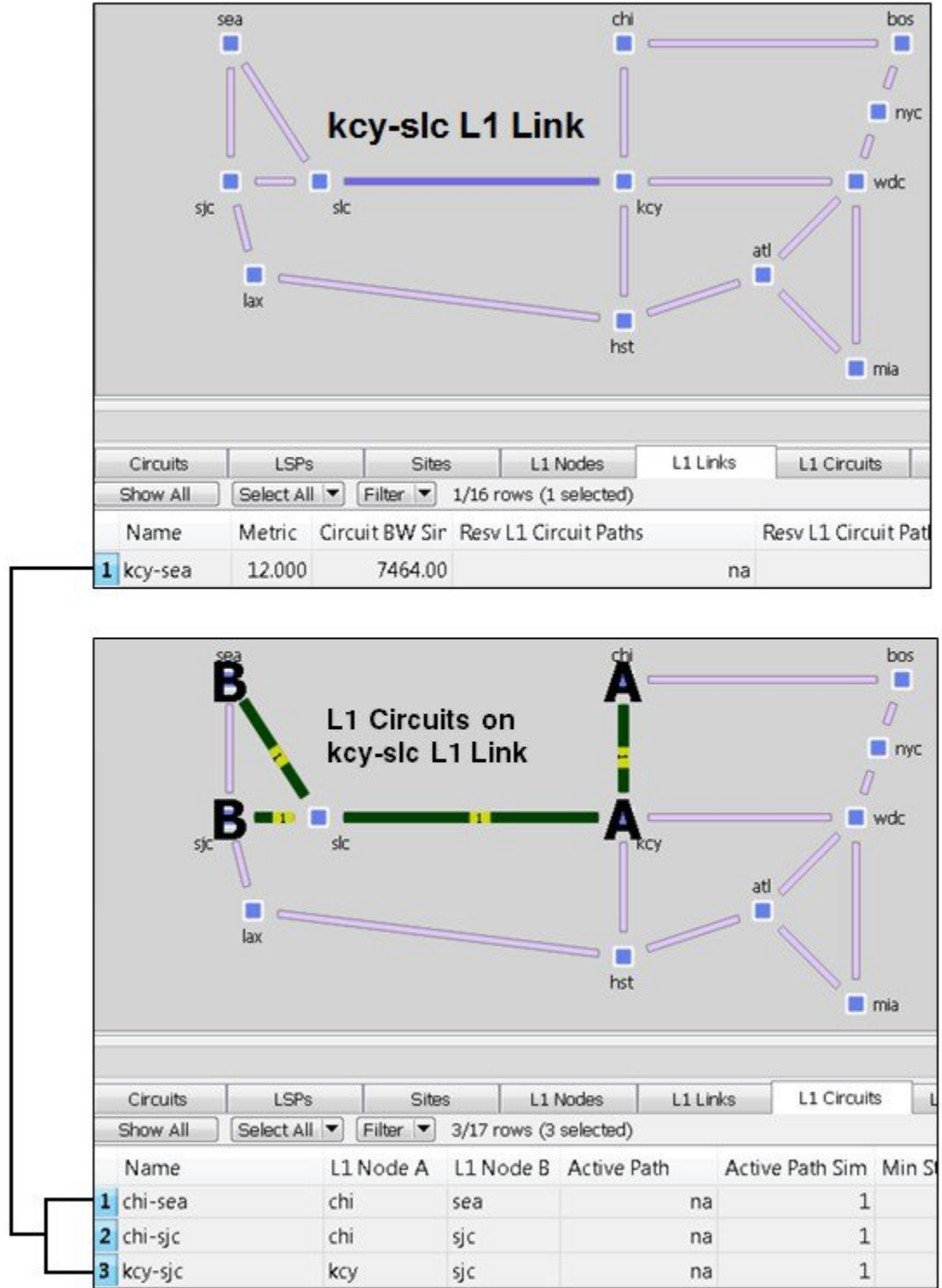
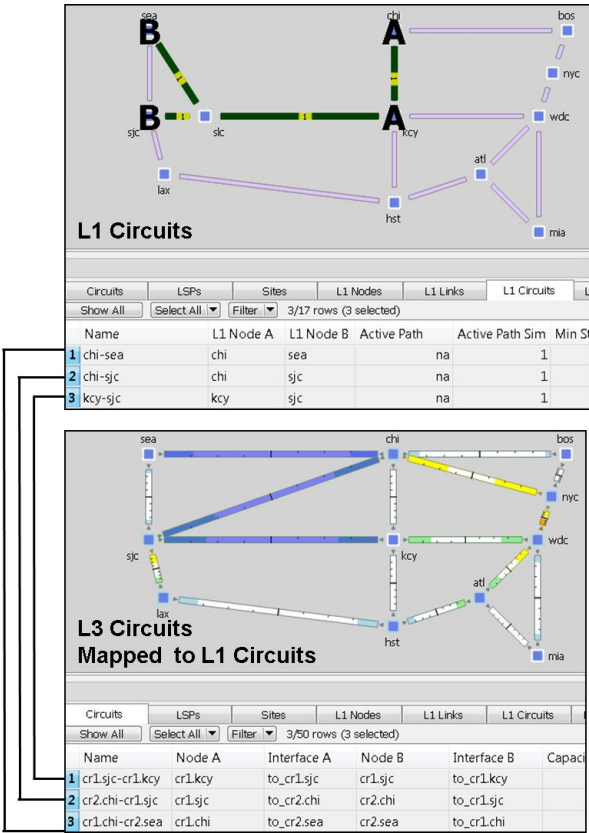
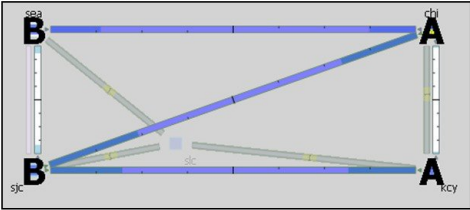


Figure 82: Relationship Between L1 Circuits and L3 Circuits



L3 View Showing the Same Mapped L3 Circuits with L1 Circuits Selected in the Background



380016

Delay and Distance

Both Delay and Distance are properties that can be set in the L1 link and L1 circuit path Properties dialog boxes. To automate the process of setting these, use the Latency and Distance Initializer. For more information on simulated Delay and Distance, including the Latency and Distance Initializer, see [Simulation, on page 69](#).

L1 Link

Distance:	1119.38
Delay:	5.57

L1 Circuit Path

Delay:	7.5	Distance:	2000
--------	-----	-----------	------

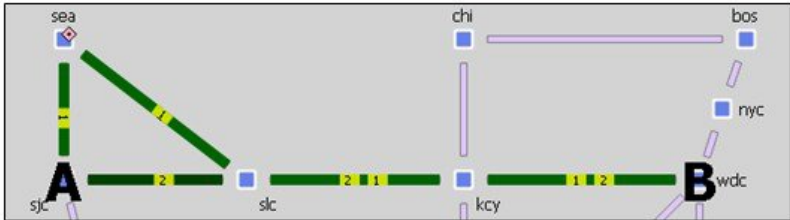
38 09 41

Column	Description
L1 Links	
Delay	One-way transmission latency over the L1 link in milliseconds (ms).
Distance	Distance of the L1 link in kilometers (km).
L1 Circuits	
Delay Sim	<ul style="list-style-type: none"> • (Derived) This valued is copied from the L1 circuit path Delay Sim column of the active path. • If no L3 Delay is explicitly set for the L3 circuit, the Delay Sim column in the (L3) Circuits table is copied from the Delay Sim column of the mapped L1 circuit, which is based on the L1 circuit path being used as the active path.
Distance Sim	<ul style="list-style-type: none"> • (Derived) This valued is copied from the L1 circuit path Distance Sim column of the active path. • If no L3 Distance is explicitly set for the L3 circuit, the Distance Sim column in the (L3) Circuits table is copied from the Distance Sim column of the mapped L1 circuit, which is based on the L1 circuit path being used as the active path.
L1 Circuit Paths	
Delay	One-way transmission latency over the L1 circuit path in milliseconds (ms).
Delay Sim	(Derived) If the L1 circuit path Delay value is entered, it is copied to the Delay Sim column. Otherwise, this is set to the sum of the delays of the L1 links over which this L1 circuit path is routed.
Distance	Distance of the L1 circuit path in kilometers (km).
Distance Sim	(Derived) If the L1 circuit path Distance value is entered, it is copied to the Distance Sim column. Otherwise, this is set to the sum of the distances of the L1 links over which this L1 circuit path is routed.

Example: [Figure 83: Example Effect of L1 Link Delay and Distance, on page 214](#) shows how L1 link Delay and Distance properties affect L1 circuit path properties (and thus, L1 circuit properties), and how they can potentially affect L3 circuit properties.

- One L1 circuit, which is mapped to an L3 circuit, has two L1 circuit paths.
- Because the active L1 circuit path (Path Option 2) has neither a Delay, nor a Distance set, its Delay Sim and Distance Sim columns are derived by adding together the Distance and Delay values of the L1 links over which it traverses. The combined Delay of these three L1 links is 6.880 ms, and the combined Distance is 900 km. The L1 circuit inherits these in its Distance Sim and Delay Sim columns.
- Because the L3 circuit mapped to the L1 circuit does not have a Delay or Distance property, its Delay Sim and Distance Sim values are copied from the L1 circuit Delay Sim and Distance Sim columns.
- Note that if the L1 circuit's active path were to change to 1 (Path Option 1), the L1 circuit and its mapped L3 circuit would have a Distance Sim of 1400 km and a Delay Sim of 10.880.

Figure 83: Example Effect of L1 Link Delay and Distance



L1 Links				
Name	L1 Node A	L1 Node B	Distance	Delay
L1link-sea-sjc	sea	sjc	200.000	2.00
L1link-sea-slc	sea	slc	400.000	3.00
L1link-kcy-wdc	kcy	wdc	300.000	1.00
L1link-sjc-slc	sjc	slc	100.000	1.00
L1link-kcy-slc	kcy	slc	500.000	4.88

L1 Circuit Paths							
L1 Circuit	L1 Node A	L1 Node B	Path Option	Distance	Distance Sim	Delay	Delay Sim
sjc-wdc	sjc	wdc	1	na	1400.000	na	10.880
sjc-wdc	sjc	wdc	2	na	900.000	na	6.880

L1 Circuits					
Name	L1 Node A	L1 Node B	Active Path Sim	Distance Sim	Delay Sim
sjc-wdc	sjc	wdc	2	900.000	6.880

Circuits								
Name	Node A	Interface A	Node B	Interface B	Delay	Delay Sim	Distance	Distance Sim
cr1.sjc-cr1.wdc	cr1.sjc	to_cr1.wdc	cr1.wdc	to_cr1.sjc	na	6.88	na	900.00

380904

L1 Failures

L1 links, L1 nodes, and L1 ports can be set to failed states similarly to L3 circuits and nodes in the L3 topology.



Note For detailed information about visual representations of failure, inactive, and non-operational states, see [Plot Legend for Design Layouts, on page 399](#). For information on how to fail objects and recover them from failures, and how to set objects to active or inactive state, see [Simulation, on page 69](#).

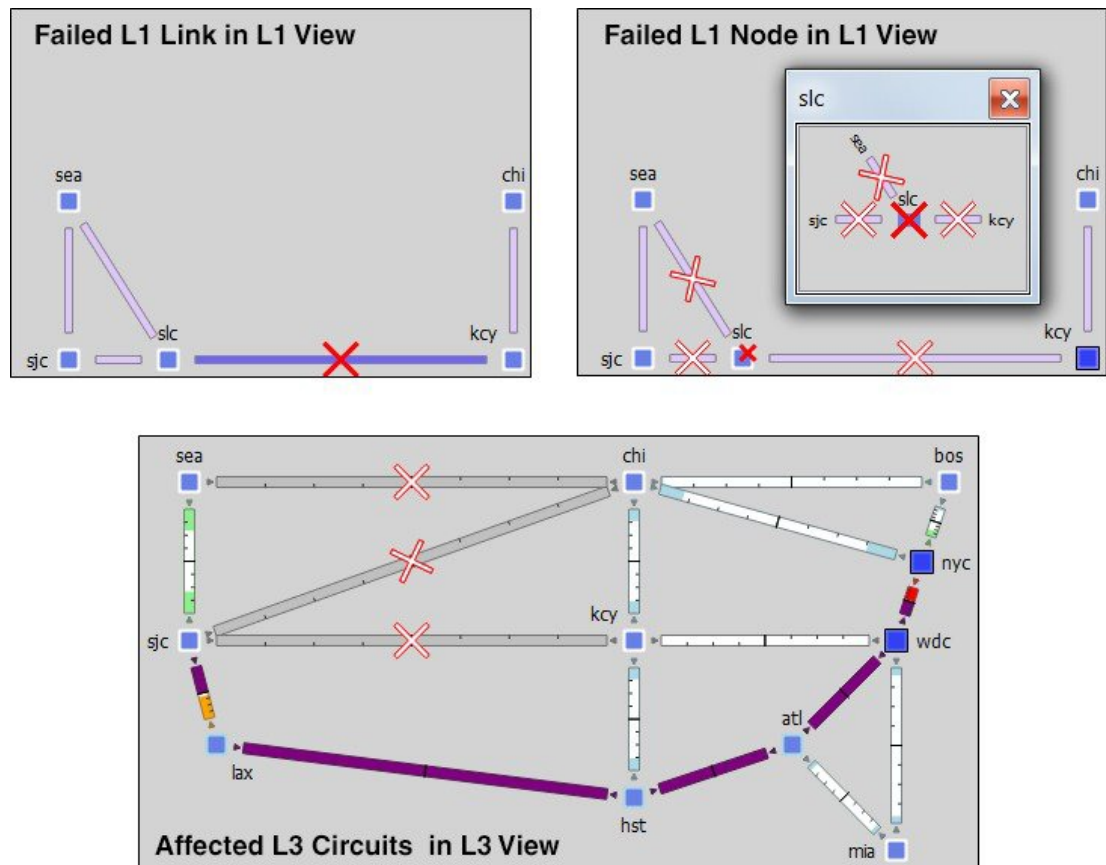
L1 Failure Effects on L3

L3 circuits are rerouted due to L1 failures if the L3 circuit is mapped to an L1 circuit that would traverse a failed L1 node or L1 link under normal operations.

Example: Drawing upon the example L1 link, L1 circuits, and L3 circuits relationships shown in [Figure 81: Relationship Between L1 Links and L1 Circuits, on page 211](#) and [Figure 82: Relationship Between L1 Circuits and L3 Circuits, on page 212](#) [Figure 84: Example Effect of Failing an L1 Link or L1 Node on L3 Circuits, on page 215](#), shows the effects of failing the key-slc L1 link. It also shows the effect of an slc L1 node failure. If an L1 node fails, the connected L1 links are non-operational. In this example, both the L1 link and L1 node failure have the effect of making the same L3 circuits non-operational.

Example: Drawing upon the example in [Figure 84: Example Effect of Failing an L1 Link or L1 Node on L3 Circuits, on page 215](#), [Figure 85: Example Effect of Rerouting L1 Circuit Paths, on page 216](#) demonstrates what occurs if the key-sjc L1 circuit path is rerouted. The L1 circuit path takes a southern route, and the L3 circuit continues to route traffic. The L3 view continues to show active routing, and by selecting the mapped L3 circuit, you can switch to the L1 view to see both the original (solid green) and rerouted, rerouted L1 path (dashed lined). Alternatively, you could see the impact of the L1 failure directly in the L3 view by showing L1 objects in the background ([Figure 86: Multi-Layer View of Rerouting L1 Circuit Paths, on page 216](#)). Note that the L3 circuit Delay Sim value (in the Circuits table) might change due to the rerouting of the L1 circuit path.

Figure 84: Example Effect of Failing an L1 Link or L1 Node on L3 Circuits



380908

Figure 85: Example Effect of Rerouting L1 Circuit Paths

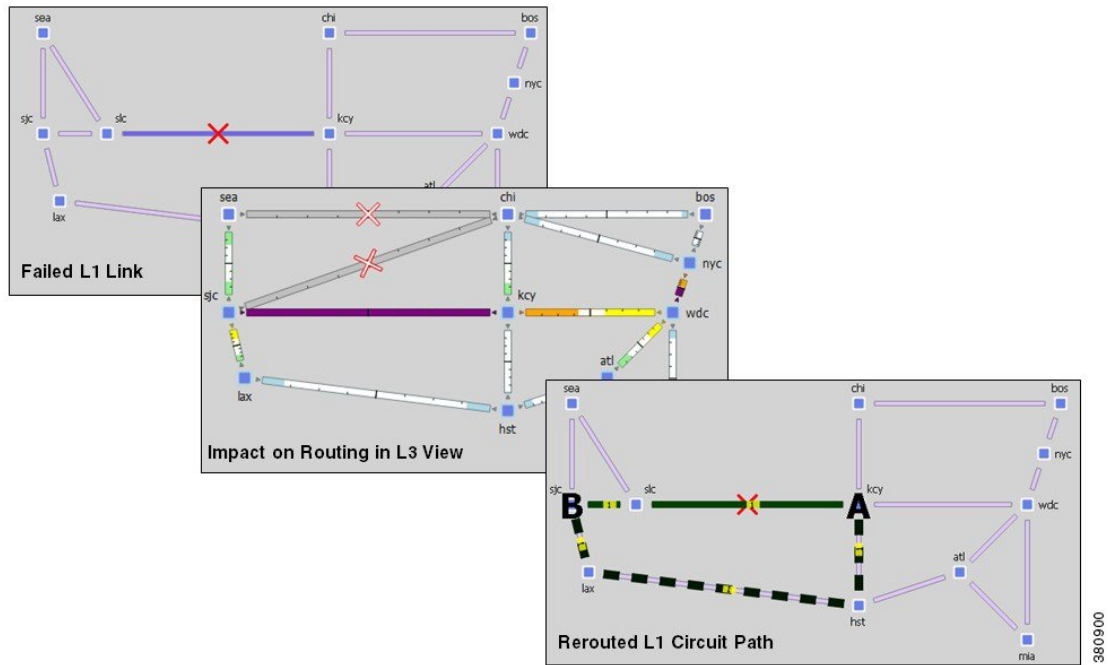
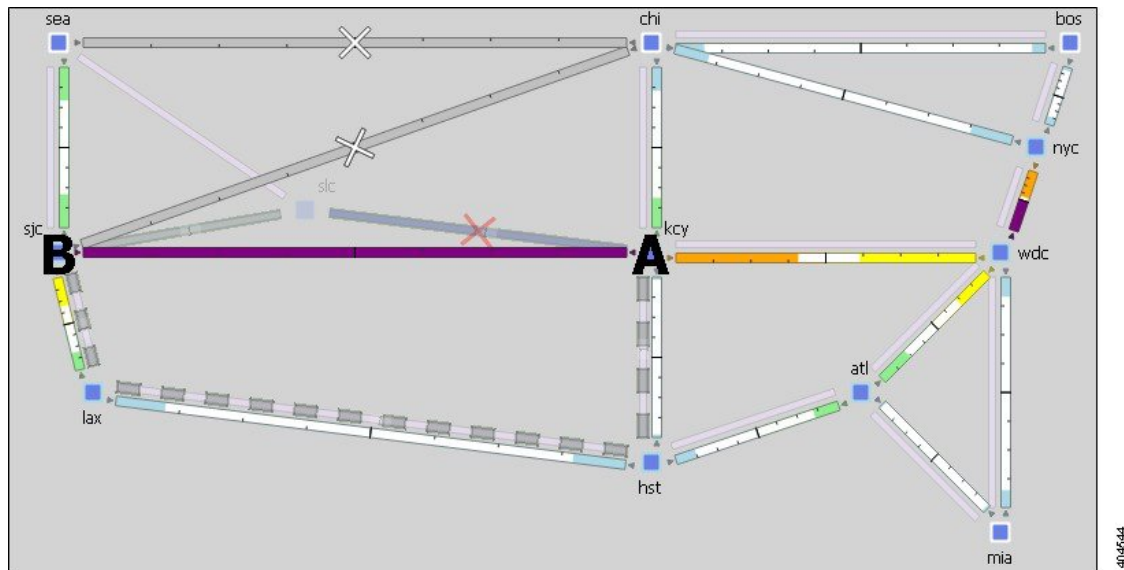


Figure 86: Multi-Layer View of Rerouting L1 Circuit Paths



Resiliency Under L1 Failures

L1 circuit paths can be rerouted around failures using either standby (static) or non-standby (dynamic) L1 circuit paths. If no restrictions are specified with respect to path hops, this reroute takes the best path available based on L1 link metrics.

- Standby L1 circuit paths are always established (available) to route. If there are multiple standby L1 circuit paths, WAE Design tries to route each available standby L1 circuit path based on their Path Option values (the lower the value, the higher the priority). In this way you can simulate optical circuit protection, such as might be configured for 1+1 protection (one primary and one backup path).
- The rerouting of non-standby L1 circuit paths under failure is determined by the “Layer 1 rerouting convergence under failures” check box.



If this check box is unchecked, non-standby L1 circuit paths do not try to route when failures occur. L3 convergence can be set to different modes, and each one can be combined with this L1 convergence mode. For example, when combined with setting the L3 simulation convergence mode to Fast Reroute, deselecting this L1 network option is useful for determining the impact on traffic before L1 and L3 restoration has occurred.

If this option is selected, non-standby L1 circuit paths are dynamically available. That is, they are available only if WAE Design tried to route all standby paths and if the number of routed standby paths is less than the L1 circuit’s Min Routed Paths property. Thereafter, WAE Design tries to route each non-standby L1 circuit path in the order of its Path Option value. This behavior models optical circuit restoration. When combined with an L3 simulation convergence mode of IP and LSP Convergence, you can model the impact failures have on traffic after both L3 and L1 have reconverged.

Regardless of this L1 network option setting, standby L1 circuit paths that are not affected by the failure continue to route. For details on how L1 circuit paths are routed, see [L1 Simulations, on page 195](#).

Example: This example assumes that the “Layer 1 rerouting convergence under failures” check box is checked. The L1C_chi-atl L1 circuit has three L1 circuit paths ([Figure 87: Example of Dynamic Non-Standby Path Routing Under Failure, on page 218](#)).

- Path Option 1 is standby, and has no hops.
- Path Option 2 is standby, and has a strict key hop, and an exclude hst hop.
- Path Option 3 is non-standby and has no hops.

The non-standby Path Option 3 is not routed until both standby paths are unroutable due to failure. Notice the Routed column changes for all three paths, indicating which paths are routable or simulated. Also notice that the Established column remains consistent. This is because whether or not an L1 circuit path is available for routing under normal operation has no bearing on whether it is routable under failure.

In this example, if the “Layer 1 rerouting convergence under failures” check box is unchecked, Path Option 3 does not try to reroute ([Figure 88: Example of Non-Standby Path Not Rerouting Under Failure, on page 218](#)).

Figure 87: Example of Dynamic Non-Standby Path Routing Under Failure

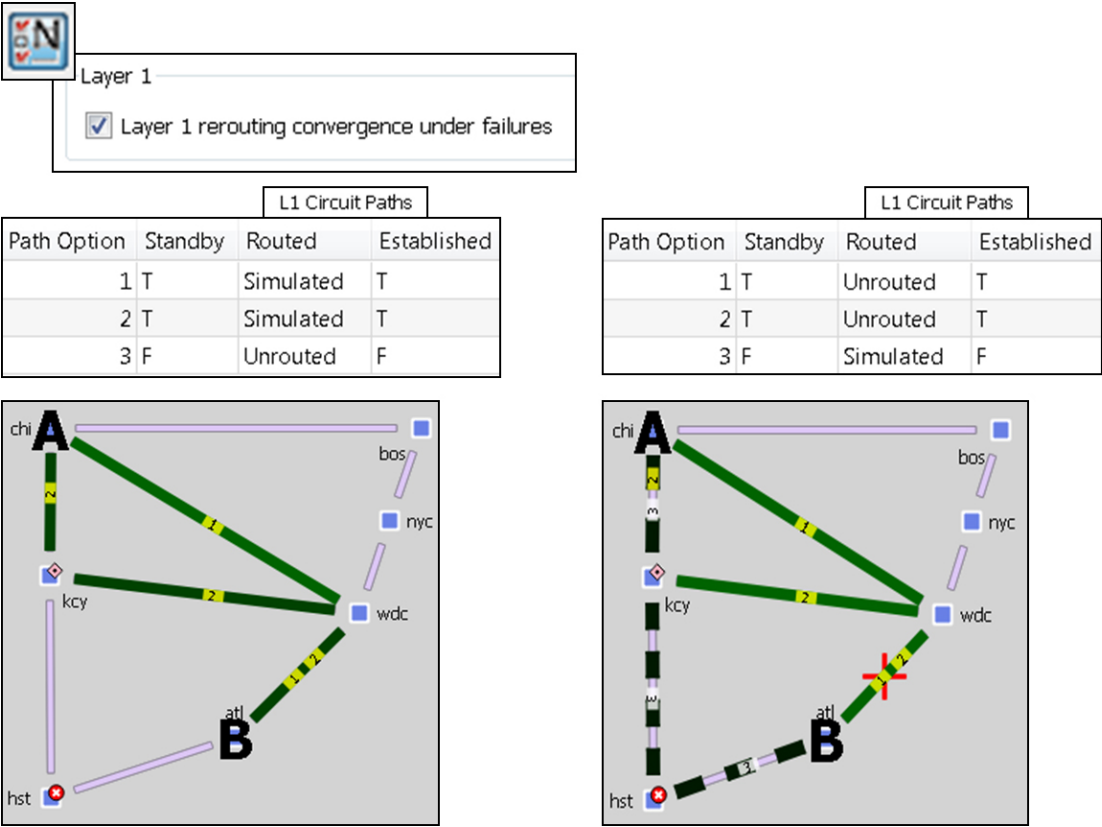
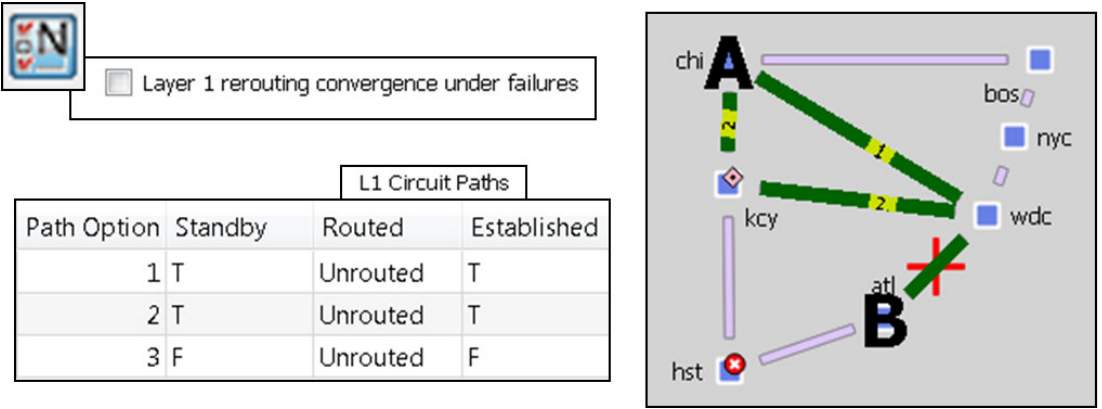


Figure 88: Example of Non-Standby Path Not Rerouting Under Failure

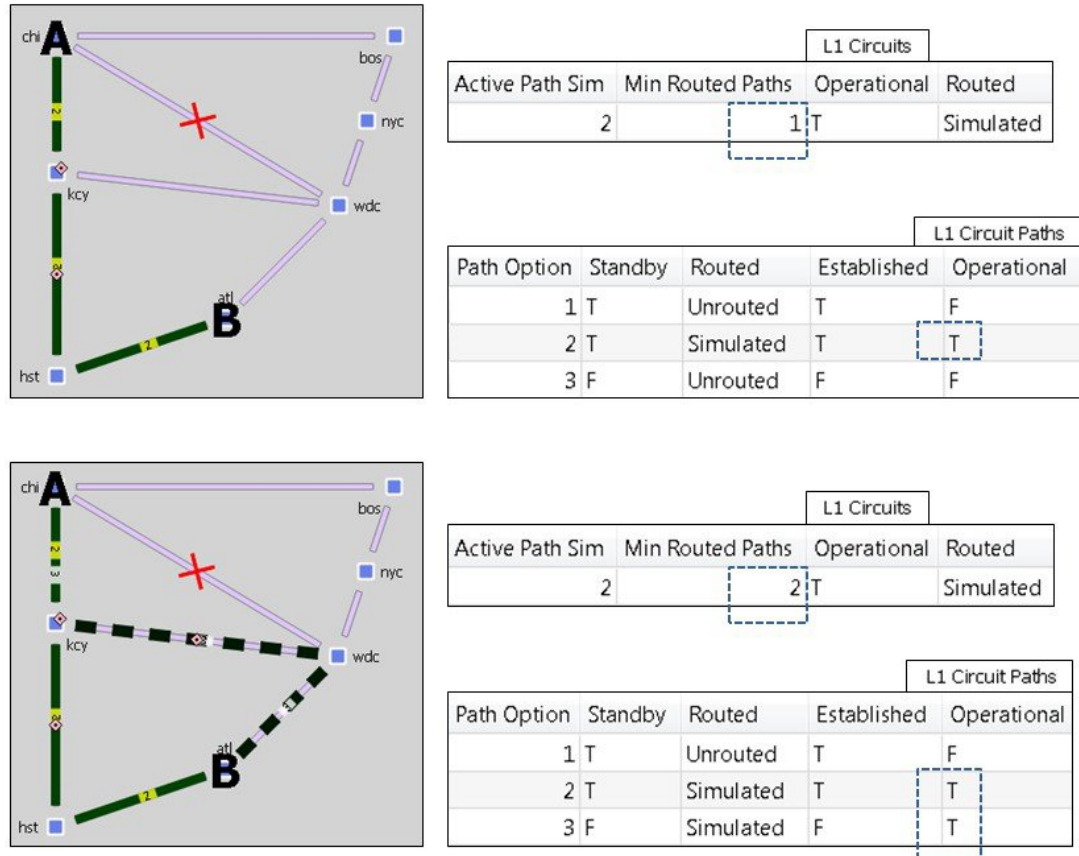


Minimum Routed Paths

The number of L1 circuit paths that WAE Design tries to route is defined by the L1 circuit’s Min Routed Paths property. If the number of routed standby paths is less than the Min Routed Paths property, WAE Design tries to route non-standby paths. This can be useful when modeling 1+1+R resilience schemes.

Example: [Figure 89: Example Resilience Scheme Using Minimum Routed Paths Property](#), on page 219 shows an L1 circuit with Min Routed Paths set to 1. It has three L1 circuit paths; two are standby and one is non-standby. When one standby L1 circuit path cannot be rerouted under failure, there is no attempt to route the non-standby path. By changing the Min Routed Path to 2, the non-standby path is routed.

Figure 89: Example Resilience Scheme Using Minimum Routed Paths Property



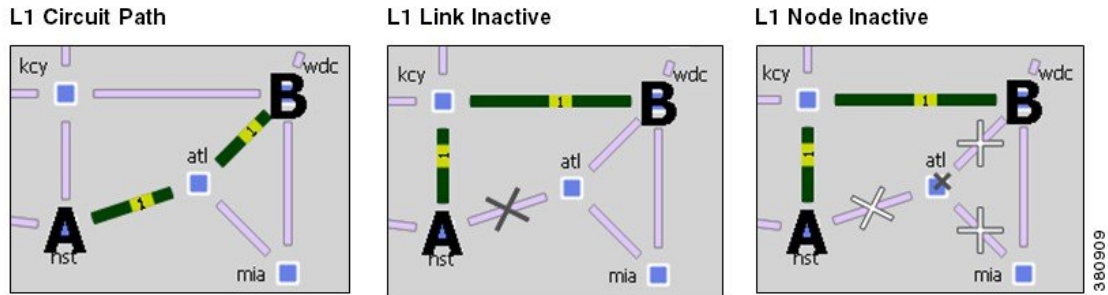
353653

Inactive State

L1 links, L1 nodes, and L1 ports can be set to inactive states similarly to L3 circuits and nodes in the L3 topology. Setting an L1 link or L1 node to inactive reroutes L1 circuit paths using that L1 link or L1 node. It has no effect on the operational state of L3 circuits unless the L1 circuit path cannot be rerouted, in which case the associated L3 circuit becomes operationally down. Note that the L3 circuit Delay Sim value (in the Circuits table) might change due to the rerouting of an L1 circuit path.

Example: [Figure 90: Example Reroute of L1 Circuit when L1 Link or L1 Node is Inactive](#), on page 220 shows a rerouted hst-wdc L1 circuit path when an L1 link that it traverses is set to inactive. The same L1 circuit path reroute occurs as a result of the atl L1 node being set to inactive.

Figure 90: Example Reroute of L1 Circuit when L1 Link or L1 Node is Inactive



L1 Simulation Analysis

L1 links and L1 nodes can be included in Simulation Analysis either as single objects or as part of SRLGs. L1 ports can be included as single objects.

Worst-Case Simulations

When included in the Simulation Analysis, their impact on L3 worst-case traffic utilization is reflected in the L3 Worst-Case Traffic view. Similarly, if an L1 link or L1 node is the cause of worst-case traffic utilization, it is reflected in the WC Failure column in the Interfaces and Circuits tables.

The L1 Links table updates the following columns when you include L1 links or L1 nodes in the Simulation Analysis:

- WC Num L1 Circuit Paths—Number of L1 circuit paths that traverse this L1 link under worst-case L1 failure simulations.
- WC Util Sim—Number of L1 circuit paths that traverse this L1 link under worst-case L1 failure simulations as a percentage of the maximum number of L1 circuit paths that can traverse this L1 link without failures.

Number of L1 circuit paths that traverse this L1 link under worst-case L1 failure simulations / Max L1 Circuit Paths (property value)

The following columns are updated in the L1 Nodes table:

- Num L1 Circuit Paths—Number of L1 circuit paths with paths through the L1 node.
- WC Failures—Failures that cause the worst-case number of L1 circuit paths.
- WC Num L1 Circuit Paths— Number of L1 circuit paths that traverse this L1 node under worst-case L1 failure simulations.

Failure Impact

The L1 Failure Impact view is available upon running a Simulation Analysis on L1 links. The L1 links are colored according to the maximum L3 interface traffic utilization level that would be caused should an L1 link fail. Similarly, L1 node border colors show the maximum L3 interface traffic utilization level that would

be caused should an L1 node fail. The color indicates the resulting L3 utilization and severity of the L3 congestion.

Example: In the L1 Failure Impact view, if the chi-bos L1 link shows a utilization of 80-90% and its color representation is orange, then if that chi-bos L1 link were to fail, one or more L3 interfaces would react by exceeding a 80% utilization level.



Note The network plot in the L1 Failure Impact view does not show utilizations unless L1 links or L1 nodes are included in Simulation Analysis.

If you are showing only an L1 view, site borders show the maximum utilization level that would be caused elsewhere in the network if L1 nodes within it fail or if intrasite L1 links within it fail. If you are showing a multi-layer (L1 and L3) view, site borders show the maximum utilization level that would be caused elsewhere in the network if a contained L3 node, L3 circuit, L1 node, or L1 link fails.

Example: [Figure 91: Example Site Borders in Failure Impact View, on page 222](#) is an example of how the site border of the sea site differs, depending on whether you are showing a Layer 3 view or a Layer 1 view. Notice that Layer 1 shows the higher failure impact utilization. Thus, when showing a multi-layer view, the border indicating the highest failure impact takes precedence.

The following columns in the L1 Links table show the impact of failing L1 links in Simulation Analysis:

- **Failure Impact**—Worst-case traffic utilization of L3 interfaces if this L1 link were to fail. For example, if the value is 70.13%, it means that if this L1 circuit failed, the resulting traffic utilization on one or more L3 interfaces would exceed 70.13%.
- **Failure Impact Interface**—The L3 interface that will experience the highest utilization as a result of the L1 link going down.

Format = if{Node|Interface}

Example: if{cr2.par|to_cr1.fra} means if the L1 link goes down, it will have the greatest traffic impact on the cr2.par to cr1.fra interface.

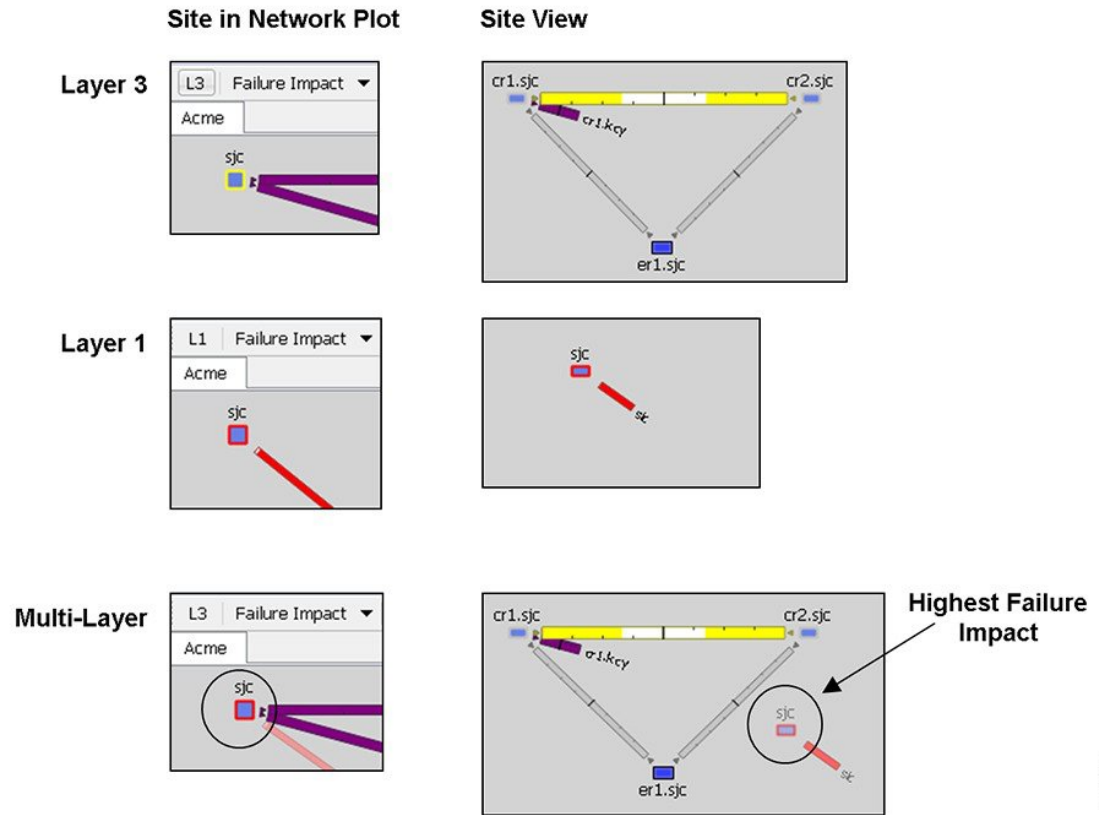
The following columns in the L1 Nodes table show the impact of failing L1 nodes in Simulation Analysis:

- **Failure Impact**—Worst-case traffic utilization of L3 interfaces if this L1 node were to fail. For example, if the value is 70.13%, it means that if this L1 node failed, the resulting traffic utilization on one or more L3 interfaces would exceed 70.13%.
- **Failure Impact Interface**—The L3 interface that will experience the highest utilization as a result of the L1 node going down.

Format = if{Node|Interface}

Example: if{cr1.chi|to_cr2.mia} means if the L1 node goes down, it will have the greatest traffic impact on the cr1.chir to cr2.mia interface.

Figure 91: Example Site Borders in Failure Impact View



407054

Worst-Case Failures

After running a Simulation Analysis, you can view the L1 failure scenarios that caused the worst-case utilization for an L3 interface. Right-click an L3 interface or L3 circuit and choose **Fail to WC**. A dialog box opens from which you can choose a failure scenario to observe. Click **OK**; the network plot changes to show the failure scenario. If you are in the L3 view and the worst-case failure is an L1 object, switch views to see the failure in the L1 view.

For information on running Simulation Analysis, failing L1 objects to their worst-case utilization, and protecting L1 objects from inclusion in Simulation Analysis, see [Simulation Analysis, on page 79](#).

Creating L1 Topology

Creating L1 Topology from L3 Topology

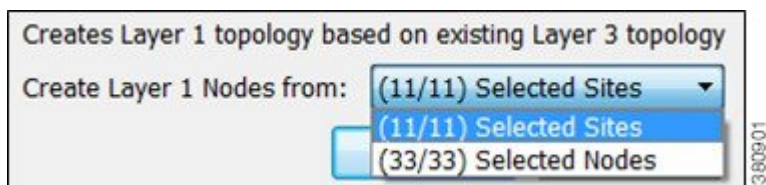
WAE Design provides a tool that constructs a complete L1 topology using an existing L3 topology as a model. You have an option to create the topology based on sites or L3 nodes. WAE Design then shows the same site in the L1 view and constructs the following L1 topology.

- Creates L1 nodes.

- Create L1 nodes from sites—WAE Design creates one L1 node within each selected site regardless of the number of L3 nodes within that site. The L1 nodes are assigned the same name as their sites. No L1 nodes are created for L3 nodes that are not residing in sites.
- Create L1 nodes from L3 nodes—WAE Design creates one L1 node for each L3 node selected. The L1 nodes are assigned the same name as the L3 nodes. If the L3 node resides within a site, the L1 node is placed there as well.
- Creates L1 links between all L1 nodes whose corresponding L3 nodes have connectivity between them. L1 link metrics are set to 1.
- Creates an L1 circuit and L1 circuit path between each pair of L1 nodes connected by L1 links. The L1 circuit Path Option value is set to 1.
- Creates a pair of L1 ports for each L1 circuit.
- Creates a list of blocked Central Frequency IDs.
- Creates the mapping between L1 circuits and L3 circuits. For more information on these mappings, see [Relationships Between L1 and L3](#), on page 208.
- Creates one standby L1 circuit path per each L1 circuit, and sets its Path Option to 1.

Follow these steps to create an L1 topology from an L3 topology. You can be in either an L1 or L3 view while performing these steps.

- Step 1** If you want to create an L1 topology based on only specific L3 objects, select those sites or L3 nodes from the L3 view. If you do not select sites or nodes, the topology is based on all sites or all L3 nodes, depending on your selection in Step 3.



- Step 2** Choose **Insert > Layer 1 > L1 Topology from L3 Topology**, or right-click the plot and choose **New > Layer 1 > L1 Topology from L3 Topology**.
- Step 3** Select either sites or nodes, and click **OK**.

Creating L1 Nodes

- Step 1** Choose **Insert > Layer 1 > L1 Node**, or right-click the plot and choose **New > Layer 1 > L1 Node**.
- Step 2** Enter a unique L1 node name.
- Step 3** Choose the site to which you want this L1 node to belong, or choose **None** if you do not want the L1 node to reside in a site.
- Step 4** (Optional) Choose the node type from the drop-down list.

- Step 5** (Optional) Enter a description.
- Step 6** (Optional) Click the **Identification** tab to enter more descriptive information.
- Step 7** Click **OK**.

Creating L1 Links

Each L1 link name and L1 node combination must be unique.

- Step 1** Choose **Insert > Layer 1 > L1 Link**, or right-click the plot and choose **New > Layer 1 > L1 Link**.
- Step 2** Enter an L1 link name and description. If the L1 link has the same source and destination as another L1 link, it must have a unique name. Otherwise, it is not required.
- Step 3** Required: Choose the two nodes you want to connect with this L1 link by first choosing the site and then choosing the node.
- Step 4** Optional selections include:

Field	For Information...
Metric	L1 Link Metrics, on page 200
Distance, Delay	Delay and Distance, on page 212
Feasibility	L1 Circuit Wavelengths, on page 201
Frequency ID Blocked List	Central Frequency ID Blocked List, on page 207
Protected, Active, Failed	Simulation, on page 69 and Simulation Analysis, on page 79

- Step 5** Check **SSON Enabled** to specify that the L1 Link is SSON enabled.
- Step 6** Click **OK**.

Creating L1 Circuits

You can manually create circuits by using the following method. Additionally, you can automatically create L1 circuits using their respective Properties dialog boxes.

Each L1 circuit name and L1 node combination must be unique.

- Step 1** Choose **Insert > Layer 1 > L1 Circuit**, or right-click the plot and choose **New > Layer 1 > L1 Circuit**.
- Step 2** Enter an L1 circuit name.
- Step 3** Choose the two L1 ports you want to connect with this L1 circuit by first choosing the sites (if applicable), the L1 nodes, and then L1 ports.

Name: (L1Circuit[*])	
L1 Port A	
Site:	chi
L1 Node:	chi
L1 Port:	chi-bos
L1 Port B	
Site:	bos
L1 Node:	bos
L1 Port:	bos-chi

Step 4 Optional selections are as follows:

Field	For Information...
Disjoint Group, Disjoint Priority	Explicit L1 Circuit Paths Initializer, on page 199
Bandwidth	Enter a value for the bandwidth that the L1 circuit requires. This value is not used for routing. It is copied to the BW Sim column in the L1 Circuits table. If no value is entered, the BW Sim column is set to the simulated capacity of the L3 circuit that is mapped to this L1 circuit. The total BW Sim for all L1 circuits routing over an L1 link is shown in the Circuit BW Sim column of the L1 Links table.
Active Path	Active L1 Path Selection, on page 196
Min Routed Paths	Minimum Routed Paths, on page 218

Step 5 Check the **Prefer Lower Frequency IDs** based on your requirements.

Step 6 Check **SSON Enabled** to specify that the L1 Circuit is SSON enabled.

Step 7 Click **OK**.

Plotting L1 Circuits

Step 1 In the L1 Circuits table, choose one or more L1 circuits.

Step 2 Right-click and choose **Plot L1 Circuits**.

To move the plot horizontally or vertically, use the left/right arrows and up/down arrows.

To select all L1 circuits within the plot, click **Select L1 Circuits**. To deselect all of these L1 circuits, click in an empty plot area.

Creating L1 Circuit Paths

Step 1 Choose one or more L1 circuits. If you do not choose any, L1 circuit paths are created for all of the L1 circuits.

- Step 2** Access an L1 circuit path in one of these ways:
- Choose **Insert > Layer 1 > L1 Circuit Paths**.
 - Right-click an L1 circuit in the L1 Circuits table and choose **Insert L1 Circuit paths**.
 - Right-click the plot and choose **New > Layer 1 > L1 Circuit Paths**.
- Step 3** Select or enter a path option.
- Step 4** If the L1 circuit path is a standby path, check **Standby**.
- Step 5** Click **OK**.

Plotting L1 Circuit Paths

- Step 1** In the L1 Circuit Paths table, choose one or more L1 circuit paths.
- Step 2** Right-click and choose **Plot L1 Circuit Paths**.
- To move the plot horizontally or vertically, use the left/right arrows and up/down arrows.
- To select all L1 circuit paths within the plot, click **Select L1 Circuit Paths**. To deselect all of these L1 circuit paths, click in an empty plot area.

Editing L1 Circuit Paths

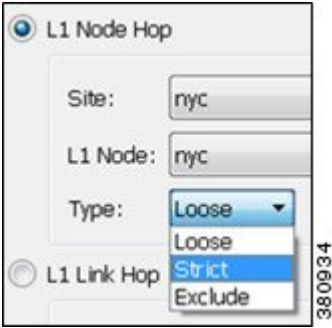
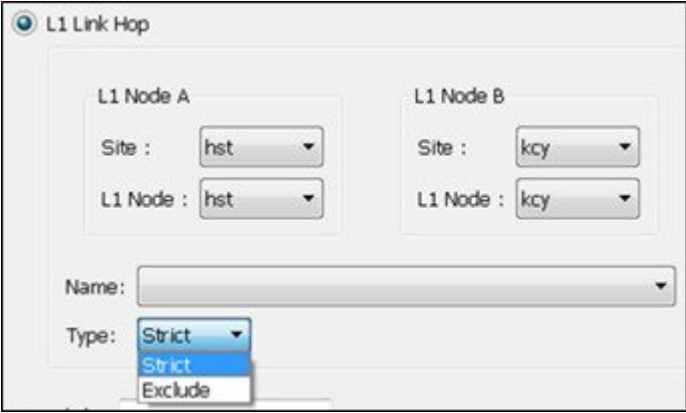
If editing L1 circuit paths, you can change the L1 circuit to which the L1 circuit path is mapped, including the sites and L1 nodes. Other options include:

Field	For Information...
Path Options	Standby Paths and Path Options, on page 196
Delay, Distance	Delay and Distance, on page 212
Setup Priority	L1 Circuit Path Setup Priority, on page 201
Central Frequency ID, Auto Spectrum Assignment	L1 Circuit Wavelengths, on page 201
Feasibility Limit, Acceptance Threshold	L1 Circuit Wavelengths, on page 201
Standby	Standby Paths and Path Options, on page 196 , Resiliency Under L1 Failures, on page 216
Hops	Explicit L1 Circuit Paths Initializer, on page 199 , Creating Individual L1 Circuit Path Hops, on page 227

Creating Individual L1 Circuit Path Hops

To manually create L1 circuit path hops, the L1 nodes or L1 links must already exist. Alternatively, you can use the [Explicit L1 Circuit Paths Initializer, on page 199](#) to set explicit L1 circuit paths.

- Step 1** Right-click an L1 circuit path in the L1 Circuits Paths table and choose **Properties**, or double-click an L1 circuit path.
- Step 2** In the Starts From drop-down list, specify the first L1 node hop in the path as either A or B.
- Step 3** If you are editing an existing hop, double-click it from the Hops list.
If you are creating a hop on this path for the first time, click **Insert After** in the Hops area.
If you are creating a hop on this path and other hops already exist, select an existing hop and then click **Insert Before** or **Insert After**, depending on the required order of the hops.
- Step 4** Create or edit the node or link hop as follows:

L1 Circuit Node Hops	L1 Circuit Link Hops
<p>In the Node Hop area, select the site and its L1 node that you want to set as the hop. Choose the type as loose, strict, or exclude.</p>  <p>Click OK.</p>	<p>Click Link Hop. Choose both endpoints (site and node) of the L1 link hop you are setting. Choose the L1 link name you are creating. Only those L1 links with the endpoints you selected are listed. Choose the type as strict or exclude.</p>  <p>Click OK.</p>

- Step 5** (Optional) Add a central frequency ID value for the L1 circuit path hop.
- Step 6** Repeat Step 3 to Step 5 as needed for each hop in the path. Because WAE Design always chooses the shortest path, you must enter enough hops to create an alternate, longer path.
- Step 7** Click **OK**.
- Step 8** In the L1 circuit path’s dialog box, verify that the list of finished hops is in the order of the desired path, and then click **OK**.

Creating L1 Waypoints

To create waypoints, we recommend that you set the plot options to be able to view them first.

- To view L1 waypoints in the network plot, you must be in the geographic view and the Layer 1 option to view them must be enabled in the Layer 1 tab of the Plot Options dialog box.
- To view waypoints within sites, the site must be viewed in a geographic layout or if using a schematic site view, the site must contain another site. This is set in the Site Canvas field in the General Options tab of the Plot Options dialog box.

The only method in which you can manually specify the Order value is through the L1 link context menu.



Note Once you enter the waypoint order value, you cannot edit it. To change its value, you must delete and recreate it, or insert another waypoint that would then alter the order value accordingly.

If no longitude and latitude coordinates are specified, the waypoint is inserted in the middle between L1 node B and the waypoint closest to L1 node B. If no other waypoints exist on the link, the new waypoint is inserted in the middle of the L1 link.

L1 Link Properties

Step 1 Right-click an L1 link and choose **Properties**, or double-click an L1 link.

Step 2 Click the **Waypoints** tab.

Step 3 If you are creating a waypoint for the first time, click **Insert After**.

If you are creating a waypoint on the L1 link and other waypoints already exist, select an existing waypoint and then click **Insert Before** or **Insert After**, depending on the required order of the waypoints.

Step 4 (Optional) Enter a description, longitude, and latitude.

Step 5 Click **OK**.

L1 Link Context Menu

Step 1 Right-click an L1 link and choose **Insert L1 Link Waypoint**.

Step 2 Enter the number representing the order in which you want the waypoint to appear in proximity to L1 node A.

Step 3 (Optional) Enter a description, longitude, and latitude.

Step 4 Click **OK**.

L1 Waypoint Context Menu

- Step 1** Right-click an L1 waypoint and choose **Insert L1 Link Waypoint > Before/After Selected Waypoint**.
- Step 2** (Optional) Enter a description, longitude, and latitude.
- Step 3** Click **OK**.
-

Duplicating L1 Objects

To duplicate an L1 link, L1 circuit, or L1 circuit path, right-click one and choose **Duplicate**.

- Duplicating an L1 circuit or L1 circuit path also duplicates its L1 circuit path hops.
- When duplicating L1 circuits, you can optionally choose to create as many duplicated L1 circuits as necessary. New L1 ports are created for each duplicated L1 circuit.

When duplicating an L3 circuit, you have the option to duplicate its associated L1 circuit if one exists and create an automatic mapping to it. If you select no, the newly duplicated L3 circuit maps to the existing L1 circuit.

Creating Central Frequency ID Blocked List

- Step 1** Choose **Insert > Layer 1 > Central Frequency ID Blacklist**, or right-click the plot and choose **New > Layer 1 > Central Frequency ID Blacklist**.
- Step 2** Enter a unique name for the central frequency ID blocked list.
- Step 3** Select from the list of frequency IDs using radio button. You can select multiple frequencies to be blocked. Use the **Filter** button to narrow the frequency IDs from the list.
- Step 4** (Optional) Enter the **Frequency ID Lower Bound** to specify the lower bound on available central frequency IDs.
- Step 5** (Optional) Enter the **Frequency ID Upper Bound** to specify the upper bound on available central frequency IDs.
- Step 6** Click **OK**.
-

Inserting L1 Diversity Constraint

- Step 1** Choose **Insert > Layer 1 > L1 Diversity Constraints**.
- Step 2** Enter a unique name for the L1 Diversity Constraint in **Name** field.
- Step 3** In the L1 Circuit A area, select the Site, L1 Node, L1 Port for L1 Port A and L1 Port B using the dropdown list. A list of L1 Circuits for the specified combination is displayed in the **Name** field. Select a L1 Circuit combination from the dropdown list.
- Step 4** Similarly, specify the L1 Port A and L1 Port B for L1 Circuit B and select from the list of matching combinations available.
- Step 5** Set the Constraints:
- **L1 Node Diversity**: Select this to set the respective paths of the L1 Circuits to be diverse in terms of L1 Nodes.

- **L1 Link Diversity:** Select this to set the respective paths of the L1 Circuits to be diverse in terms of L1 Links.
- **SRLG Diversity:** Select this to set the respective paths of the L1 Circuits to be diverse in terms of SRLGs.

Note WAE Design allows you to select multiple levels of diversity at the same time.

- **Active:** Set this property to Active for simulation to consider the L1 Diversity Constraint.

Step 6 Click **OK**.

Creating L1 and L3 Mappings

Creating L3-L1 Links Between Nodes

You can model the physical relationship between an L3 node and an L1 node to create an L3-L1 link, which is an object listed in the L3-L1 Links table. Creating these relationships can greatly expedite the process of creating L1 circuits from L3 circuits. For information on how these are used in that process, see [Mapping Multiple L1 Circuits to L3 Circuits, on page 231](#).



Note For L3-L1 links to be valid for creating L1 circuits, there must be a pair of them. That is, there must be one entry for each L3 end node of the L3 circuit. The L1 circuit that is created uses the L1 nodes to which those L3 nodes are mapped.

Example: This L3-L1 Links table shows an example of three L3-L1 links:

- L3 node cr1.hst is connected to (linked to) the L1 node hst.
- L3 node cr1.okc is connected to the L1 node okc.
- L3 node cr2.wdc is connected to the L1 node wdc.

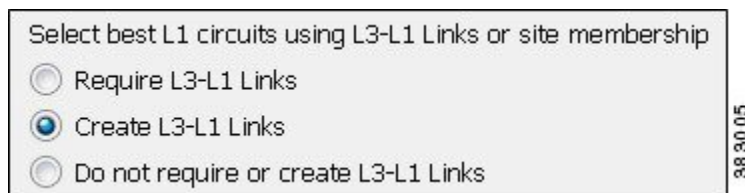
Node	L1 Node
cr1.hst	hst
cr1.okc	okc
cr2.wdc	wdc

383004

-
- Step 1** Choose **Insert > Layer 1 > L3-L1 Link**, or right-click in an empty area and choose **New > Layer 1 > L3-L1 Link**. Note that preselecting an L3 node or L1 node has no effect on the selections opened in the dialog box.
- Step 2** Choose the L3 node and the site in which it is contained, if applicable.
- Step 3** Choose the L1 node and the site in which it is contained, if applicable.
- Step 4** Click **OK**.
-

Mapping Multiple L1 Circuits to L3 Circuits

The L1 Circuits for L3 Circuits tool creates L1 circuits based on L3 circuits and associates them with the L3 circuits using the following sequence of priorities. For each L1 circuit created, an associated L1 circuit path is also created. Nested objects are taken into account.



- If the L3 circuit already has an associated L1 circuit, one is not created.
- WAE Design checks the L3-L1 Links table for valid entries.
 - If multiple pairs of L3-L1 links exist that could create the L1 circuit, one is randomly used.
 - If no valid L3-L1 link entry exists and if “Require L3-L1 Links” is selected, no L1 circuit is created.
 - If no valid L3-L1 link entry exists and if “Do not require or create L3-L1 Links” is selected, for each L3 circuit WAE Design checks for sites that contain both the L3 end node and at least one L1 node.
 - If multiple L1 nodes are collocated within the same sites as the L3 nodes, one is randomly selected.
 - If this collocation is not found, WAE Design checks for L1 nodes within the parent sites of the sites containing the L3 end nodes. If none are found, an L1 circuit is not created.
- If “Create L3-L1 Links” is selected, the L3-L1 links are created that map the respective L3 and L1 nodes. If needed, you can make changes to the L1 circuit’s endpoints and the L3-L1 links that are created by editing their respective Properties dialog boxes.

Creating L1 circuits in this manner automatically creates and assigns L1 ports for the newly created L1 circuits.

-
- Step 1** In this process, you can optionally use L3-L1 links. If you are using them, verify you have the set of L3-L1 links you need, and if not, create them. See [Creating L3-L1 Links Between Nodes, on page 230](#).
- Step 2** If you want to create L1 circuits for only selected L3 circuits, first select those L3 circuits. Otherwise, L1 circuits are created for as many L3 circuits as possible.

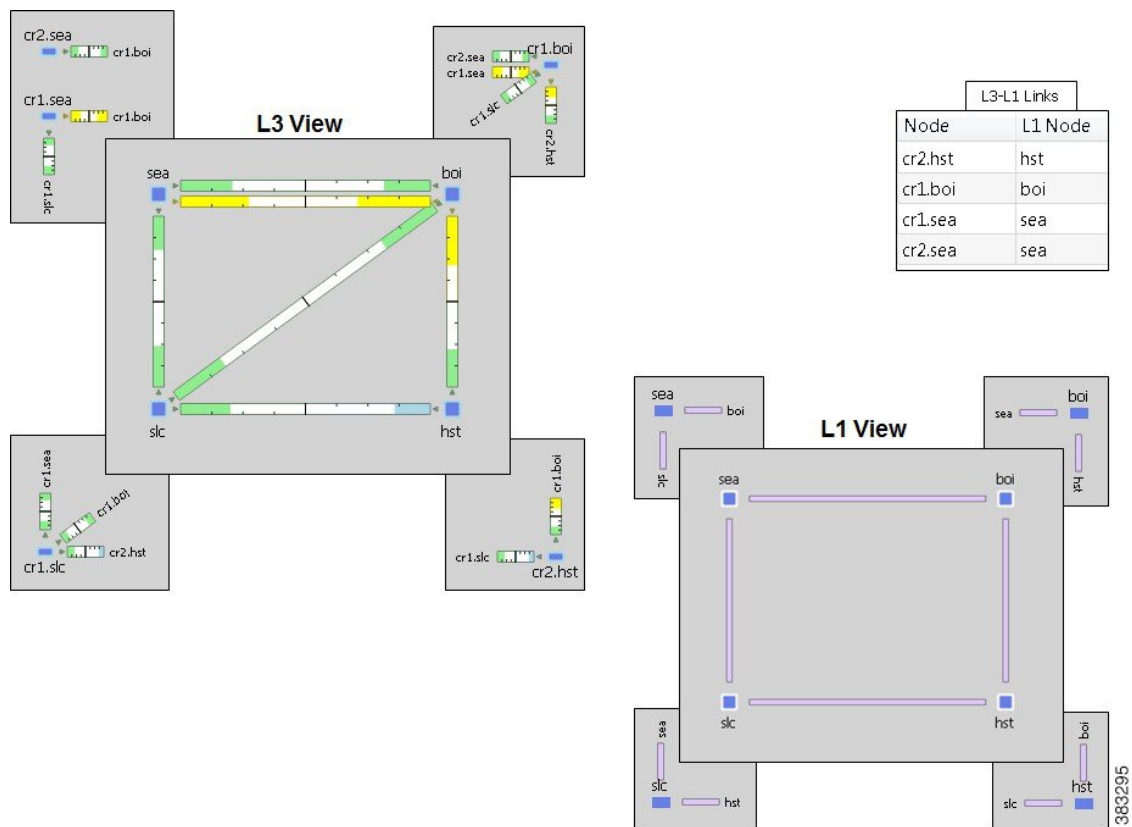
Step 3 Right-click in an empty area and choose **Layer 1 > L1 Circuits for L3 Circuits**, or choose **Insert > Layer 1 > L1 Circuits for L3 Circuits**.

Step 4 Choose either to require the use of existing L3-L1 links, to create them, or neither. Then click **OK**.

Examples

Figure 92: [Topology and Existing L3-L1 Links](#), on page 232 shows a plan file containing six L3 circuits and four L3-L1 links. All sites except sea contain one L3 node and one L1 node. The sea site contains two L3 nodes and one L1 node. Refer to the diagram for details of the objects' relationships and to see the objects in the L3-L1 Links table. There are no L1 circuits.

Figure 92: Topology and Existing L3-L1 Links



If you click **Require L3-L1 Links**, WAE Design creates three L1 circuits: one between hst and boi, and two between sea and boi.

Require L3-L1 Links

L1 Circuits		
Name	L1 Node A	L1 Node B
L1Circuit[1]	boi	hst
L1Circuit[2]	boi	sea
L1Circuit[1]	boi	sea

38 32 98

If you click **Do not to require or create L3-L1 Links**, WAE Design creates six L1 circuits: three using the L3-L1 Links table and the remaining three by determining that the source and destination sites both contain L3 and L1 nodes.

Do not require or create L3-L1 Links

L1 Circuits		
Name	L1 Node A	L1 Node B
L1Circuit[1]	boi	sea
L1Circuit[1]	boi	hst
L1Circuit[1]	boi	slc
L1Circuit[1]	hst	slc
L1Circuit[1]	sea	slc
L1Circuit[2]	boi	sea

38 32 96

Using the same topology as in [Figure 92: Topology and Existing L3-L1 Links, on page 232](#) but with no pre-existing L3-L1 links, if you click **Create L3-L1 Links**, WAE Design creates the same six L1 circuits as if you had selected no options. However, it also creates five L3-L1 links: one for each L3 node having an L1 circuit for which an L1 circuit was created.



Created L3-L1 Links

L3-L1 Links	
Node	L1 Node
cr1.boi	boi
cr2.hst	hst
cr1.sea	sea
cr2.sea	sea
cr1.slc	slc

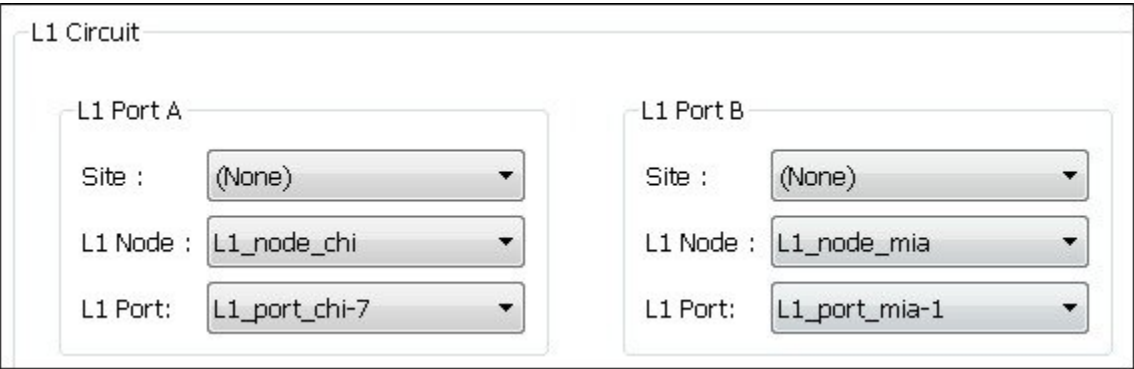
Created L1 Circuits

L1 Circuits		
Name	L1 Node A	L1 Node B
L1Circuit[1]	boi	sea
L1Circuit[1]	boi	hst
L1Circuit[1]	boi	slc
L1Circuit[1]	hst	slc
L1Circuit[1]	sea	slc
L1Circuit[2]	boi	sea

383297

Mapping an L1 Circuit to an L3 Circuit

- Step 1** Right-click an L3 circuit in the Circuits table and choose **Properties**, or double-click an L3 circuit.
- Step 2** Click the **Advanced** tab.
- Step 3** In the L1 Circuit area, choose the applicable site, L1 nodes, and L1 ports for ports A and B.



- Step 4** Create and map a new L1 circuit to the L3 circuit, or map an existing one to the L3 circuit.
 - Map existing L1 circuit—Click **Select** and then choose the L1 circuit from the drop-down list of all the available circuits.
 - Create L1 circuit—Click **Create** and enter a new name for the L1 circuit.

Then click **OK**. Creating L1 circuits in this manner automatically creates and assigns L1 ports for the newly created L1 circuits.

Mapping an L1 Circuit to an L3 Port Circuit

Each port circuit can be indirectly associated with an L1 circuit via direct mappings of L1 ports to L3 ports.

- Step 1** Right-click an L3 port circuit in the Port Circuits table and choose **Properties**, or double-click an L3 port circuit.
- Step 2** In the L1 Circuit area, select the sites (if applicable), L1 nodes, and optionally L1 ports for both ends of the L1 circuit. Note that if specifying L1 ports and mapping an L1 circuit, the pair must be already associated with an L1 circuit.
- Step 3** Map an existing L1 circuit or create a new one, and then click **OK**.
- To map an existing L1 circuit, click **Select** and then choose the L1 circuit from the drop-down list of all the available circuits. Note that this list is populated only if the L1 ports selected are connected via an L1 circuit.
 - To create an L1 circuit, click **Create**. Enter a new L1 circuit name or use the default name. After you click **OK**, you can create L3-L1 links. If you are creating an L1 circuit and if you did not choose L1 ports, L1 ports are created for the new L1 circuit.
-

Mapping L1 Ports to L3 Ports

If an L3 port circuit exists whose L3 ports are associated with L1 ports, that L3 port circuit is automatically associated with the L1 circuit that also contains those same L1 ports.

Mapping L1 ports to L3 ports automatically associates the L1 circuit to the port circuit if a port circuit exists.

- Step 1** Right-click a port in the Ports table and choose **Properties**, or double-click a port.
- Step 2** Click the **Layer 1** tab.
- Step 3** If applicable, choose the L1 node's site.
- Step 4** Choose the L1 node and L1 port, and then click **OK**.
-

Visualizing the L1 Network

Many of the visualization tools available for the L3 view are also available for L1. These tools are described in the *Cisco WAE Network Visualization Guide*.

- **Assign Sites to Nodes**—Automatically create sites based on L1 node names and a set of parameters that define which part of the node name to use.
- **Arrange Sites and Nodes**—Arrange sites and nodes using tools that offer a variety of alignment options or arrange sites to specific geographic locations. You can also show and hide sites, and change where their site name is displayed in the plot.

- Set Plot Backgrounds—Set the background for the network plot, including schematic view, outline of a geographic background, and detailed geographic background.
- Create or Edit Layouts—Create multiple layouts for a single plan file, each containing its own set of visual properties.

Editing L1 Plot Options

The Plot Options dialog box has a Layer 1 tab that lets you change how L1 links appear, whether to view another layer in the background, and whether to show or hide sites from another layer. Because these are plot options, they apply to the current layout and are saved on a per-layout basis when you save the plan file. You can also save them as user defaults. For information on user and system defaults, see the *Cisco WAE Network Visualization Guide*.

-
- Step 1** Click the **Plot Options** icon in the top right of the Visualization toolbar, or choose **Edit > Plot > Plot Options > Edit Options**.
- Step 2** Click the **Layer 1** tab.
- Step 3** Edit Layer 1 fields as described in the following sections, and then click **OK**.
-

L1 Link Appearance

- L1 Link Width—The L1 link width is only for visualization purposes and is not related to any property.
- L1 Link Text—By default, there is no text on L1 links. Use this option to display the link name, distance, delay, or the number of L1 circuit paths using the link.
- Show L1 Link Waypoints—If checked, L1 waypoints are shown in the network plot.
- Visualize L1 Link Utilization—If checked, L1 link utilization is shown in the network plot.

Multi-Layer Visualization

The first two background options let you show or hide Layer 1 in the Layer 3 view, and vice versa.

You can also set the degree of opacity of the background objects. The higher the value, the denser (darker) the background objects appear.



Note In the Visualization toolbar, there is a Set to Foreground icon that brings selected objects to the foreground and sends unselected objects to the background. This, too, has an opacity value for specifying the transparency of background objects. This feature has precedence over the layer opacity set in the Plot Options dialog box.

By default, sites containing only L1 nodes are hidden in the L3 view, and sites containing only L3 nodes are hidden in the L1 view. Use these selections to toggle whether to hide sites containing only L1 or L3 nodes. If you are showing a layer in the background, you cannot hide per-layer sites within that view. You can show and hide individual sites by right-clicking one and choosing **Layout > Hide/Show Site**.

Background Layer

- Show Layer 1 in the Background of the L3 View
- Show Layer 3 in the Background of the L1 View

Opacity of Background Layer

- Hide L1 Sites in L3 View
- Hide L3 Sites in L1 View

S1515



CHAPTER 13

Quality of Service Simulation

Quality of Service (QoS) is a means of ensuring high-quality performance for critical applications. The concept is that because requirements of some users and services are more critical than others, some traffic requires preferential treatment.

Using WAE Design QoS features, you can ensure that service levels are met without reactively expanding or over-provisioning the network. QoS features are available for undifferentiated traffic, for service classes alone, for interface queues alone, and for service classes that are mapped to queues.

- Undifferentiated traffic—Aggregate traffic on an interface.
- Service class—A user-defined classification of traffic that is not discovered by WAE. Examples include voice, video, and data. Service classes apply to the entire network unless you map them to specific queues.
- Queue—In live networks, traffic waits in conceptual lines (queues) and then is forwarded over an interface on a per-queue basis according to QoS parameters. Similarly in WAE Design, each queue has a set of user-defined QoS parameters (interface queue properties) that specify how these queues are prioritized and what percentage of traffic they carry. An interface contains zero or more queues that are discoverable by WAE. You can also manually create and configure them. The traffic per queue is also discovered.

This section contains the following topics:

- [QoS Parameters, on page 239](#)
- [QoS Bound and QoS Violations, on page 241](#)
- [Viewing Queue and Service Class Information, on page 247](#)
- [Viewing QoS Bounds and QoS Violations, on page 247](#)
- [Creating Service Classes, on page 248](#)
- [Creating Queues, on page 249](#)
- [Assigning Queues to Interfaces, on page 250](#)
- [Mapping Service Classes to Queues, on page 250](#)
- [Creating or Editing Policy Groups for Interfaces, on page 250](#)
- [Creating or Editing Service Class Policies, on page 251](#)
- [Editing Interface Queues Properties \(QoS Requirements\), on page 252](#)

QoS Parameters

In WAE Design, QoS requirements are defined by policies and interface queue properties ([Figure 93: Policies and Interface Queue Parameters, on page 241](#)).

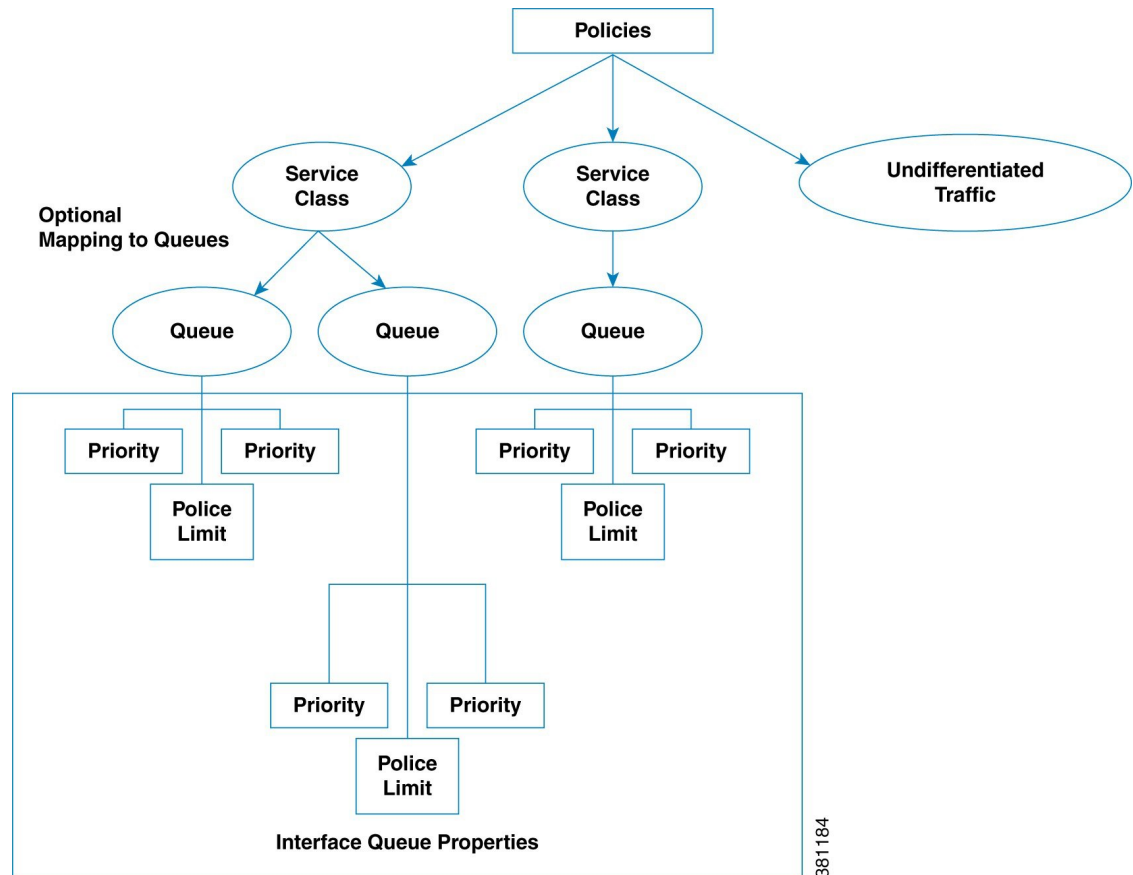
Policy—Maximum percentage of traffic capacity that can be utilized by either a service class or by undifferentiated traffic. There are two policies: one for normal operation and one for worst-case scenarios. Policies set on service classes do not affect QoS requirements of any other service class. Nor would this parameter have any effect on live network behavior.

In addition to setting policies for the network, you can refine the policy to a group of interfaces (called an *interface policy group*). For example, you might need to model behavior for service classes offered only on interfaces of a specific capacity, or you might need to observe service class traffic on only one area of the network.

Interface queue properties—Configured parameters that would affect routing behavior in a live network. In WAE Design, the interface queue properties are priority, weight, and police limit.

- The *priority* identifies the precedence of the queue. For example, traffic in a priority 1 queue is routed before traffic in a priority 2 queue. Queues with the same priority evenly share the capacity based on weighted-round robin (WRR) calculations. You can change this behavior using the weight and police limit parameters. There are an unlimited number of priorities, though most networks only use no more than three. By default, queues do not have priorities.
- The *weight* is the percentage of preference given to queues of an equal priority level, which enables the network to fairly distribute the load among available resources. For example, if 10 Gbps were passing through a 10GbE interface on two priority 1 queues, by default 5 Gbps would pass through each queue. However, if you set the weight of one queue to 75% and the other to 25%, the distribution would be 7 Gbps and 2.5 Gbps, respectively. By default, all queues have a weight of 100%.
- The *police limit* is the maximum percentage of available capacity permitted through a queue of a given priority level, thereby preventing traffic from higher priority queues from starving lower priority queues. For example, if the interface is a 20GbE and a priority 1 queue has a police limit of 40%, then only 8 Gbps of interface traffic can go through this queue. By default, all queues have a police limit of 100%. To see examples of this *starvation*, refer to the examples in [Policies and QoS Bound Calculations, on page 243](#), where you can see that lower priority queues received zero traffic due to priority settings.

Figure 93: Policies and Interface Queue Parameters



QoS Bound and QoS Violations

WAE Design uses the concepts of *QoS bound* and *QoS violation* as a way of identifying whether QoS parameters are being met or surpassed, thus better enabling you to plan for service requirements across the network. Policies and queue properties determine the QoS bound calculation. In turn, this calculation determines whether there is a violation.

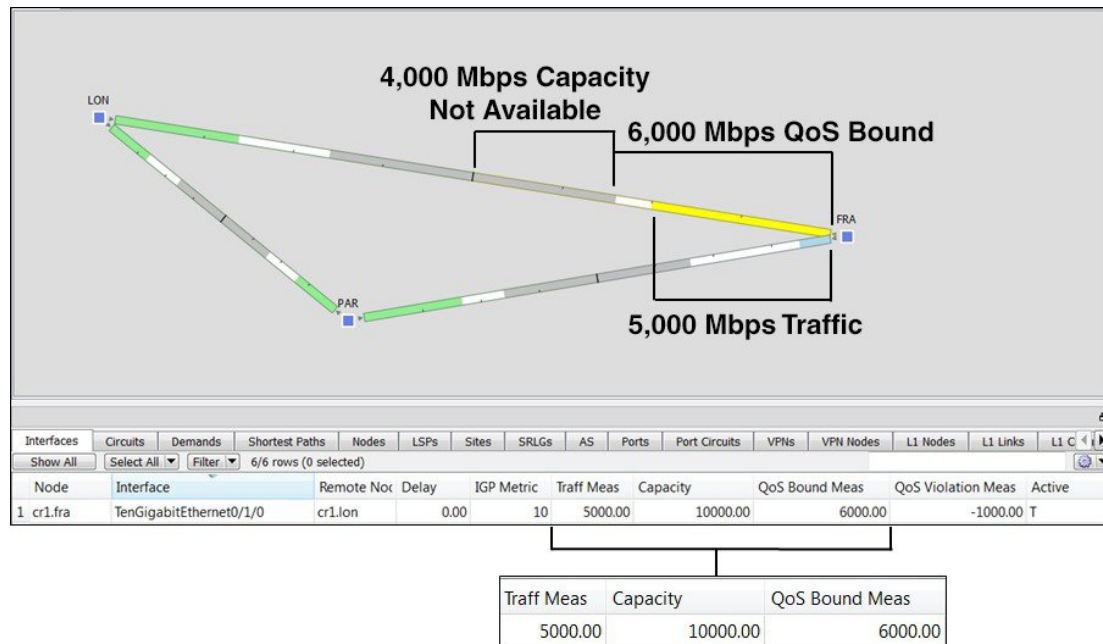
QoS Bound—Maximum interface capacity available without violating these QoS requirements. A separate QoS bound is calculated for both policy and interface queue properties.

QoS Bound for...	Calculation Based on...
Interface Queues	Combination of interface queue properties, or in live networks, it is the capacity percentage that is discovered.
Service class	Policy
Service class mapped to queues	The lower of these two calculations is used: <ul style="list-style-type: none"> • Policy for service class • Queue properties for queues

QoS Bound for...	Calculation Based on...
Undifferentiated traffic	Policy

In the plot, the QoS bound on an interface is a combination of the color and white (Figure 94: QoS Bound and Available Interface Capacity, on page 242). The interface capacity that is not available because it exceeds the QoS bound is in gray. Columns that convey the QoS bound information are QoS Bound Meas, QoS Bound Meas (%), QoS Bound Sim, and QoS Bound Sim (%).

Figure 94: QoS Bound and Available Interface Capacity



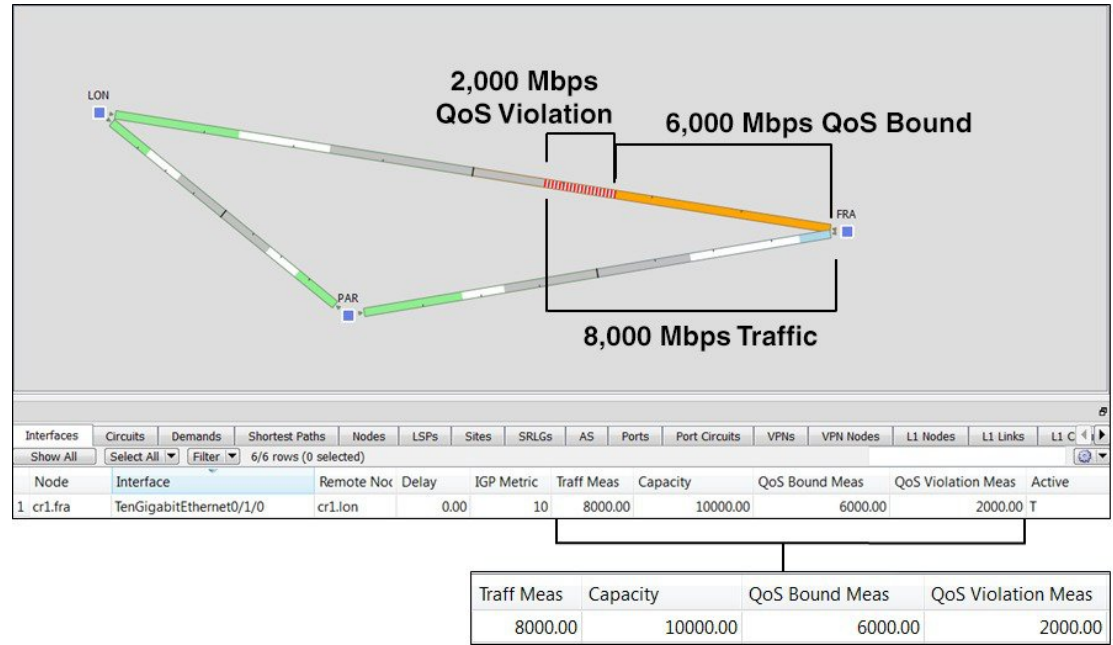
381183

QoS bound calculations are a set of decisions being made to determine how to raise traffic on the queues until that traffic cannot be raised any further. This capacity, or the reason the traffic cannot be raised further, is defined both by the QoS parameters and the amount of traffic. For example, when traffic arrives at Queue X, WAE Design fixes the traffic on all other queues and then determines how it can raise the traffic on Queue X until some other traffic blocks it.

For those queues that do not reach full capacity, unused queue capacity is made available for other queues.

QoS Violation—Total traffic minus the capacity permitted for the queue (QoS bound). A violation occurs if the maximum QoS capacity allotted through policies and interface queue properties is exceeded. If the number appearing in the QoS Violation column is positive, the allotted capacity has been surpassed. If the number is negative, the allotted capacity has not been reached. In the plot, traffic exceeding the QoS bound appears in red and white stripes on the interface in violation (Figure 95: Example QoS Violation, on page 243).

Figure 95: Example QoS Violation



381185

Policies and QoS Bound Calculations

If no other QoS parameters are set via the interface queue properties of priority, weight, and police limit, the QoS bound is equivalent to the policy set.

Table 11: Example Policies and QoS Bound Calculations

Example Configuration	QoS Bound	QoS Violation (Positive # = Violation)
Interface capacity = 10,000 Mbps Undifferentiated traffic = 5000 Mbps Normal operation policy = 60%	6000 Mbps (60%)	-1000 Mbps (-10%) Because this number is negative, there is no capacity violation (Figure 94: QoS Bound and Available Interface Capacity, on page 242).
Interface capacity = 10,000 Mbps Undifferentiated traffic = 8000 Mbps Normal operation policy = 60%	6000 Mbps (60%)	2000 Mbps (20%) Because this number is positive, there is a capacity violation (Figure 95: Example QoS Violation, on page 243).
Interface capacity = 10,000 Mbps Voice traffic = 6000 Mbps Video traffic = 2000 Mbps Voice normal operation policy = 90% Video normal operation policy = 60%	Voice = 9000 Mbps (90%) Video = 6000 Mbps (60%)	Voice = -3000 Mbps (30%) Video = -4000 Mbps (40%)

Interface Queue Properties and QoS Bound Calculations

WAE Design simultaneously calculates QoS bound for each queue in the interface. In so doing, WAE Design uses the interface queue parameters (priority, weight, and police limit) and the traffic measured or simulated for all queues in the interface. Priority is always considered first. If there are queues of equal priority, then weight is applied next.

- Queues with priority 1 share all available interface capacity. Their weight and police limits further refine how much each priority 1 queue can use (their QoS bound). Each priority 1 queue can borrow available capacity from other priority 1 queues up to the limit of their QoS bound.
- The available capacity for priority 2 queues is the total interface capacity less all capacity consumed by priority 1 queues. The process then begins again for all priority 2 queues. Their weight and police limits determine their QoS bound, and priority 2 queues can borrow capacity from each other up to the limits set by the QoS bound.
- This process continues for each successive priority level. Traffic that is outside any QoS bound is dropped to the lowest priority of all traffic on the interface.

For discovered networks with measured traffic, if no WAE Design QoS parameters are set, the QoS bound is based on whatever capacity percentages the live network has for each queue.

Priority

Provided policies are not set that further affect the QoS bound, a queue's QoS bound is calculated as follows:

- Priority 1 QoS bound = 100% of the interface capacity.
- Priority 2 QoS bound = Total interface capacity – amount of traffic consumed by priority 1 queues.
- Priority 3 QoS bound = Total interface capacity – amount of traffic consumed by (priority 1 + priority 2 queues).
- QoS bound for each succeeding priority follows this same pattern where the traffic consumed by all higher priority queues is subtracted from the total interface capacity.

Table 12: Examples of Priority QoS Bound Calculations

Example Configuration	QoS Bound	QoS Violation (Positive # = Violation)	QoS Bound Calculations
Interface capacity = 20,000 Mbps EF traffic = 6000 Mbps; priority = 1 BE traffic = 3000 Mbps; no priority set	EF = 20,000 Mbps BE = 14,000 Mbps	EF = -14,000 Mbps BE = -11,000 Mbps	EF = Total interface capacity because it is the only priority 1 queue BE = 20,000 (interface capacity) – 6000 (consumed by higher priority queues)
Interface capacity = 10,000 Mbps EF Traffic = 6000 Mbps; priority = 1 BE Traffic = 5000 Mbps; priority = 2	EF = 10,000 Mbps BE = 4000 Mbps	EF = -4000 Mbps BE = 1000 Mbps	EF = Total interface capacity because it is the only priority 1 queue BE = 10,000 (interface capacity) – 6000 (consumed by higher priority queues)

Weight

The weight identifies the forwarding precedence for queues of equal priority. If weights for queues of the same priority do not add up to 100%, weights are converted proportionally so they do add up to 100%.

Table 13: Examples of Weight QoS Bound Calculations

Example Configuration	QoS Bound	QoS Violation (Positive # = Violation)	QoS Bound Calculations
Interface capacity = 10,000 Mbps AF1 traffic = 3000 Mbps; priority = 1; weight = 100% AF2 traffic = 6000 Mbps; priority = 1; weight = 100%	AF1 = 5000 Mbps AF2 = 7000 Mbps	AF1 = -2000 Mbps AF2 = -1000 Mbps	AF1 = Half of capacity for priority 1 queues because both queues have equal weights AF2 = 5000 (half of capacity) + 2000 (unused AF1 capacity)
Interface capacity = 10,000 Mbps AF1 = 5000 Mbps; priority = 1; weight = 60% AF2 traffic = 6000 Mbps; priority = 1; weight = 40%	AF1 = 6000 Mbps AF2 = 5000 Mbps	AF1 = -1000 Mbps AF2 = 1000 Mbps	AF1 = 60% of capacity for all priority 1 queues AF2 = 10,000 (interface capacity) - 5000 (consumed by AF1 queue)

Police Limits

Priority 1 queues have 100% of the interface traffic, and thus starve out the remaining queues. To prevent this queue starvation, use police limits to configure how much of the maximum percentage should be available for a given priority level.

Table 14: Examples of Police Limit QoS Bound Calculations

Example Configuration	QoS Bound	QoS Violation (Positive # = Violation)	QoS Bound Calculations
Interface capacity = 10,000 Mbps EF traffic = 1000 Mbps; priority = 1; police limit = 50% BE traffic = 2000 Mbps; priority = 2	EF = 5000 Mbps BE = 9000 Mbps	EF = -4000 Mbps BE = -7000 Mbps	EF = 50% of total interface capacity BE = 10,000 (interface capacity) - 1000 (capacity consumed by EF)
Interface capacity = 10,000 Mbps EF traffic = 1000 Mbps; priority = 1; police limit = 5% BE traffic = 2000 Mbps; priority = 2	F = 500 Mbps BE = 9500 Mbps	EF = 500 Mbps BE = -7500 Mbps	EF = 5% of total interface capacity BE = 10,000 (interface capacity) - 500 (capacity consumed by EF)

Example Configuration	QoS Bound	QoS Violation (Positive # = Violation)	QoS Bound Calculations
Interface capacity = 10,000 Mbps EF = 3000 Mbps; priority = 1; police limit = 20% AF1 traffic = 4000 Mbps; priority = 2; police limit = 75% AF2 traffic = 2500 Mbps; priority = 2; police limit 25%	EF = 2000 Mbps AF1 = 6000 Mbps AF2 = 4000 Mbps	EF = 1000 Mbps AF1 = -2000 Mbps AF2 = -1500 Mbps	EF = 20% of total interface capacity AF1 = 75% of (10,000 [interface capacity] – 2000 [capacity consumed by EF]) AF2 = 10,000 (interface capacity) – 2000 (capacity consumed by EF) – 4000 (capacity consumed by AF1)

Interface QoS Bound Calculations Using Multiple QoS Parameters

WAE Design calculates a QoS bound for interface queues based on all three parameters if they are all configured: priority, weight, and police limits.

Table 15: Examples of Interface QoS Bound Calculations Using Multiple QoS Parameters

Example Configuration	QoS Bound	QoS Violation (Positive # = Violation)	QoS Bound Calculation
Interface capacity = 10,000 Mbps EF = 3000 Mbps; priority = 1; police limit = 20% AF1 traffic = 4000 Mbps; priority = 2; weight = 75% AF2 traffic = 2500 Mbps; priority = 2; weight = 25%	EF = 2000 Mbps AF1 = 6000 Mbps AF2 = 4000 Mbps	EF = 1000 Mbps AF1 = -2000 Mbps AF2 = -1500 Mbps	EF = 20% of total interface capacity AF1 = Maximum of these two values. <ul style="list-style-type: none"> 75% of (10,000 [interface capacity] – 2000 [capacity consumed by EF]) 8000 (available capacity) – 2500 (AF2 traffic) AF2 = Maximum of these two values. <ul style="list-style-type: none"> 25% of (10,000 [interface capacity] – 2000 [capacity consumed by EF]) 8000 (available capacity) – 4000 (AF1 traffic)

Service Class QoS Bound Calculations Using Multiple QoS Parameters

If service classes have policies and they are mapped to queues, WAE Design calculates a QoS bound for both. WAE Design then uses the lowest value of the two so as to enforce restrictions in the strictest possible manner.

Example:

Interface capacity = 10,000 Mbps

QoS bound for service class = 50%, or 5000 Mbps based on policy

QoS bound for EF queue = 7500 Mbps based on combined parameters of priority, weight, and police limit

The QoS bound for this service class is 5000 Mbps because the policy QoS bound calculation is lower.

Viewing Queue and Service Class Information

Table 16: Queue and Service Class Information

To View	Show or Select
Queue information	Show the Interface Queue table. Select the queue from the QoS drop-down menu. Both the plot and the Traff Meas and Traff Sim columns display traffic data specific to the queue type selected.
Per-queue traffic in the Interfaces table	Select the queue from the QoS drop-down menu. Both the plot and the Traff Meas and Traff Sim columns display data specific to the queue type selected.
Per-service-class traffic	Select the service class from the QoS drop-down menu. Both the plot and the Traff Meas and Traff Sim columns in the Interfaces table display data specific to the service class selected.
Service class demands	Show the Service Class column in the Demands table.

Viewing QoS Bounds and QoS Violations

As [Figure 94: QoS Bound and Available Interface Capacity, on page 242](#) and [Figure 95: Example QoS Violation, on page 243](#) demonstrate, the QoS bounds and QoS violations appear in the plot view. [Table 17: QoS Bounds and QoS Violations, on page 247](#) lists the available column options to display numeric values of the QoS bound calculations. For information on QoS values as they relate to VPNs, see [VPN Simulation, on page 279](#).

Table 17: QoS Bounds and QoS Violations

To View	Show This Column in Interfaces, Circuits, or Interface Queues Table
Measured Traffic	
Maximum capacity before a QoS bound is violated under normal operations	QoS Bound Meas
QoS bound as a percentage of total interface capacity	QoS Bound Meas (%)
QoS violations under normal operations; if the number is positive, there is a violation	QoS Violation Meas
QoS violation as a percent of the total interface capacity	QoS Violation Meas (%)
Simulated Traffic	
Maximum capacity before a QoS bound is violated under normal operations	QoS Bound Sim
QoS bound as a percentage of total interface capacity	QoS Bound Sim (%)

To View	Show This Column in Interfaces, Circuits, or Interface Queues Table
QoS violations under normal operations; if the number is positive, there is a violation	QoS Violation Sim
QoS violation as a percent of the total interface capacity	QoS Violation Sim (%)
Worst-Case Traffic	
Maximum capacity before a QoS bound is violated under worst-case operations	WC QoS Bound
WC QoS bound as a percentage of total interface capacity	WC QoS Bound (%)
QoS violations under worst-case operations; if the number is positive, there is a violation	WC QoS Violation
WC QoS violation as a percent of the total interface capacity	WC QoS Violation (%)
Service class causing the worst-case utilization	WC Service Class

Creating Service Classes

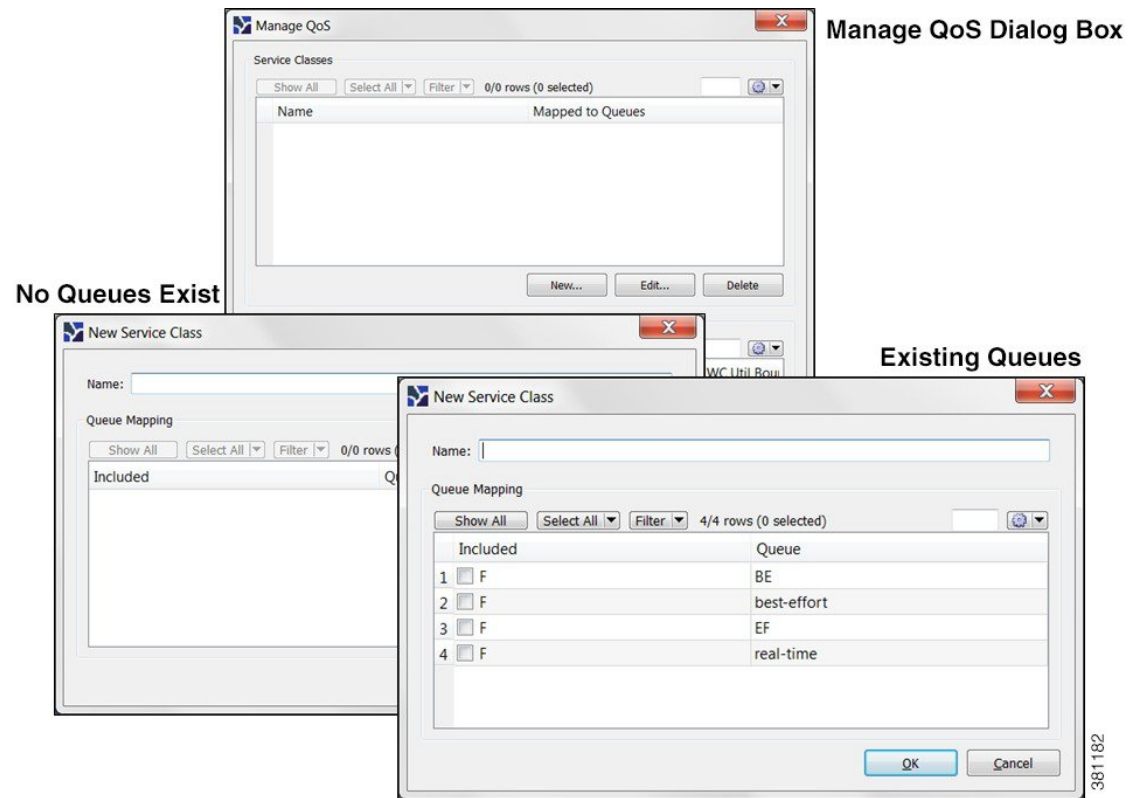
Step 1 Open the Manage QoS dialog box in one of two ways:

- Choose **Edit > Manage QoS**.
- Choose **Manage QoS** from the QoS drop-down menu in the toolbar.

Step 2 Click **New**. The New Service Class dialog box that opens contains queues if they have already been discovered or if they have been manually created. Otherwise, the dialog box is empty. For instructions on how to create queues, see [Creating Queues, on page 249](#).

- a) Enter a unique name.
- b) (Optional) If queues exist and if you want to map this new service class to one or more queues, select them from the list. If queues do not exist, but you want them, you must manually create the queues and then return to this dialog box to select them. See [Mapping Service Classes to Queues, on page 250](#).
- c) Click **OK**.

Step 3 Click **OK** to save and exit.

What to do next**Figure 96: Create Service Classes**

Creating Queues

WAE Design discovers queues. However, you can manually add them. Once discovered or created, queues appear in the Interface Queues table.

-
- Step 1** By default, queues are assigned to all interfaces. If you want this new queue to be assigned to specific interfaces, you must first select one or more interfaces.
- Step 2** Open the New Interface Queues Properties dialog box in one of two ways:
- Choose **Insert > Interface Queues**.
 - Right click in the plot area choose **New > Interface Queues**.
- Step 3** Enter the queue name.
- Step 4** (Optional) Enter the queue properties of priority, weight, and police limit. For information on how these queue properties behave, see [Interface Queue Properties and QoS Bound Calculations, on page 244](#).
- Step 5** Click **OK**. The new queue appears as an option in the QoS drop-down menu in the toolbar, as well as in the Manage QoS dialog box.
- Step 6** (Optional) Map a service class to the queue. For instructions, see [Mapping Service Classes to Queues, on page 250](#).

Another way to create queues is to edit the name of an existing one. For instructions, see [Assigning Queues to Interfaces, on page 250](#).

Assigning Queues to Interfaces

The easiest way to assign queues to interfaces is to select the interface before creating the queue (see [Creating Queues, on page 249](#)). However, you can follow these steps to reassign the queue to different interfaces.

- Step 1** From the Interface Queues table, select one or more queues.
 - Step 2** Double-click; a Properties dialog box opens.
 - Step 3** From the Node list, select the node associated with the interface.
 - Step 4** From the Interface list, select the interface to which you are assigning the queue.
 - Step 5** (Optional) Change the queue name or add a new one.
 - Step 6** (Optional) Specify QoS requirements in the Priority, Weight, and Police Limit fields. By default, queues do not have a priority and both their weight and police limits are 100%. For information on how these properties behave, see [Interface Queue Properties and QoS Bound Calculations, on page 244](#).
 - Step 7** Click **OK**.
-

Mapping Service Classes to Queues

To map service classes to queues, those queues must first exist either because they were discovered by WAE or because you manually added them.

- Step 1** Open the Manage QoS dialog box in one of two ways:
 - Choose **Edit > Manage QoS**.
 - Choose **Manage QoS** from the QoS drop-down menu in the toolbar.
 - Step 2** From the Service Classes list, select one service class.
 - Step 3** Click **Edit**.
 - Step 4** In the Edit Service Class dialog box, select one or more queues. Click **OK**.
 - Step 5** Repeat for each service class to which you are mapping queues.
 - Step 6** Click **OK** to save and exit.
-

Creating or Editing Policy Groups for Interfaces

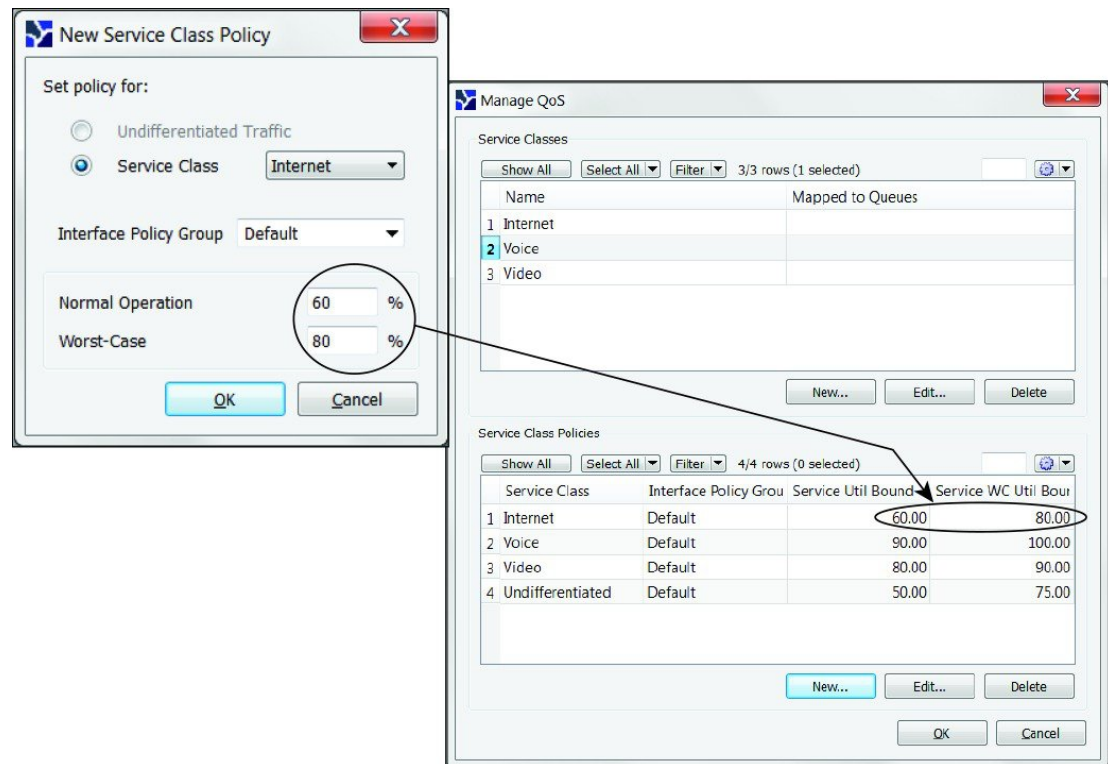
Creating a policy group for interfaces lets you set policies for the group in the Manage QoS dialog box.

-
- Step 1** Select one or more interfaces.
- Step 2** Double-click to open the Properties dialog box.
- Step 3** Click the **Advanced** tab.
- Step 4** In the QoS Policy Group field, you have three options:
- To add this interface to an existing policy group, select it from the drop-down list.
 - To change the name of an existing policy, select it and then type over it.
 - To add a new policy group, select the empty row in the drop-down list and then type the name of the new group.
- Step 5** Click **OK**.
- Step 6** To assign a service class to this policy group, choose **Manage QoS** from the QoS drop-down menu in the toolbar.
-

Creating or Editing Service Class Policies

You can configure policies for undifferentiated traffic and for service classes.

- Step 1** Open the Manage QoS dialog box in one of two ways:
- Choose **Edit > Manage QoS**.
 - Choose **Manage QoS** from the QoS drop-down menu in the toolbar.
- Step 2** In the Service Class Policies area, click **New**, or select a service class row and click **Edit**.
- If creating a policy for undifferentiated traffic, select that option. If creating a policy for an existing service class, select it from the Service Class drop-down list.
 - (Optional) To apply this service class mapping to a group of interfaces, select from or enter the name in the Interface Policy Group drop-down list. You can enter a name that does not exist and then create the policy group for a set of interfaces. For further instructions, see [Creating or Editing Policy Groups for Interfaces, on page 250](#).
 - In the Normal Operation field, enter the percentage of bandwidth capacity that you do not want this interface (or group of interfaces) to exceed for this traffic or service class under normal conditions.
 - In the Worst-Case field, enter the percentage of bandwidth capacity that you do not want this interface (or group of interfaces) to exceed for this traffic or service under worst-case operating conditions.
 - Click **OK**. These values now appear in the Manage QoS dialog box where you can edit them as needed.
- Step 3** Click **OK** to save and exit.
-

What to do next*Figure 97: Manage QoS Dialog Box*

381181

Editing Interface Queues Properties (QoS Requirements)

The three parameters are priority, weight, and police limit.

-
- Step 1** In the Interface Queues table, right-click one or more queues and choose **Properties**.
- Step 2** In the Properties dialog box, change one or more fields to create the desired QoS requirement.
- Step 3** Click **OK**.
-



CHAPTER 14

BGP Simulation

This chapter describes how WAE Design models multi-AS networks and simulates basic BGP routing. WAE Design does not directly emulate BGP routing configurations, such as local prefs and MEDs. Rather, it provides a high-level modeling of typical peering policies, such as standard customer, transit, and settlement-free arrangements for service providers. This model lets you quickly and easily evaluate the effects of peering locations and basic policy variations.

Additionally, you can extend these high-level models to significantly more complex policy-based routing situations using *external endpoints* as demand sources and destinations. For information on demands and on external endpoints, see [Traffic Demand Modeling, on page 43](#) and [Advanced Routing with External Endpoints, on page 265](#).

This section contains the following topics:

- [Internal and External AS Types, on page 253](#)
- [Inter-AS Routing , on page 254](#)
- [BGP Simulations, on page 259](#)
- [Trimming ASes, on page 260](#)
- [Creating ASes, on page 261](#)
- [Associating Nodes with an AS, on page 262](#)
- [Editing AS Routing Policies, on page 262](#)
- [Assigning Names to ASNs, on page 263](#)

Internal and External AS Types

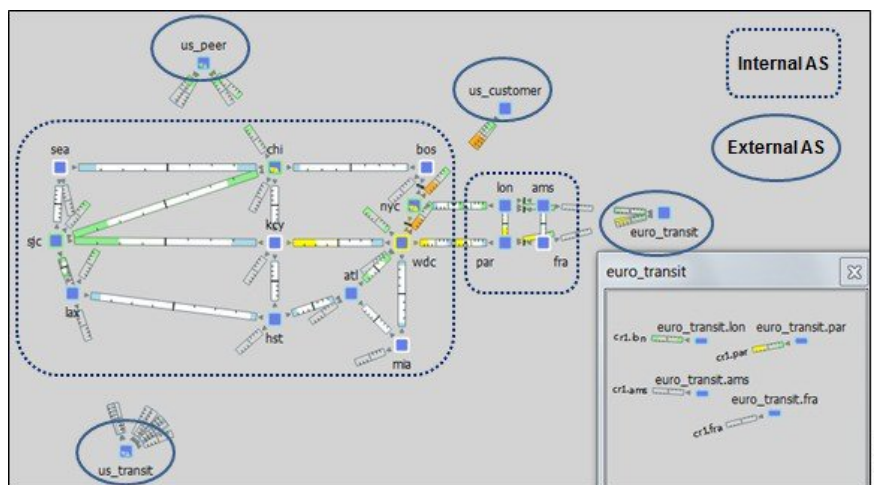
To model a multi-AS network, each node is assigned an AS, and each AS is defined as either *internal* or *external* . A typical multi-AS model in WAE Design consists of the following:

- A single internal AS representing the full topology of your network.
- Individual peering nodes of neighboring external ASes.
- Peering circuits connecting the internal AS to the nodes in the external ASes.

Generally, there are many external ASes in the plan file, but usually only one or a few internal ASes. All nodes in an external AS are typically placed in the same site, although you can place them in any site. [Figure 98: Example Multi-AS Model, on page 254](#) shows an example model with two internal ASes and four external ASes.

AS names and their types are defined in the AS Properties dialog box and listed in the AS table. Nodes are assigned to ASes in the node Properties dialog box.

Figure 98: Example Multi-AS Model

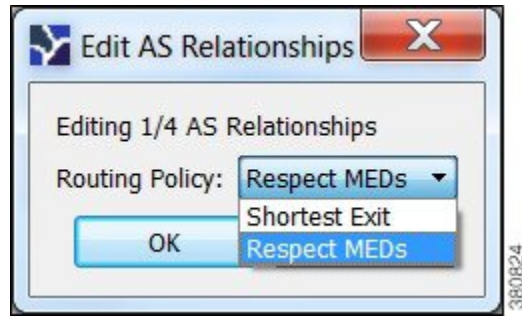


Inter-AS Routing

Route Selection Between Internal ASes

Demands routed within a single AS have a specified source node and destination node where traffic originates and terminates. Demands routed between two connecting internal ASes are specified in the same manner: with a source node in the first AS and a destination node in the second.

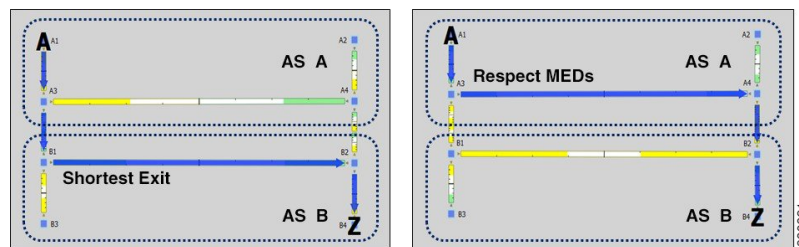
WAE Design routes within an AS, to and from the border exit point, are determined by the IGP protocols. The selection of border exit point is modeled by the Routing Policy, which is set to either Shortest Exit or Respect MEDs. This property is set in the Edit AS Relationships dialog box, which is accessed through the AS Properties dialog box.



- Shortest exit—The border exit node is selected, which is closest to the source node, within the IGP of the source AS. If there is a tie, the exit node with the lowest BGP ID is used.
- Respect MEDs—The border exit node is selected, which is closest to the destination node, within the IGP of the destination AS. If there is a tie, the exit node with the lowest BGP ID is chosen.

Figure 99: Route Selection Based on Routing Policy, on page 255 shows these modeled as two internal ASes: AS A and AS B. The same demand (A1 to B4) is routed using each of the two different routing policies.

Figure 99: Route Selection Based on Routing Policy



Routes Between External and Internal ASes

Table 18: Typical AS Routing Configurations, on page 255 lists typical routing configurations that can be constructed by applying different combinations of Routing Policies for traffic in both directions between two ASes.

- In a peer relationship, routing in both directions is Shortest Exit, which means each controls its own border exit points.
- For a customer relationship, the customer determines the border exit points for traffic in both directions.
- For a transit relationship, the transit AS provides paid transit to the internal AS, so the internal AS determines all border exit points.

Table 18: Typical AS Routing Configurations

Type	Policy to	Policy from
Peer	Shortest Exit	Shortest Exit
Customer	Respect MEDs	Shortest Exit

Type	Policy to	Policy from
Transit	Shortest Exit	Respect MEDs

Like traffic routed within an AS, traffic routed between ASes is represented by demands. However, for demands from and/or to external ASes, the external AS is defined as the source or destination of the demand. Optionally, the specific node in the external AS from which the traffic enters or exits the internal AS is also specified.

Failover between nodes in the external AS can be modeled. For example, if the traffic is sourced from an external AS and if the peering circuit from which traffic is entering the internal AS under normal operation fails, the traffic can enter the internal AS from a different interface or peering node in the same external AS. In the Demands table, the sources and destinations are represented as follows.

AS{<ASN>}:if{node_name|interface_name}

Example: AS{33287}:if{cr01.newyork.ny|POS3/7/0/0}

For more detailed information on demand sources and destinations, see [Traffic Demand Modeling](#).

The AS that controls the routing chooses which peering node to use. If the internal AS controls the routing, then because the topology of the internal AS is known, you can simulate the routing to the peering node. However, because WAE Design has limited knowledge of the external AS topology, if the external AS controls the routing, you cannot predict how traffic will be distributed among the exit points.

	From AS Type	To AS Type	Routing Policy	Routing Control
1	External	Internal	RespectMEDs	Internal
2	External	Internal	ShortestExit	External
3	Internal	External	RespectMEDs	External
4	Internal	External	ShortestExit	Internal

The AS that controls the routing is determined by the AS type, direction of the demand, and the Routing Policy property as described in [Table 19: Determining the AS that Controls the Routing, on page 256](#).

Table 19: Determining the AS that Controls the Routing

Direction	Routing Policy	AS with Routing Control
External AS to Internal AS (Ingress)	Respect MEDs	Internal
	Shortest Exit	External

Direction	Routing Policy	AS with Routing Control
Internal AS to External AS (Egress)	Respect MEDs	External
	Shortest Exit	Internal

Two ASes can be in one of four different routing relationships to one another, depending on which of the two routing policies is chosen in each direction ([Table 20: Effects of Routing Policy and Routing Control](#), on page 257).

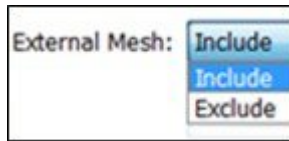
- If traffic is routed to an external AS when it has control and there is no knowledge of its topology, a set of demands is created from the source in the internal AS (or from another external AS), each with a destination set to one of the border nodes in the external AS. This way, any division of traffic between the exit points can be modeled.
- If traffic is routed to an external AS when an internal AS has control, a single demand is created from the source to the AS itself. WAE Design simulations determine the correct exit point for this single demand based on the source.
- If traffic is to be routed from an external AS when it has control, a demand is created from each node in the external AS to each node in the internal AS.
- If traffic is to be routed from an external AS when an internal AS has control, a demand is created to each node in the internal AS using the external AS as the source. The demand originates from one or multiple nodes in the external AS, depending on the topology and the metric cost to reach the destination node. For example, a single demand from an external AS to a specific node could be sourced from two different nodes in the external AS, each carrying 50% of the demand traffic.

Table 20: Effects of Routing Policy and Routing Control

Direction	Routing Policy	AS with Routing Control	Demand Source or Destination Endpoint in Remote AS	Number of Demands
External AS to Internal AS (Ingress)	Respect MEDs	Internal	Entire external AS	One only
	Shortest Exit	External	Border nodes	One for each node
Internal AS to External AS (Egress)	Respect MEDs	External	Border nodes	One for each node
	Shortest Exit	Internal	Entire external AS	One only

External Meshes

An external mesh consists of two or more external ASes with a Type property of External. An internal AS typically restricts advertisement of BGP routes for some external ASes to other external ASes. For example, destinations reachable through the transit network would not be advertised to a peer, or vice versa. In WAE Design, these restrictions are represented by the absence of demands between the two external ASes.



Each AS has a property called External Mesh, which WAE Design uses when inserting demand meshes into a plan. Demands are created for external ASes only if one or both ASes have External Mesh set to Include. If both ASes are set to Exclude, no demands are created for the external AS. For example, in [Figure 100: External Mesh Control, on page 258](#) the peer and transit ASes are both set to Exclude, so no demands are created between those ASes. All other external AS demands are included in the demand mesh. [Table 21: External Mesh Settings for Common AS Relationships, on page 258](#) shows the External Mesh settings for common AS relationships.

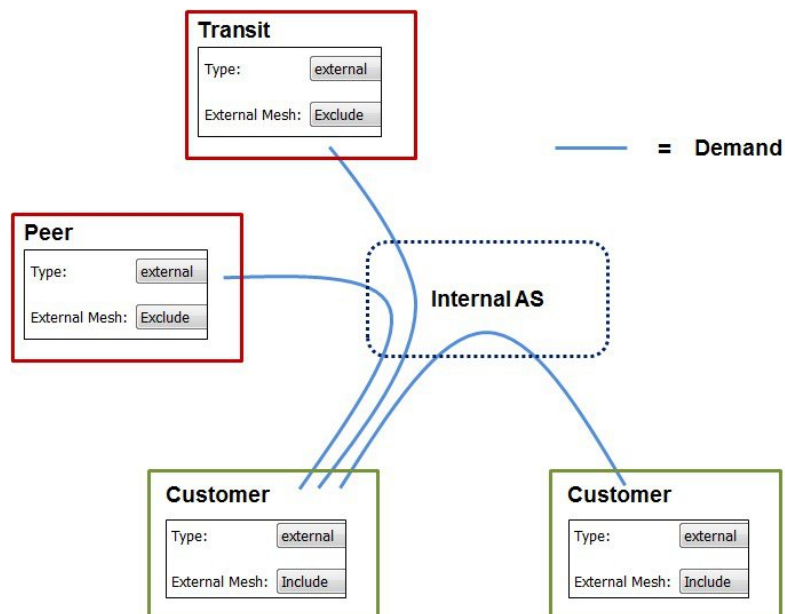
The External Mesh property is set in the AS Properties dialog box.

Table 21: External Mesh Settings for Common AS Relationships

Relationship	External Mesh Setting	Result
Peer	Exclude	Demands permitted to/from customers only
Customer	Include	Demands permitted to/from all external ASes
Transit	Exclude	Demands permitted to/from customers only

For internal ASes, the External Mesh property is ignored. More complex route advertisement policies cannot be represented by these simple External Mesh settings. In this case, demand mesh creation must be performed in several steps, possibly using a script and the `dmd_mesh_creator` CLI tool.

Figure 100: External Mesh Control



BGP Simulations

As with all WAE Design simulations, AS routing uses demands. An IP simulation for a particular failure scenario and traffic level performs these steps.

Step 1 Demands are routed using the established LSPs (if applicable) and using the specified BGP protocols given the specified failure scenarios.

Step 2 Interface utilizations are calculated from the demand traffic using the specified traffic level.

WAE Design allows routes to be calculated between selected nodes even if no demands are present. In this case, only the first step applies.

BGP demands do not failover between external ASes. That is, all traffic to or from an external AS behaves the same under peering failures to an external AS. You can change this default behavior using external endpoints to simulate specific external AS nodes where traffic goes in and out of the network, as well as set priorities so that if one traffic source or destination goes down, the traffic can still be sourced from or delivered to another external AS node.

BGP Routing Details

BGP Multihop

WAE automatically constructs BGP pseudonodes where necessary when BGP multihops are detected.

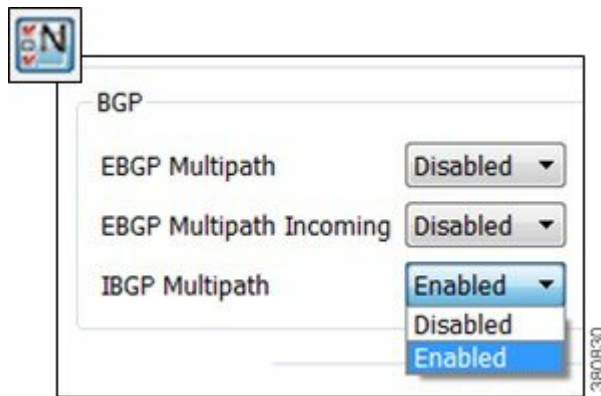
WAE Design models the nodes in external ASes that are directly connected, for example through eBGP, to nodes in internal ASes. One exception is that you can model BGP multihops by setting the node Type property to psn (pseudonode), such as might occur at a peering exchange. This pseudonode can represent the switch that connects a number of external AS nodes to the same internal AS node. In this instance, multiple external AS nodes are connected by circuits to a BGP psn node, and this node is connected to a node in the internal AS.



Note In all cases, eBGP multipaths across parallel border circuits is assumed.

BGP Load Balancing

BGP load balancing to an external AS uses eBGP multipaths or eBGP multihops. WAE Design models these two eBGP load balancing designs in the same manner, though in the GUI they are identified only as multipaths. BGP multipath options are disabled by default.



To set BGP multipath options globally, follow these steps:

-
- Step 1** Choose **Edit > Network Options** or click the **Network Options** icon in the Visualization toolbar.
- Step 2** Click the **Protocols** tab.
- Step 3** For each BGP multipath option that you want enabled, choose **Enabled**, and then click **OK**.
- **EBGP Multipath**—Turns on eBGP multipath within the internal ASes. Demand routings through the internal AS to an external AS are divided among external routes with equal-cost BGP exit routes.
 - **EBGP Multipath Incoming**—Turns on eBGP multipath in all external ASes. Demand routings from external ASes to an internal AS are divided among external routes with equal-cost BGP exit routes.
 - **IBGP Multipath**—Turns on iBGP multipath within the internal ASes. Demand routes through an internal AS to an external AS are divided among internal paths to equal-cost BGP exit routes.
-

BGP Next Hop

In networks, there are two common configurations for the BGP next-hop IGP metric used in the path selection. One is to set the next-hop self on the iBGP peers (next-hop self = on). The other is to configure IGP metrics on eBGP interfaces, and to inject the interface prefix into the IGP database by setting the interface to be a passive IGP interface (next-hop self = off).

WAE Design does not have an explicit next-hop self setting so it simulates paths as if next-hop self is off. That is, the IGP metric of the egress peering interface is included in the IGP distance to the peering router and is used in the iBGP path selection. However, next-hop self to an external AS can effectively be simulated by setting the metrics on all egress interfaces to that external AS to 0. (You can set the IGP metric in either the Interfaces Properties or Circuit Properties dialog box accessed through the context menu.)

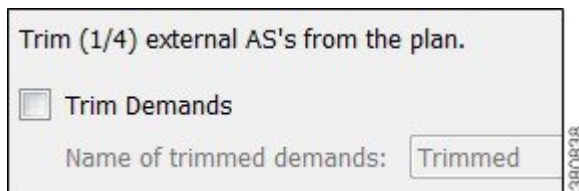
Trimming ASes

A plan file that uses BGP discovery typically has more external ASes in the plot than you are interested in viewing or simulating. Some ASes might have very low traffic levels and others might have only one circuit connecting into the internal AS. You can trim these unwanted ASes to simplify the network. Trimming removes the AS from the plan file, but retains the traffic in and out of the AS for simulation. That is, its interface traffic is still included in the total traffic of connected nodes.

Typically, plan files are trimmed before demands are created. However, you can optionally trim demands in a plan file. If any demand is sourced from a trimmed external AS, the demand source is moved to the first hop on the demand path in the remaining network. Likewise, if a demand has a destination that is a trimmed external AS, the demand destination is moved to the last hop in the remaining network. If two or more demands in the same service class are trimmed so that their resulting source and destination nodes are equal, these demands are aggregated into one demand with their traffic summed. Demands split by ECMPs are converted into multiple demands, each with traffic divided proportionally to the split ECMP.

Note that multicast demands are removed, and cannot be trimmed.

Step 1 Select one or more ASes to trim from the AS table.



For example, sort the ASes by traffic level and then select all ASes that are below the level of interest. Alternatively, filter the table for traffic using the less than operator (<) and the traffic level, and then select the filtered ASes.

Step 2 Right-click one of the selected ASes and choose **Trim**.

Step 3 (Optional) Select the option to trim demands.

By default, the moved demands are named “Trimmed,” though you can enter a different name.

Step 4 Click **Yes**.

Creating ASes

Follow these steps to create an empty AS. After creating the AS, you still need to associate nodes with it and create the relationship between this AS and others. See [Associating Nodes with an AS, on page 262](#) and [Editing AS Routing Policies, on page 262](#).

Step 1 Choose **Insert > AS**, or right-click in the empty plot and choose **New > AS**.

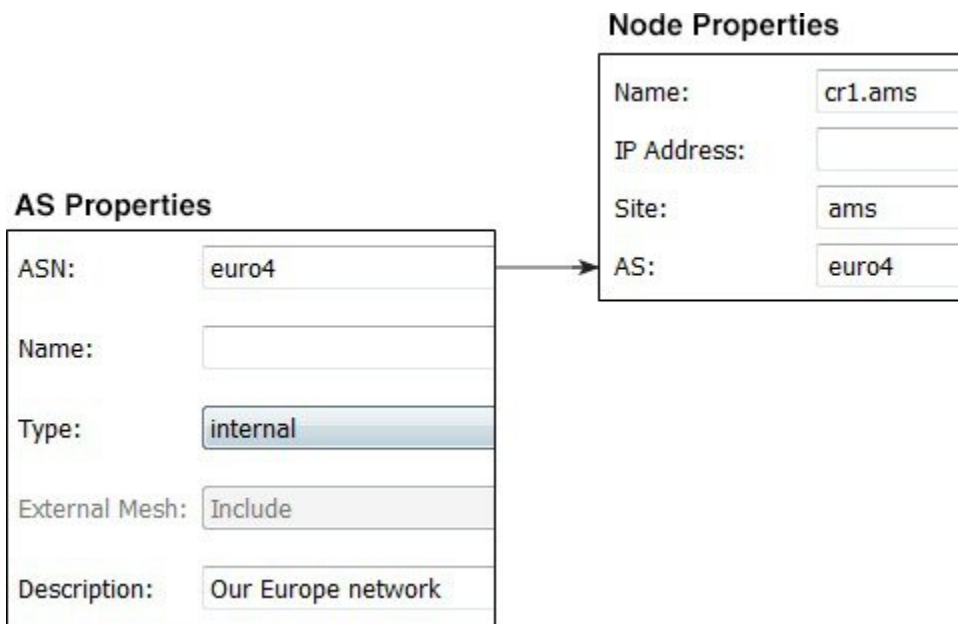
Step 2 Configure the AS properties:

- ASN—AS number, which is a text string that can be a number or name.
- Name—AS name.
- Type—Internal ASes have a full topology. External ASes have a collapsed topology with just border nodes and a virtual node.
- External Mesh—When creating a demand mesh, this option tells WAE Design whether to create external meshes. When one or both ASes are set to Include, WAE Design creates a mesh between the external ASes (default). If both are set to Exclude, no demands are created.
- Description—A text description of the AS.

- Step 3** To change the routing policy, click **Edit AS Relationships**. (The default is Respect MEDs.) For information on how these options affect simulation, see [Route Selection Between Internal ASes, on page 254](#).
- Step 4** Click **OK** to create an empty AS.

Associating Nodes with an AS

- Step 1** Choose one or more nodes, and double-click any one of them to open the node Properties dialog box.

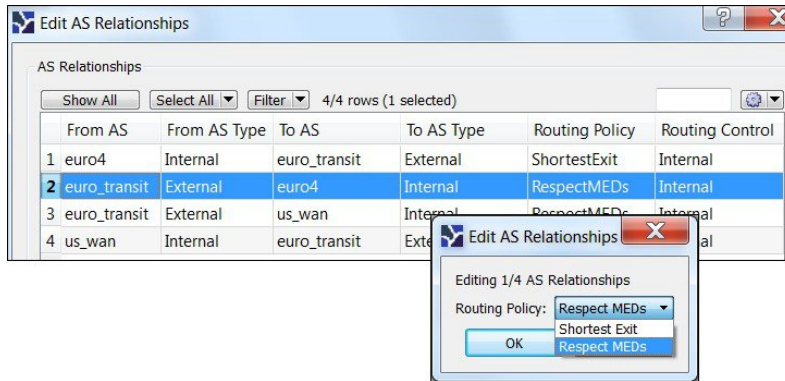


- Step 2** In the AS drop-down list, choose the AS to which you want to assign the nodes. Then click **OK**.

Editing AS Routing Policies

To create AS relationships, set the routing policy.

- Step 1** Choose an AS from the AS table.
- Step 2** Follow one of these methods to open the Edit AS Relationships dialog box:
- Choose **Edit > AS Relationships**.
 - Right-click an AS and choose **Edit AS Relationships**.
 - Right-click an AS and choose **Properties**. Click **Edit AS Relationships**.



- Step 3** Choose the AS pair that you want to configure. There is a separate line for each direction in the relationship so you can configure them independently.
- Step 4** Click **Edit**, set the Routing Policy to **Respected MEDs** (default) or **Shortest Exit**, and click **OK**.
- Step 5** Click **OK** again.

Assigning Names to ASNs

WAE Design can assign names to ASNs by looking them up in the \$CARIDEN_HOME/etc/ASNs.txt file. This file contains a list of all assigned ASNs with their names and descriptions.

- Step 1** Choose one or more ASes from the AS table. If no ASes are selected, names are assigned to all ASes in the plan.
- Step 2** Choose **Initializers > Assign Names to ASNs**, or right-click an AS and choose **Assign Names to ASNs**.
- Step 3** To assign names only if the ASN does not have one, check **Only those with blank Name fields**.
- Step 4** Click **OK**.



CHAPTER 15

Advanced Routing with External Endpoints

To model basic IGP routing, demands are sourced or destined for nodes within the topology. To model basic inter-AS routing, the sources and destinations are neighboring external ASes, or a combination of the external AS and the peering node in that AS. However, more complex routing situations require the use of *external endpoints* as the source or destination. External endpoints can contain multiple member nodes and ASes, and you can specify when traffic enters or exits from each of them individually. This allows you to simulate routing within and between ASes where multiple traffic entry and exit points are used simultaneously. You can also prioritize where the traffic fails over to other nodes and ASes.

There are numerous use cases both for IGP and inter-AS routing:

- Simulate content caching failovers for in-network source of demands that are backed up by another in-network source. If connectivity is lost to the first, traffic is sourced from the second.
- Simulate edge routing with a single entry point into the network edge and a specific failover point. Alternatively, you could model multiple entry points, depending on which is closest to the destination.
- Simulate complex BGP routing policies from a transit provider. For example, you can specify a transit entry location and failover location per destination.
- Simulate failover between peering ASes; for example, from one single-connection transit provider to another.

This section contains the following topics:

- [Routing with External Endpoints, on page 265](#)
- [States, on page 269](#)
- [In-Network Routing Examples, on page 270](#)
- [Intra-AS Routing Examples, on page 273](#)
- [Creating External Endpoints and Their Members, on page 276](#)

Routing with External Endpoints

An external endpoint is a WAE Design object that identifies specific entry (source) or exit (destination) points for demands. These are identified in the External Endpoints table by a name and an optional tag.

Each external endpoint consists of one or more members that are defined as nodes, external ASes, or a combination of an external AS and external node. By setting a demand's source or destination to an external endpoint, you can simulate traffic going from multiple sources to a single destination, from a single source

to multiple destinations, or multiple sources going to multiple destinations. Because of this flexibility, they are useful for specifying secondary entry and exit points in the event of failures.

External Endpoint Members

Each member is assigned properties that prioritize traffic entering and exiting that member, in what order to fail over to another member, and how to distribute traffic for members of equal priority. These properties are set when you create the member, and all members are listed in the External Endpoint Members table.

Member Endpoint: Node

Site: chi

Node: cr2.chi

Priority (1,2,3...) : 1

Equal Priority Routing: Fix Traffic

Traffic Balance (%): 50

380816

- Member Endpoint—Defines whether the member is a node, AS, or external node via an AS.
- Priority—The sequential order in which external endpoint members are used in the simulations should failures occur.
- Equal priority routing—If members have the same priority, this property identifies how the traffic is distributed.



Note Members within an external endpoint that have the same priority must either all be Shortest Path or none of them be Shortest Path.

- Shortest Path—Use the member that results in the use of shortest path between source and destination.

If a member is using a shortest path, the Traffic Balance (%) column in the External Endpoint Members table shows “na.”

- Fix Traffic—Set the traffic across members of equal priority as defined in the Traffic Balance (%) field.
- Deduce Traffic—Behaves the same as Fix Traffic in that it sets traffic across members of equal priority as defined in the Traffic Balance (%) field. However, upon running Demand Deduction, the Traffic Balance (%) field is updated based on the measured traffic in the network. Note that Demand Deduction only estimates the traffic balances for external endpoints with a priority that is in use in the current no-failure simulation. Thus, Deduce Traffic is usually set to Priority 1.

Routing Simulations



Note Although this section describes demands as being sourced from an external endpoint, the same methodology holds true if a demand's destination is an external endpoint.

If a demand's source is defined as an external endpoint, the following selection of external endpoint members ensues.

Step 1 Members with the highest priority (lowest number) are used as the demand's source. For example, if the external endpoint has two members with a priority of 1, the demand is sourced from both members provided they are available.

If one or more of the members are not available, the traffic from the unavailable members is evenly redistributed to the other top priority members.

Step 2 If none of the top priority members are available to source the traffic but there are next-priority members available, Step 1 is repeated for the next priority external endpoint members. Only if all members with the same priority fail does the traffic get routed according to the next priority in the sequence.

Note that if a failure occurs that does not affect the external endpoint member's ability to send or receive traffic, then traffic is rerouted as usual without a need to use the additional members.

If an external endpoint member is an external AS, with or without a node specified, then the routing from or to that member is determined by the BGP routing policy determined by the AS relationships. The distribution of traffic between external endpoints with the same priority is the same as for node members.

Traffic Distribution

The traffic distribution through these demands is based on the Equal Priority Routing property, and if applicable, the Traffic Balance (%) property used to define the external endpoint members.

- If there is only one member and it is defined as Shortest Path, the demand takes the shortest path as defined by the IGP metrics.

Of the routable demands, if the Traffic Balance (%) values are all empty, the traffic is routed and equally load balanced across the demands with the shortest IGP paths. Note in the case of multiple internal ASes, the shortest IGP route is the shortest route in the first AS the demand enters.

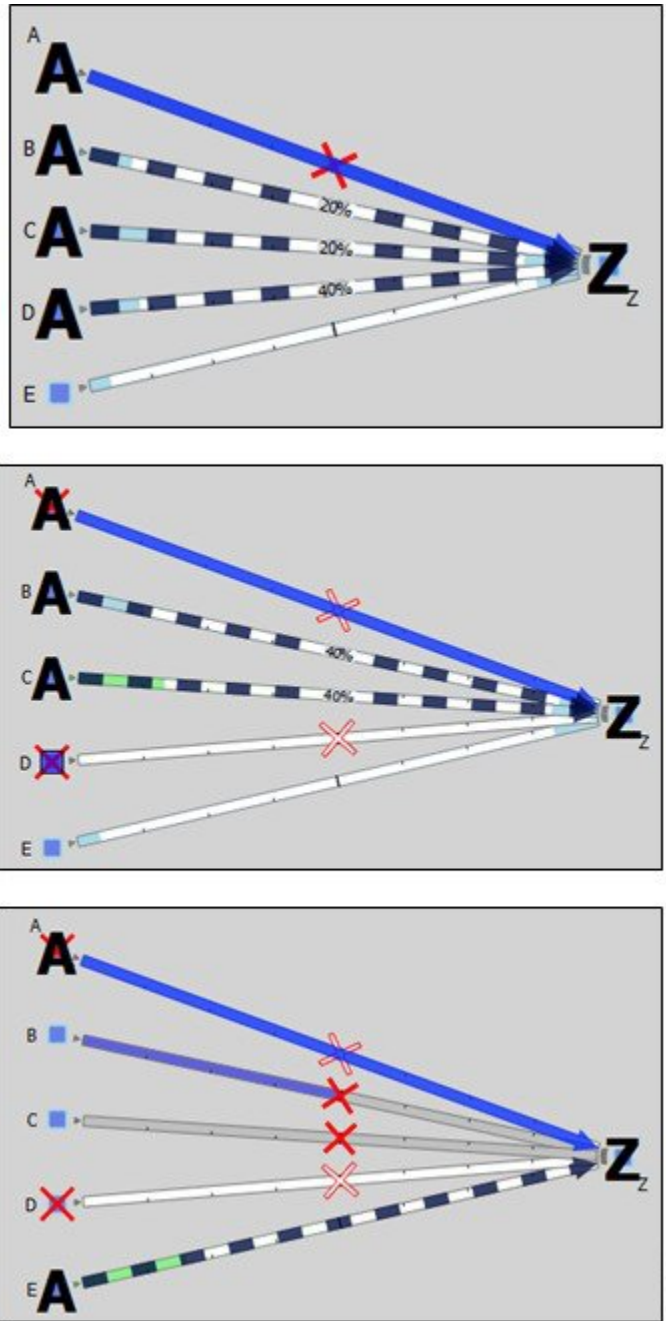
- If multiple members of the same priority are set to Shortest Path, the demand takes the path with the shortest IGP path. If all interfaces between the source members and the destination have the same shortest IGP paths, then the traffic is load balanced equally across them.
- If one or more members of the same priority have their Equal Priority Routing property set to Fix Traffic or Deduce Traffic, the demand traffic is split according to each member's Traffic Balance (%) value.
 - If the Traffic Balance percentages across sources with the same priority sum to less than 100%, the overall demand traffic is decreased to that percentage.
 - If an external endpoint member of the same priority fails, the traffic on the remaining members increases in proportion, so that the same amount of traffic is still routed.

Example: Node A failed. Nodes B, C, and D each have a priority of 2 and are each a Fix Traffic type. Their traffic balances are 20%, 20%, and 40%, respectively. The demand has 1000 Mbps of traffic.

Member	Priority	Type	Traffic Balance (%)
A	1	ShortestPath	na
C	2	FixTraffic	20.00
B	2	FixTraffic	20.00
D	2	FixTraffic	40.00
E	3	ShortestPath	na

- Because Node A failed, the demand routes 200 Mbps traffic through node B, 200 Mbps through C, and 400 Mbps through D, totaling 800 Mbps ([Figure 101: Example Effect of Failures When Using External Endpoints, on page 269](#)).
- If Node D fails, the demand routes 400 Mbps traffic through Node B and 400 through C. If Node B fails too, the entire 800 Mbps is routed through C ([Figure 101: Example Effect of Failures When Using External Endpoints, on page 269](#)).
- If all three priority 2 members fail, 1000 Mbps is routed through node E, which is the priority 3 member ([Figure 101: Example Effect of Failures When Using External Endpoints, on page 269](#)).

Figure 101: Example Effect of Failures When Using External Endpoints







States

External endpoint members can be failed, set to an inactive state, and included in Simulation Analysis to calculate a worst-case failure analysis. Additionally, they can be included in SRLGs, which in turn can also be failed, set to inactive, and included in Simulation Analysis. [Table 22: L1 Link Failure and Inactive States](#), on page 270 shows the icons used for these states.

If the external endpoint member that you fail or set to inactive is an AS, entry through any of its nodes is not possible. The same is true if an SRLG is failed or set to inactive and if it contains an external endpoint member that is an AS.

Table 22: L1 Link Failure and Inactive States

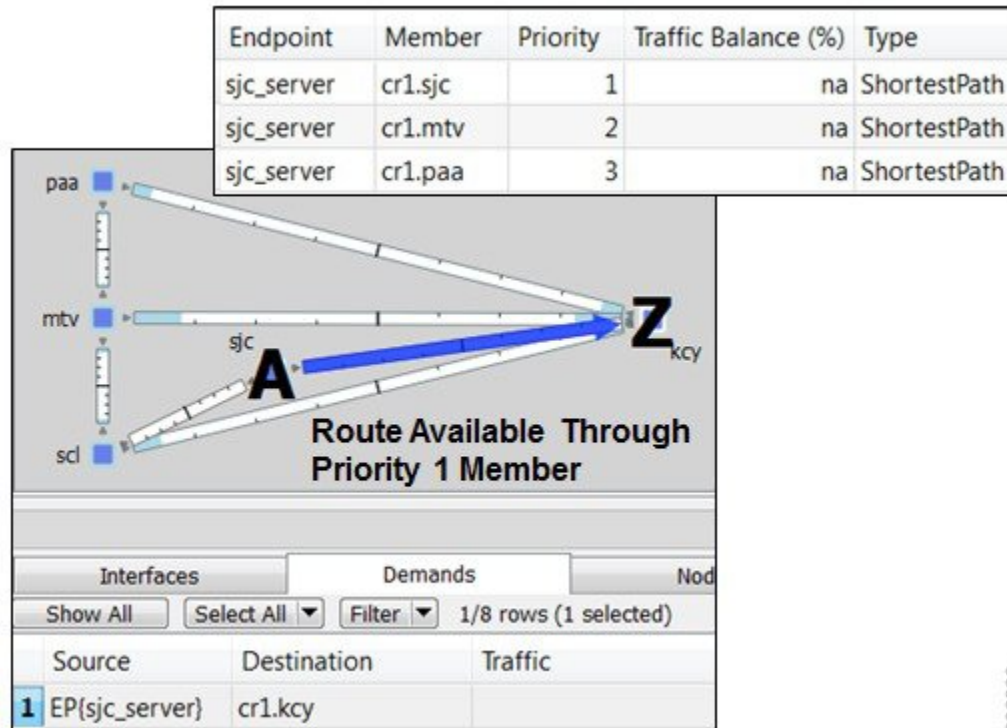
External Endpoint Member	Description
If the external endpoint member is a node, this icon appears under that node in the plot. If the member is an external AS or an external AS via a node, this icon appears under all nodes in the AS.	
	Failed external endpoint member.
	External endpoint member is not operational because it is contained in a failed SRLG.
	External endpoint member is not active.
	External endpoint member is not operational because it is contained in an SRLG that is not active.

In-Network Routing Examples

Content from Multiple Data Centers

This example demonstrates servers that cache content for customers in Kansas City (cr1.kcy). The primary server is located in San Jose (cr1.sjc), with backup servers in Mountain View (cr1.mtv) and Palo Alto (cr1.paa). Each of the servers is a member of an external endpoint (sjc_server) that is used as the source of the demand to cr1.kcy.

Figure 102:



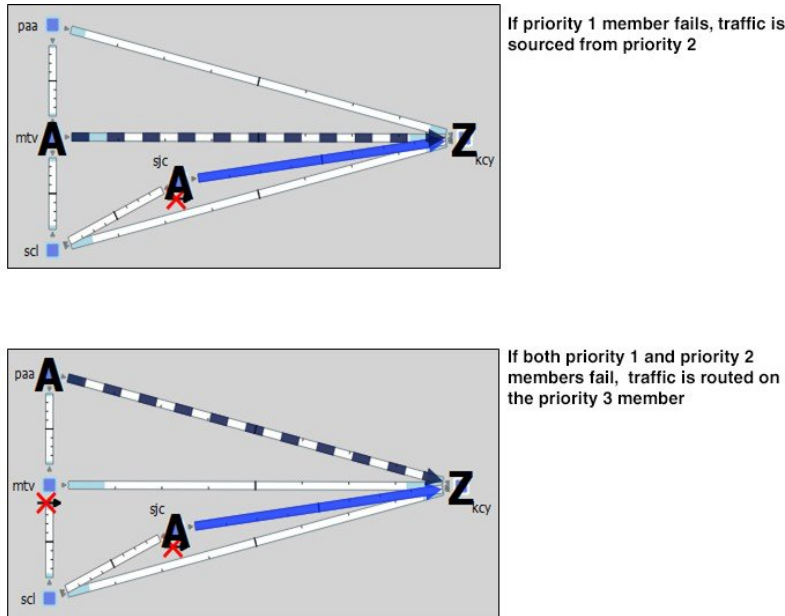
383288

- Member cr1.sjc has a priority of 1, and it takes the shortest path.
- Member cr1.mtv has a priority of 2, and it takes the shortest path.
- Member cr1.paa has a priority of 3, and it takes the shortest path.

Figure 103: Example Failover for Multiple Data Centers, on page 272 shows how the failure scenario works.

- When the priority 1 server node fails (cr1.sjc), content continues to be delivered using cr1.mtv as the source.
- If cr1.mtv also fails, cr1.paa sources the traffic to cr1.kcy.

Figure 103: Example Failover for Multiple Data Centers



380796

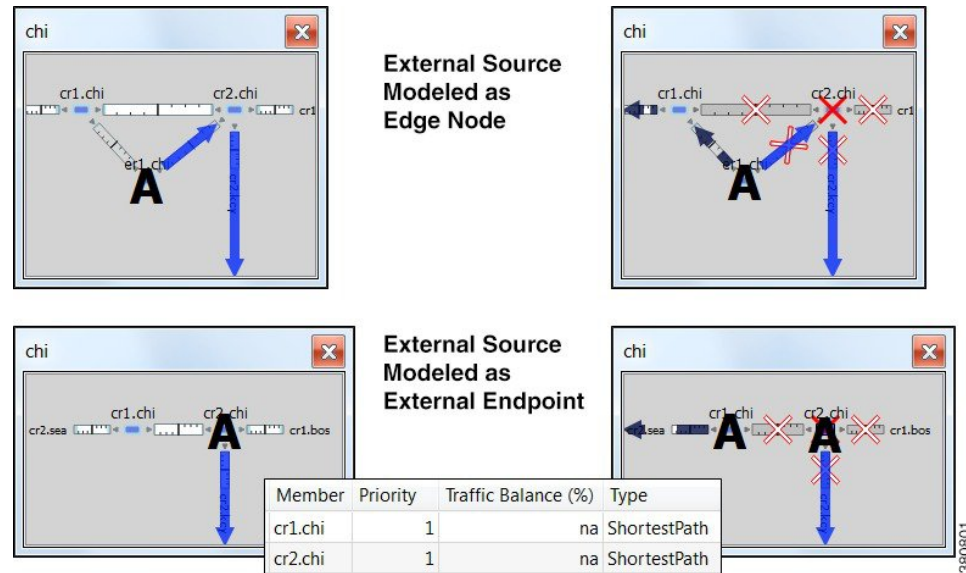
External Source

One method of modeling an external demand source with multiple entry points is to construct an edge node, connecting it to these entry points, and giving the interfaces equal metrics. Then source the demand from this edge node.

Alternatively, you can construct these multiple entry points using a single external endpoint as the demand source, thus reducing the complexity of the topology.

In this example, a demand has an external source with two entry points that come into the network at Chicago. [External Source, on page 272](#) demonstrates two ways to model this source. One uses additional topology (edge node) to source the demand. The other uses a single external endpoint that has two members (cr1.chi and cr2.chi), both set to priority 1 with shortest path traffic.

Figure 104: Example External Demand Source



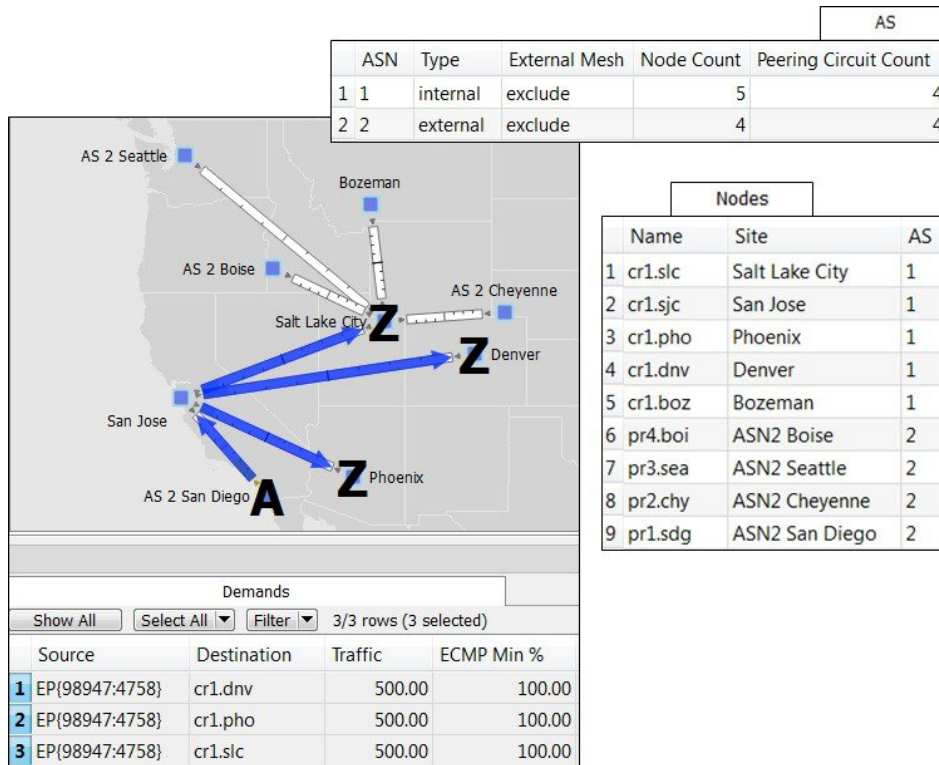
Intra-AS Routing Examples

Failover to Nodes in External ASes

This example demonstrates a content provider (Acme) who sources most incoming Internet traffic through a transit provider, and this traffic is then distributed to Acme's customers. Acme has identified specific policies for each market using communities, and these policies determine, per location, which transit provider peering points are used for primary routing and failover. The network is set up as follows ([Figure 105: Example Transit Peer AS 2 San Diego Sends Traffic to Customers in an Internal AS, on page 274](#)):

- The San Diego, Boise, Cheyenne, and Seattle nodes each belong to the same transit provider, AS 2.
- The remaining nodes belong to Acme's internal AS (AS 1), and these are Acme's customers.
- The San Diego node (pr1.sdg) in AS 2 is sending traffic to three key customers located in Phoenix (cr1.pho), Salt Lake City (cr1.slc), and Denver (cr1.dnv).

Figure 105: Example Transit Peer AS 2 San Diego Sends Traffic to Customers in an Internal AS



In WAE Design, Acme uses an external endpoint named after community 98947:4758. It contains the following members, all of which are in the same AS 2 transit provider:

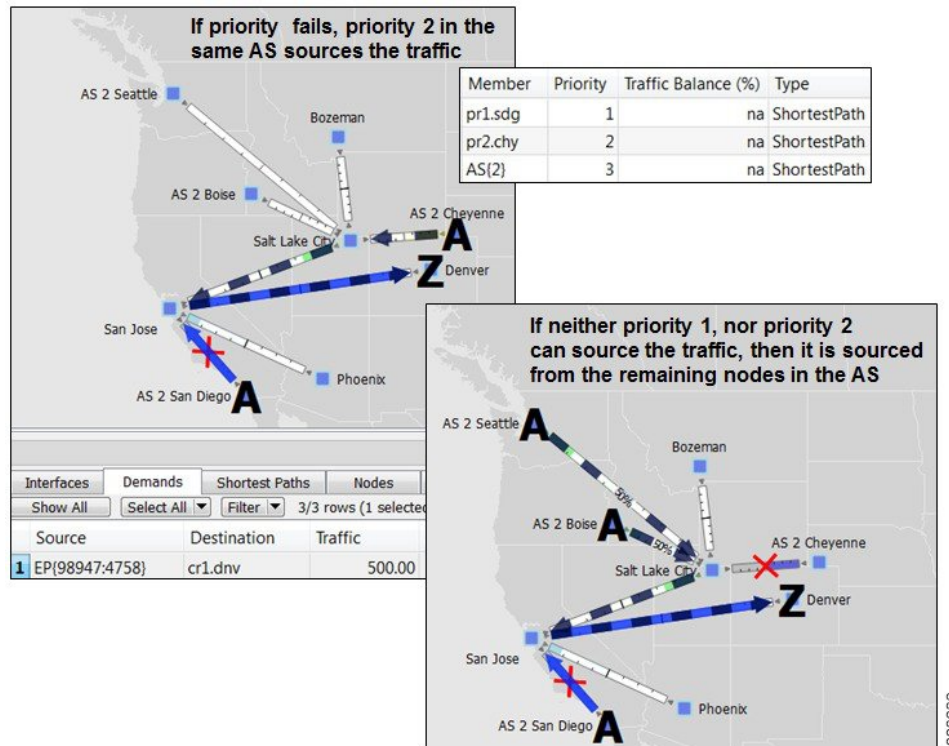
- pr1.sdg set to priority 1 with shortest path traffic
- pr2.chy set to priority 2 with shortest path traffic
- AS 2, which means the remaining nodes in the AS, set to priority 3 with shortest path traffic

As [Figure 105: Example Transit Peer AS 2 San Diego Sends Traffic to Customers in an Internal AS, on page 274](#) shows, multiple demands are created using this external endpoint (98947:4758) as the source, and each sending traffic to a different customer destination. [Figure 106: Example Failover Within the Same External AS, on page 275](#) focuses on the traffic destined for Denver (cr1.dnv), showing that the failover scenario behaves as follows:

- If a failure prevents the peer pr1.sdg node from sending traffic to cr1.pho, a failover occurs to pr2.chy within that same peer to avoid a service disruption.
- If failures prevent pr1.sdg or pr2.chy from sending traffic to cr1.dnv, the remaining nodes in the AS source the traffic. Although this is set to use the shortest path, because there are multiple paths, the traffic is distributed evenly.

The same result could have been achieved by setting the priority 1 to an interface going from San Jose to pr1.sdg and by setting the priority 2 to an interface going from Salt Lake City to pr2.chy.

Figure 106: Example Failover Within the Same External AS



Failover from an External AS to Another

In this example, a small ISP (AcmeNet) buys transit from two other service providers. AcmeNet peers with AS 2 in Phoenix and AS 3 in Denver. AcmeNet has a customer base in Phoenix (cr1.pho). The customer has a preferred access to the Internet through San Francisco (pr1.sfo) in AS 2 (Figure 107: Example Transit Peer ASN2 San Francisco Sends Traffic to Customers in an Internal AS, on page 276), and the secondary access is through Denver (pr2.dnv) in AS 3 (Figure 108: Example Failover to a Different External AS, on page 276). In WAE Design, AcmeNet simulates this network by creating an external endpoint (Internet) containing two different external ASes as members.

- AS 2 is priority 1 with shortest path
- AS 3 is priority 2 with shortest path

The demand is sourced from this external endpoint named Internet.

Figure 107: Example Transit Peer ASN2 San Francisco Sends Traffic to Customers in an Internal AS

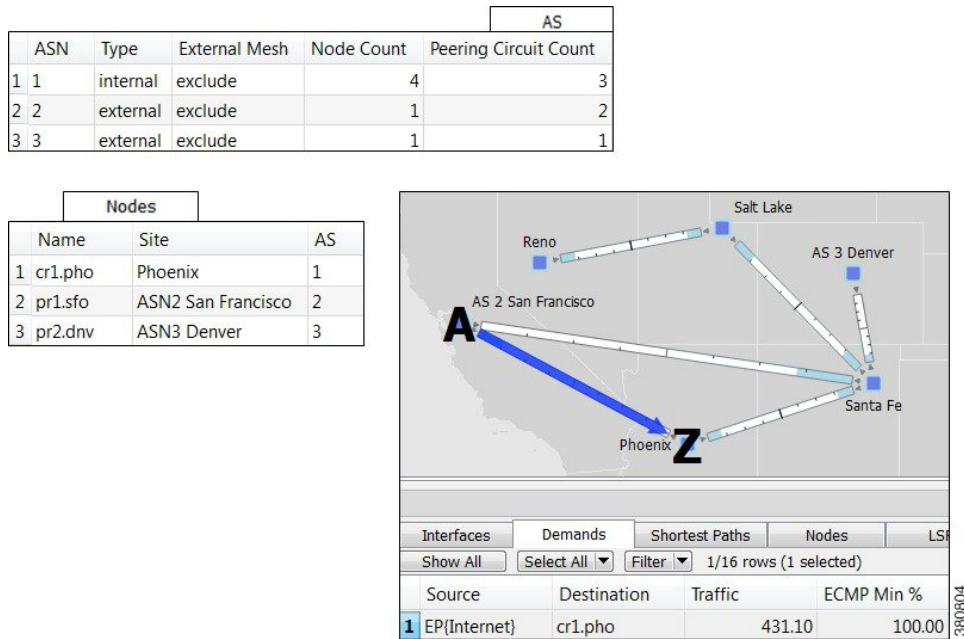
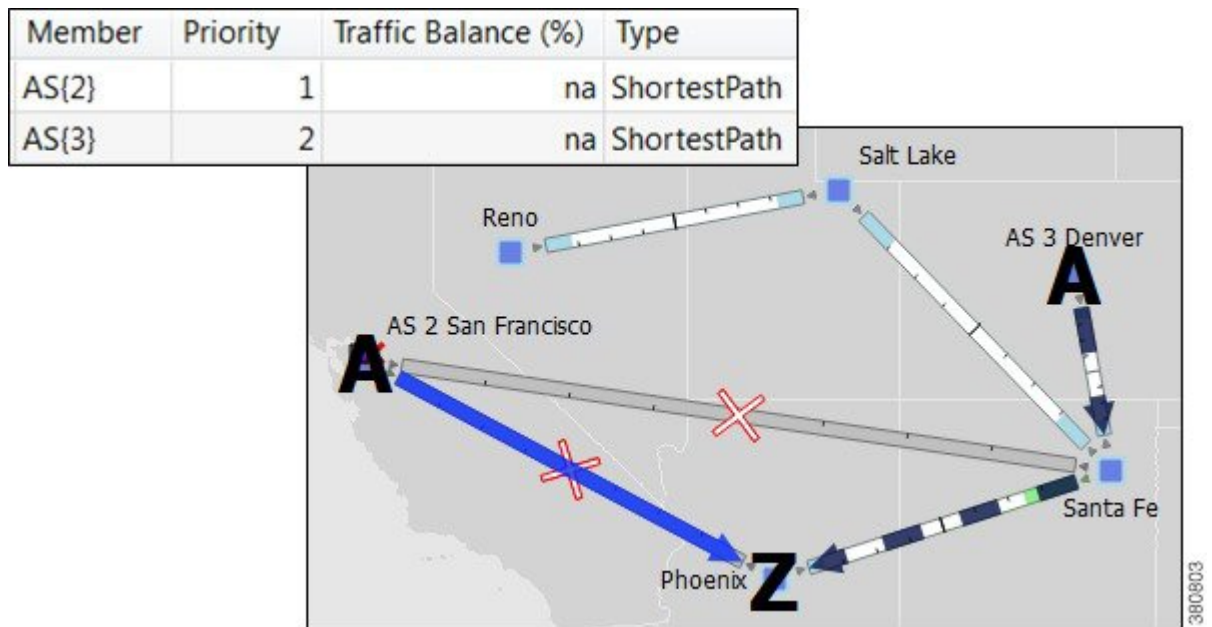


Figure 108: Example Failover to a Different External AS



Creating External Endpoints and Their Members

The recommended method of creating external endpoint members is to do so while creating the associated external endpoint, as follows.

Step 1 Right-click in an empty area and choose **New > Demands > External Endpoint**, or choose **Insert > Demands > External Endpoint**. The External Endpoint Properties dialog box opens.

To edit an existing external endpoint, double-click it from the External Endpoints table.

Step 2 Enter a unique external endpoint name in the Name field.

Step 3 To add a new member, click **New**.

To edit an existing member, select it from the Members table and click **Edit**.

An external endpoint member dialog box opens.

- a) Specify the member type as **Node**, **Interface**, or **External AS**.
 - For nodes, choose the site and node name.
 - For interfaces, choose the site, node, and interface. Using an interface lets you specify the exact interface on which the demand traffic is going into or out of a node.
 - For ASes, choose the AS name, and select either the “unspecified” option or node name within the AS. The “unspecified” option routes traffic evenly throughout the entire AS.
- b) Enter a priority number to determine the sequence in which the member is used with other members in this external endpoint. Multiple members can have the same priority.
- c) Select whether to use a shortest path route, or to direct a specified percentage of traffic across members that have the same priority (Fix Traffic or Deduce Traffic).
- d) If balancing traffic, enter the percentage of traffic this member should route.

If you set Deduce Traffic, this value is updated by Demand Deduction provided there are no members with a higher priority that are in use under normal (non-failure) operation.
- e) Click **OK**.

Step 4 Click **OK** in the External Endpoint Properties dialog box.



CHAPTER 16

VPN Simulation

The WAE Design virtual private network (VPN) model is a representation of a virtual subnetwork within the plan file. Viewing and simulating VPN within WAE Design facilitates many network tasks and can answer questions, such as:

- Which VPNs are on my network? Where and how are they configured?
- Which VPNs are using congested interfaces?
- Which VPNs will experience congestion under any of a given list of failure scenarios?
- Which failures scenarios cause the worst-case congestion or latency for a VPN?

There are many varieties of VPNs. For example, there are Layer 2 (L2) VPNs and Layer 3 (L3) VPNs, each with different categories within it, and there are vendor-specific VPN implementations. Each VPN type has its own specific configuration and terminology. The WAE Design VPN model supports a number of these VPN types based on either route-target or full-mesh connectivity.

This section contains the following topics:

- [VPN Model, on page 280](#)
- [VPNs, on page 281](#)
- [VPN Nodes, on page 283](#)
- [Layer 3 VPN Example, on page 284](#)
- [VPN Simulation Analysis, on page 287](#)
- [Creating VPN Nodes, on page 288](#)
- [Creating VPNs, on page 288](#)
- [Adding VPN Nodes to VPNs, on page 290](#)

VPN Model

VPN Objects

Object	Description	Examples
VPNs	A set of VPN nodes that can exchange data with each other.	<ul style="list-style-type: none"> Layer 2 VPN: The VPN represents an individual VPLS containing virtual switch interfaces (VSIs). Layer 3 VPN: The VPN represents sets of VRFs associated with a set of VPN nodes that forward traffic between themselves. Often, this set of VRFs signifies a single customer or service.
VPN nodes	Connection points in a VPN. They exist on standard nodes, and each node can contain multiple VPN nodes. A VPN node can be in only one VPN.	<ul style="list-style-type: none"> Layer 2 VPN: The VPN node represents the VSIs configured on each router. Layer 3 VPN: The VPN node represents the VRF instances configured on each router.

VPN Topology and Connectivity

WAE Design VPN topology route connections are established through route targets (RTs) or through a full mesh of VPN nodes. The Connectivity property is set in the VPN Properties dialog box.



Knowing a VPN's topology and connectivity lets WAE Design calculate which demands between VPN nodes carry traffic for a particular VPN, and thus which interfaces carry traffic for that VPN. In turn, WAE Design can calculate the vulnerability of a VPN to certain failure and congestion scenarios.

A demand is associated with a VPN, meaning it carries traffic for that VPN, if the following is true:

- The two VPN nodes are in the same VPN.
- The demand is in the same service class as the VPN.
- Only for VPNs with RT connectivity, the RT Export property of one VPN node must match the RT Import property of another VPN node.

Once demands are associated with the VPN, this configuration simulates the associated access circuits exchanging traffic as if they were on the same LAN.

Note that a demand associated with a VPN can additionally contain other traffic that is for that VPN.

Connectivity	Description
Full mesh	Full-mesh connectivity is a complete mesh of connections between VPN nodes in a VPN so they can all communicate with one another. This connectivity is typical in a VPLS, where all VSIs identify one another based on a common AGI.
Route targets	<p>Route targets model the more complex connectivity used in Layer 3 VPNs, such as hub-and-spoke networks. Here, the VRFs exchange data with one another based on the matching of RT Export and RT Import properties set for each VPN node.</p> <p>Having an import/export pair does not create bidirectional communication. Rather, traffic flows in the opposite direction of the routed advertisements. For example, if node A's RT Import matches node B's RT Export, traffic can flow from node A to B.</p> <p>For traffic to flow from node B back to node A, node B must have an RT Import that matches an RT Export of node A. This combination of matching imported and exported RTs defines which VPN nodes can exchange data. The VPN name identifies the VPN itself.</p> <div style="text-align: center;"> <p>VPN Node A VPN Node B</p> <p>RT Import: 2:1; 3:1; 100:100 RT Import: 1:1</p> <p>RT Export: 1:1 RT Export: 2:1</p> <p style="text-align: right; font-size: small;">381261</p> </div>

VPNs

Each VPN consists of a set of VPN nodes that can exchange data within it. VPNs have three key properties that uniquely identify them and define how the traffic within them is routed:

- Name—Unique name of the VPN.
- Type—Type of VPN. You can choose from the defaults (VPWS, VPLS, or L3VPN), or you can enter a string value to create a new one. Once entered, the new VPN type appears in the drop-down list and is available for other VPNs and VPN nodes.
- Connectivity—Determines how WAE Design calculates connectivity and associated demands for VPNs:
 - Full Mesh—Connectivity is between all nodes in the VPN. WAE Design ignores the RT Import and RT Export properties of the VPN nodes.
 - RT—Connectivity is based on the RT Import and RT Export properties of its VPN nodes.

VPNs Table

The VPNs table lists the VPN properties, its associated service class, traffic, and the number of VPN nodes within that VPN ([Table 23: VPNs Table Columns for Normal Operation](#), on page 282). For information on QoS measurements, see [Quality of Service Simulation](#), on page 239. For information on the Worst-Case columns not listed here, see [Table 25: Simulation Analysis Columns in the VPNs Table](#), on page 287.



Note Because the traffic and QoS calculations are based on all interfaces within the VPN for the service class specified for that VPN, the plot view might differ from the table. For example, the plot view could show Internet traffic while a VPN carrying voice traffic is selected.

Table 23: VPNs Table Columns for Normal Operation

Columns	Description
All traffic and QoS violations are based on traffic carried on all interfaces used by the VPN for the service class defined for that VPN.	
Service Class	Service class associated with this VPN. All values within the table are associated with this service class.
Num Nodes	Number of VPN nodes in this VPN.
Util Meas	The maximum measured utilization of all interfaces used by this VPN.
Util Sim	The maximum simulated utilization of all interfaces used by this VPN.
Total Src Traff Meas	Total amount of measured source traffic on this VPN.
Total Dest Traff Meas	Total amount of measured destination traffic on this VPN.
QoS Violation Sim	Maximum QoS violation under normal operations for all simulated traffic for all interfaces used by this VPN. If the number is positive, there is a violation.
QoS Violation Sim (%)	QoS violation as a percent of the total simulated interface capacity.
QoS Violation Meas	Maximum QoS violation under normal operations for all measured traffic for all interfaces used by this VPN. If the number is positive, there is a violation.
QoS Violation Meas (%)	QoS violation as a percent of the total measured interface capacity.
Latency	Maximum latency of all demands used by this VPN.
Tags	User-defined identifiers that makes it easy to group VPNs.

VPNs are not selectable from the network plot; you can only select and filter to VPNs through tables. When selected, all VPN nodes within the VPN are highlighted in the plot ([Figure 109: VPN Nodes Within a VPN, on page 284](#)).

Identifying Interfaces Used by VPNs

To view which interfaces are associated with a VPN, right-click a VPN in the VPNs table and choose **Filter to Interfaces**. This is useful for viewing the VPN topology in the network plot. If you then choose all of these filtered interfaces, you can see the VPN outlined in the network plot.

To view which VPNs are associated with an interface, right-click an interface in the Interfaces table and choose **Filter to VPNs**. This is useful for determining which VPNs are affected if a circuit fails or goes down for maintenance.



Note Utilization measurements might be different between the tables because the VPN table calculates measurements only for the service class associated with that VPN.

VPN Nodes

VPN nodes are defined by properties that determine which VPNs the nodes belong to and how the demands are routed. The following are required properties:

- **Node**—Name of the node on which the VPN node resides. This node name corresponds with one in the Nodes table.
- **Type**—The type of VPN. You can choose from the defaults (VPWS, VPLS, or L3VPN), or you can enter a string value to create a new one. Once entered, the new VPN type appears in the drop-down list and is available for other VPN nodes and VPNs.
- **Name**—Name of the VPN node.
- **VPN**—Name of the VPN in which this VPN node resides. The drop-down lists shows existing VPNs of the same type set in the Type field. You can create a VPN node without setting its VPN, but without it, the VPN node is not included in simulations as a member of any VPN.

To simulate RT connectivity, you must set the VPN Connectivity property to RT and then set the RT Import and RT Export properties on the individual VPN nodes within it.

- **RT Import and RT Export**—The pairing of RT values identifies which VPN nodes connect with each other. For more information, see [VPN Topology and Connectivity, on page 280](#).
- **(Optional) RD**—Route distinguisher (RD) uniquely identifies routes within a VRF as belonging to one VPN or another, thus enabling duplicate routes to be unique within a global routing table.

VPN Nodes Table

The VPN Nodes table lists the VPN node properties, as well as columns that identify the VPN nodes' relationship within the VPN and its traffic ([Table 24: VPN Nodes Table, on page 283](#)).

Table 24: VPN Nodes Table

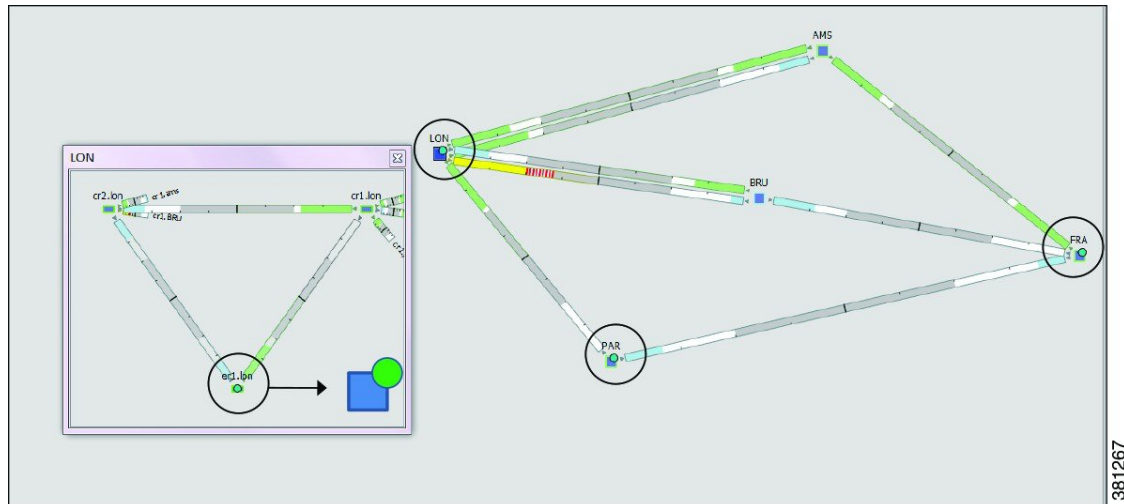
Columns	Description
Total Connect	Number of VPN nodes that are connected to this VPN node as defined by the RT Import and RT Export pairings. These may or may not be in the same VPN.
VPN Connect	Number of VPN nodes that are connected to this VPN node and are in the same VPN as defined by the VPN column.

Columns	Description
Num VPN Nodes	Number of nodes in the VPN that this VPN node belongs to as defined by the VPN column. This value is “na” if the VPN node does not belong to a VPN.
Src Traff Meas	Total amount of measured traffic entering the VPN at this node (source traffic).
Dest Traff Meas	Total amount of measured traffic leaving the VPN at this node (destination traffic).
Tags	User-defined identifier that makes it easy to group VPN nodes into a single VPN. If you give a VPN node a tag, when you create a VPN later, you can identify its VPN nodes using tags.

VPN nodes are not selectable from the network plot; you can only select and filter to them through tables.

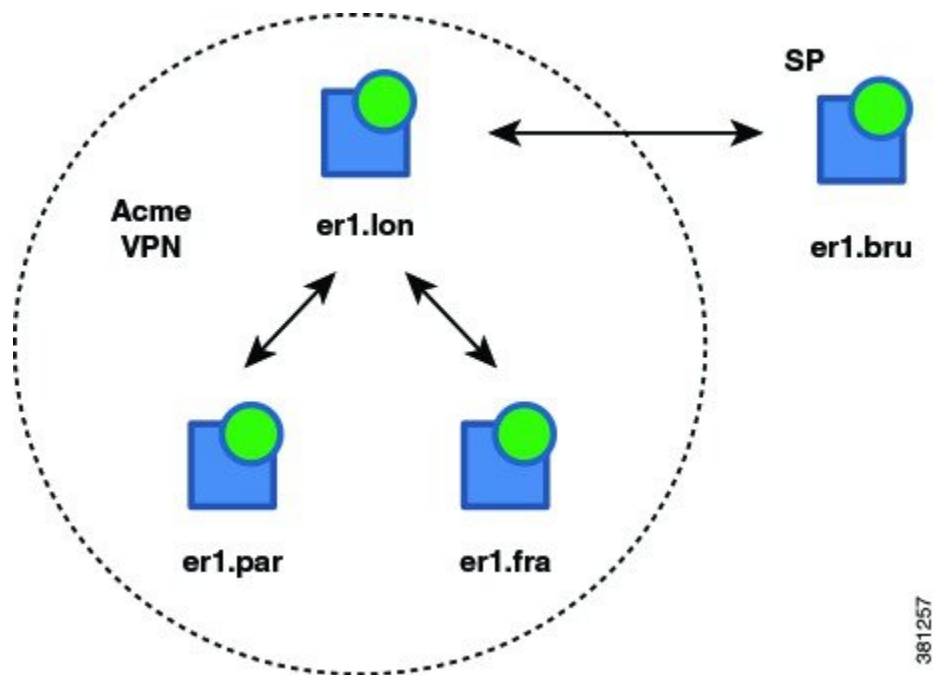
Once selected from the VPN Nodes or VPNs tables, the associated site and the nodes within that site appear with a green circle on it (Figure 109: VPN Nodes Within a VPN, on page 284).

Figure 109: VPN Nodes Within a VPN



Layer 3 VPN Example

This example illustrates a scenario where the Acme manufacturing company has three offices, but permits the two branch (er1.par and er1.fra) offices to exchange data only with headquarters (er1.lon).



381257

Additionally, headquarters communicates with an SP VPN node (er1.bru) that is not in the Acme VPN. [Figure 110: Example RT Connectivity and Acme VPN Footprint, on page 286](#) shows the footprint of the Acme VPN and the RTs set for all VPN nodes in this example.

- The VPN is named Acme, and it is set to a Connectivity of RT and a Type of L3VPN.
- In turn, each branch office is set to the Acme VPN, with a Type of L3VPN.
- To exchange data with two other VPN nodes in the Acme VPN, headquarters (er1.lon) imports the offices' exported route targets of 2:1 (er1.par) and 3:1 (er1.fra).
- In turn, headquarters (er1.lon) exports a route target of 1:1.

All three of these other VPN nodes import it (both offices and the SP VPN node).

Because the SP VPN node (er1.bru) is not in the Acme VPN, its communication with er1.lon is not within the context of that VPN.

Acme VPN	Acme VPN Nodes	SP VPN Node
Name: Acme	Type: L3VPN	Type: L3VPN
Type: L3VPN	Name: Acme_VRF	Name: Management
Connectivity: RT	VPN: Acme	VPN:

The VPN footprint in [Figure 110: Example RT Connectivity and Acme VPN Footprint, on page 286](#) shows that if the circuit between er1.fra and er1.bru became congested or failed, the VPN would be impacted. However, a failure of the circuit between the two branch offices would not impact it. This failure is illustrated

in [Figure 111: Example Failure Between Branch Offices in the Acme VPN](#), on page 286, which shows that none of the demands associated with the VPN are rerouted.

Figure 110: Example RT Connectivity and Acme VPN Footprint

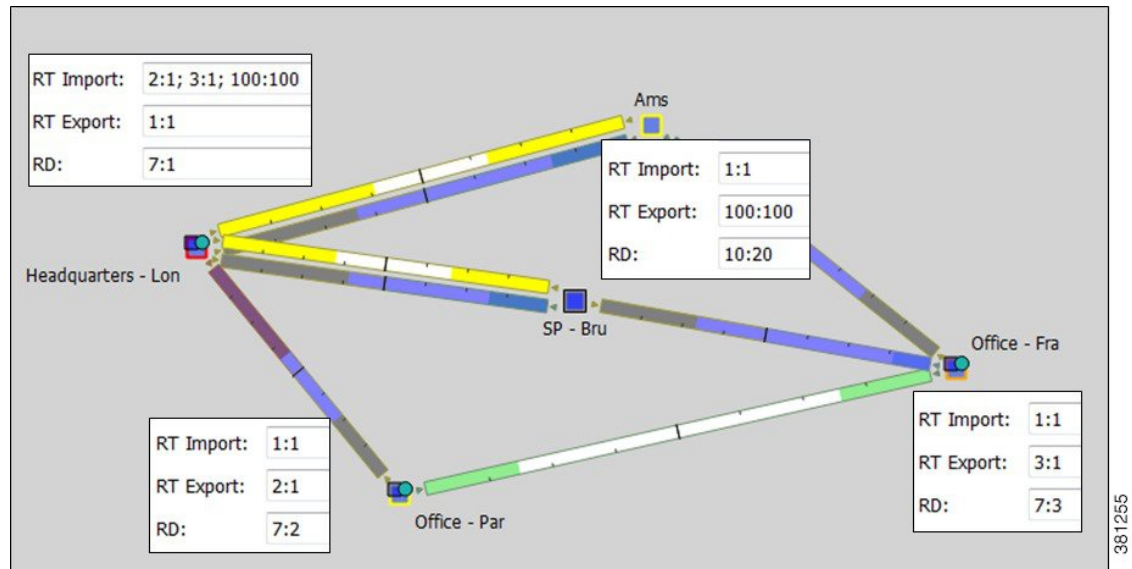
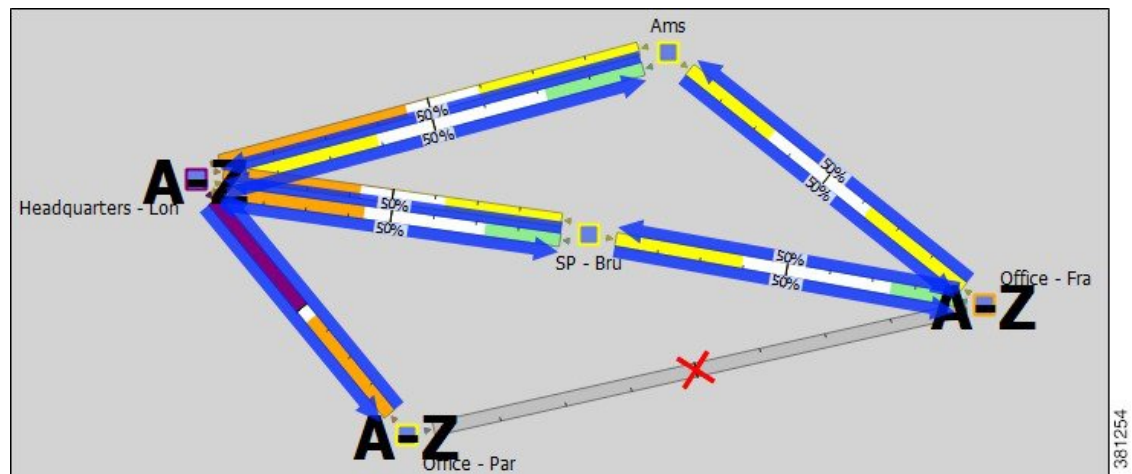


Figure 111: Example Failure Between Branch Offices in the Acme VPN



For this example, [Figure 112: VPN Nodes Filtered to Acme VPN, and Acme VPN Filtered to Demands](#), on page 287 illustrates the filtering of VPN nodes to its associated Acme VPN and the filtering of the Acme VPN to its associated demand traffic. It also shows the calculations of the Total Connect and VPN Connect columns in the VPN Nodes table.

- The Total Connect for the VPN node residing on er1.lon, headquarters is highest because it exchanges data with three other VPN nodes.
Each of the offices and the service provider VPN node have only 1 in the Total Connect column because they each exchange data only with (have RT pairings with) headquarters.
- The VPN Connect for the VPN node residing on er1.lon, headquarters is highest because it exchanges data with and is in the same VPN as the two offices; all three VPN nodes share the same VPN name.

Each office has 1 in the VPN Connect column because it communicates with only one VPN node in the same VPN.

The service provider VPN node (er1.bru) has 0 VPN Connects because it does not reside in a defined VPN.

Figure 112: VPN Nodes Filtered to Acme VPN, and Acme VPN Filtered to Demands

These VPN Nodes ...

Node	Type	Name	VPN	Description	RT Import	RT Export	RD	Total Connect	VPN Connect	Num VPN Nodes
1 er1.fra	Layer3	Acme_VRF	Acme	Acme Inc. Frankfurt	1:1	3:1	7:3	1	1	3
2 er1.lon	Layer3	Acme_VRF	Acme	Acme Inc. London, HQ	2:1; 3:1; 100:100	1:1	7:1	3	2	3
3 er1.par	Layer3	Acme_VRF	Acme	Acme Inc. Paris	1:1	2:1	7:2	1	1	3

Filter to this VPN. This VPN filters to ...

Name	Type	Service Class	Num Nodes	Util Meas	Util Sim	WC Util	WC Failures	WC Traffic Level	Latency
1 Acme	Layer3	VPN	3	42.88	38.03	na	na	na	0.00

These Demands

Source	Destination	Service Class	Traffic	ECMP Min %	Maximum Latency	Diff Min Possible Latency	Path Metric	Routed
1 er1.lon	er1.fra	VPN	25.97	50.00	0.00	0.00	220 T	
2 er1.lon	er1.par	VPN	77.98	100.00	0.00	0.00	210 T	
3 er1.par	er1.lon	VPN	344.39	100.00	0.00	0.00	210 T	
4 er1.fra	er1.lon	VPN	133.48	50.00	0.00	0.00	220 T	

381260

VPN Simulation Analysis

When you run a Simulation Analysis, you have the option to record worst-case utilization and latency for VPNs in the VPNs table (Table 25: Simulation Analysis Columns in the VPNs Table, on page 287). You can then right-click a VPN to fail it to its worst-case utilization or to fail it to its worst-case latency.

Tools

- Simulator
- Simulation Analysis...

Calculate demand worst-case latency

Calculate VPN worst-case utilizations and latency

381269

Table 25: Simulation Analysis Columns in the VPNs Table

Columns	Description
All calculations are based on traffic carried on all interfaces used by the VPN for the service class defined for that VPN.	
WC Util	Worst-case VPN utilization over all failure scenarios.
WC Failures	Failures causing the worst-case utilization of the VPN.

Columns	Description
WC Traffic Level	Traffic level causing the utilization of the interface identified in the WC Util column.
WC QoS Violation	Highest worst-case QoS violation for all interfaces used by this VPN. A QoS violation is equal to the worst-case traffic minus the worst-case capacity permitted (worst-case QoS bound).
WC QoS Violation (%)	Highest worst-case QoS violation for all interfaces in this VPN expressed as a percentage of total capacity.
WC Latency	Maximum VPN latency over failure scenarios considered.
WC Latency Failures	Failures causing the worst-case VPN latency.

Creating VPN Nodes

-
- Step 1** Right-click in an empty plot area and choose **New > VPNs > VPN Node**, or choose **Insert > VPNs > VPN Node**.
 - Step 2** In the Site and Name fields, choose the site in which the VPN node will exist, and choose the node on which the VPN node is being configured.
 - Step 3** Choose a VPN type or enter a string value for a new one. The defaults are VPWS, VPLS, and L3VPN.
 - Step 4** In the Name field, enter the name of the VPN node, which does not have to be unique.
 - Step 5** From the VPN drop-down list, choose the VPN to which you are adding this VPN node. If you do not see the VPN that you expect to see, verify that you correctly chose the type.
 - Step 6** (Optional) Enter a description that identifies the VPN node. For example, a customer name might be helpful.
 - Step 7** If the Connectivity for the VPN is RT, enter the applicable route targets in the RT Import and RT Export fields. All VPN nodes with the same import RT as another VPN node's export RT can receive traffic from that VPN node. Those VPN nodes with the same export RT as another VPN node's import RT can send traffic to that VPN node.
 - Step 8** (Optional) Enter a route distinguisher in the RD field.
 - Step 9** Click **OK**.
-

Creating VPNs

You can create VPNs from existing VPN nodes or you can create new VPNs and then later add VPN nodes with them.

Creating VPNs from Existing VPN Nodes

When you create VPNs from existing VPN nodes, all VPN nodes are assigned to these newly created VPNs and the existing VPNs become empty. This is because VPN nodes can belong to only one VPN at a time.

Inserts VPNs for (3/3) selected VPN Nodes

RegExp substitution of VPN Node Name

VPN Node Name: (.+)

VPN Name: \$1

Service Class: Internet

Preview

	Name	RD	Tags	New VPN Name
1	East_Region-A		ABC_East	East_Region-A
2	East_Region-B			East_Region-B
3	West_Region-B		ABC_West	West_Region-B

381270

- Step 1** If you are creating a VPN for specific nodes, choose VPN nodes from the VPN Nodes table.
- Step 2** Right-click in an empty plot area and choose **New > VPNs > VPNs from VPN Nodes**, or choose **Insert > VPNs > VPNs from VPN Nodes**.
- Step 3** From the drop-down list, choose the method for creating the VPN: VPN node name, RD, or VPN node tag.
- Step 4** If applicable, enter the VPN node name or VPN node RD, and enter the VPN name. These two fields work together to create and name the VPN. Both fields use regular expressions. The \$ in the VPN Name field identifies which parenthetical expression in the VPN Node Name or VPN Node RD field to use. For example, \$2 means use the second set of parenthesis from which to create the VPN name.
- The default is a regular expression that matches the entire VPN node name and to create one VPN for each unique VPN node name. That is, the default in VPN Node Name is (.+) and the default VPN Name is \$1, which creates a VPN with a name that is identical to each VPN node (or all VPN nodes if none are selected).

If your convention is to use the same VRF name or the same service ID for every VPN node, this default works well. If, however, the VPN name is encoded in the VRF name or service ID, use a regular expression to isolate the part of the VPN node name that is to be used.

Example: By adding characters before or after the parenthesis, you can create a set of VPNs that are similar to VPN node names.

Selected VPN node names: AG-VPN-AMS and AG-VPN-FRA

VPN Node Name: AG-(.)

VPN Name: \$1

Results in two VPNs: VPN-AMS and VPN-FRA

Example:

Selected VPN node names: vrf_AKD_V001_Amsterdam, vrf_AKD_V001_Paris, and vrf_AKD_V001_Frankfurt

VPN Node Name: (vrf)_(.)_(V[0-9]+)_(.)

VPN: \$2

Results in one VPN: AKD

- If you created a VPN from VPN node RDs (Step 3), WAE Design uses regular expressions for both the VPN Node RD and VPN Name fields.

Example: Create a VPN named “7” from three existing VPN nodes with RDs of 7:1, 7:2, and 7:3.

VPN Node RD: (.+):(.+)

VPN:\$1

- If you created a VPN from VPN node tags, WAE Design uses a tag to create the new VPN. If a VPN node has more than one tag, only the first tag listed is used. (To create VPN node tags or to change the order of their appearance, use the VPN Node Properties dialog box. Open it by double-clicking one or more VPN nodes.)

Step 5 To see a list of VPN nodes that will be included in the VPN and the VPN names being created, click **Update Preview**.

Step 6 Choose the service class for the VPN and click **OK**.

Creating New VPNs

Step 1 Right-click in an empty plot area and choose **New > VPNs > VPN**, or choose **Insert > VPNs > VPN**.

Step 2 Enter a unique name for the VPN.

Step 3 Choose the VPN type: VPWS, VPLS, or L3VPN.

Step 4 Choose the service class for the VPN.

Step 5 Click **OK**.

Step 6 (Optional) Add VPN nodes to the newly created VPN.

Adding VPN Nodes to VPNs

Step 1 Right-click one or more VPN nodes in the VPN Nodes table and choose **Properties**.

Step 2 In the drop-down list, choose the VPN to which you are adding the VPN nodes.

Step 3 Click **OK**.



CHAPTER 17

Multicast Simulation

WAE Design supports source-specific multicast (SSM), which is a method of delivering multicast packets to receivers only from a specified source address that is requested by the receiver. By limiting the source, SSM reduces resource requirements and improves security.

This section contains the following topics:

- [SSM Parameters](#) , on page 291
- [Discovered Versus Simulated Multicast Flows](#), on page 291
- [Multicast Demands](#), on page 294
- [Creating Multicast Flows](#), on page 297
- [Creating Demands for Multicast Flows](#), on page 299
- [Toggling Multicast Flow Destinations to On and Off](#), on page 299

SSM Parameters

SSM is specified by an (S,G) parameter per multicast flow. The (S,G) pair is labeled using a dot-decimal notation similar to IP addresses (for example, 1.1.1.1, 2.2.2.2). Each (S,G) pair is the name of a multicast flow, and each flow is listed in the Multicast Flows table.

Discovered Versus Simulated Multicast Flows

How you work with multicast flows within WAE Design differs depending on whether the multicast flow was discovered by WAE or whether you are simulating it ([Table 26: Discovered Versus Simulated Multicast Flows](#) , on page 291).

Table 26: Discovered Versus Simulated Multicast Flows

	Discovered Multicast Flow	Simulated Multicast Flow
Creation	WAE discovers multicast flows—(S,G) pairs—and multicast traffic is derived from the Multicast Flow Traffic table (accessed from the Multicast Flows tab after choosing Edit > Traffic).	First, manually create the multicast flow, defining both the source (S) and the destinations (G). Next, create a demand that links the source to these multicast destinations, allowing the routing to be simulated through the network.

	Discovered Multicast Flow	Simulated Multicast Flow
Hops	Each discovered multicast flow includes multicast flow hops, which are node-interface combinations through which the multicast path flows. The Multicast Demand initializer uses multicast flow hops to determine multicast sources and destinations.	When you create a demand, it determines the path to take.
External hops	WAE discovers multicast flow hops on interfaces that are external to the plan file. These are interfaces from a plan node to an external node, or from an external node to a plan node. The Multicast Demand initializer uses multicast flow external hops to determine multicast sources and destinations.	When you create a demand, it determines the path to take, but multicast external flow hops are not identified.
Destinations	WAE does not identify a list of multicast destinations for each flow. The Multicast Demand initializer generates destinations from the Multicast Flow Hops and Multicast Flow External Hops tables.	Specify destinations (nodes, interfaces, external ASes, or external endpoints) when you create multicast flows. When creating multicast demands, specify these as multicast destinations.
Applicable tables	<ul style="list-style-type: none"> • Multicast Flows • Multicast Flow Hops • Multicast Flow External Hops • Multicast Flow Traffic (accessed from the Multicast Flows tab after choosing Edit > Traffic) 	<ul style="list-style-type: none"> • Multicast Flows • Multicast Flow Destinations • Demands
Plot displays	<ul style="list-style-type: none"> • Selecting a multicast flow from the Multicast Flow table highlights a pink path on the plot (Figure 113: Multicast Flows, on page 293). • Selecting a multicast flow hop from the Multicast Flow Hops table highlights a pink path on the plot (Figure 114: Multicast Flow Hops, on page 293). 	<ul style="list-style-type: none"> • Selecting multicast demands from the Demands table highlights the multicast path in blue on the plot and labels the source as A and the destinations as Z. • Selecting multicast flow destinations from the Multicast Flow Destinations table labels the final destinations with a Z in the plot.

Figure 113: Multicast Flows

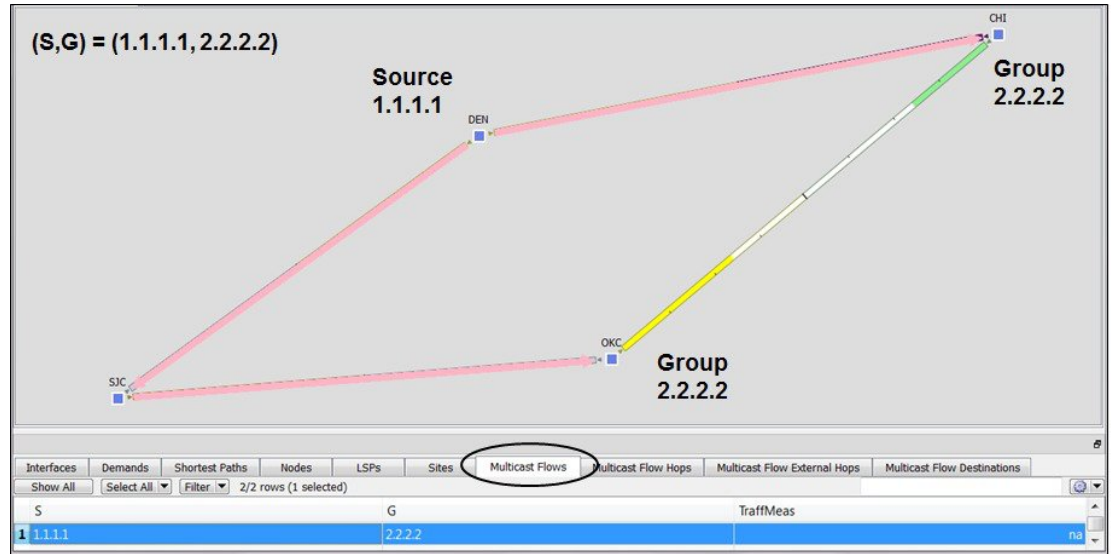
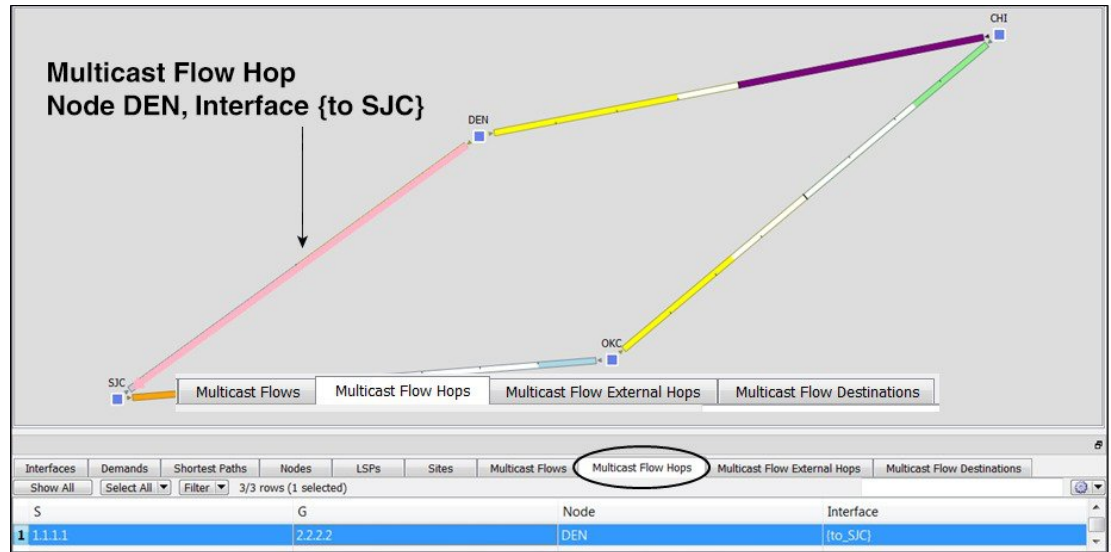


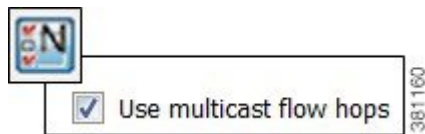
Figure 114: Multicast Flow Hops



Global Multicast Simulations

Flow Hops

If a plan file contains multicast information, it includes the current hops taken by multicast flows. You can specify that WAE Design multicast simulations should follow these flow hops if possible. This is useful, for example, when calculating incremental routing changes on the current network state, such as those caused by a failure. For planning purposes, when the current state is not relevant, you can change this behavior to disregard the multicast flow hops.

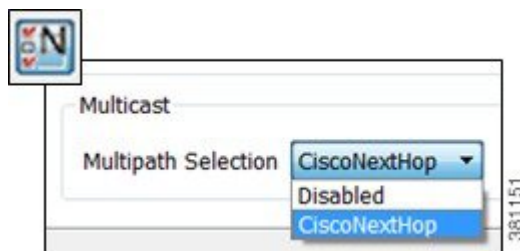


WAE Design uses the network state in its multicast simulation. You can set simulations to take into account multicast flow hops, as follows.

-
- Step 1** Choose **Edit > Network Options** or click the **Network Options** icon in the Visualization toolbar.
 - Step 2** Click the **Simulation** tab.
 - Step 3** To use or disregard multicast flow hops in simulation, check or uncheck **Use multicast flow hops**, and click **OK**.
-

Cisco Next Hop

For manually inserted demands, all SSM demand traffic for the (S,G) pair (multicast flow) goes through any interface that is traversed by that demand.



However, you can set WAE Design to use Cisco next hops, which are calculated using a hash on S and G for a multicast flow (S,G). The hash calculation is different in IOS and IOS XR. The default behavior is that of IOS. The IOS XR hash is used on all nodes whose OS field starts with IOS XR.

-
- Step 1** Choose **Edit > Network Options** or click the **Network Options** icon in the Visualization toolbar.
 - Step 2** Click the **Protocols** tab.
 - Step 3** From the Multipath Selection drop-down list, choose **CiscoNextHop**, and click **OK**.
-

Multicast Demands

For discovered multicast flows, you can insert demands manually or use the initializer.

For simulated multicast flows, create demands manually. For information on demands in general, see [Traffic Demand Modeling, on page 43](#).

To find demands associated with a multicast flow, right-click either a multicast flow or multicast flow destination, and choose **Filter to Demands**. Conversely, to find multicast flows associated with demands, right-click a demand and choose **Filter to > Multicast Flows**.

Initialized Multicast Demands

The initializer creates simulated multicast flows corresponding to the discovered multicast flows. The demand traffic for all traffic levels existing is set to the traffic measured in the multicast flow. Following is how WAE Design determines the source and destination.

- **Source**—WAE Design first looks for an external endpoint with the same source (S) name as the source of the discovered multicast flow. If this source does not exist, WAE Design infers it from the multicast flow hops that have traffic going out of them, but none coming in. The inferred source can be a node or external endpoint.
- **Destination**—WAE Design first looks for multicast flow destinations with the same (S,G) name as those in the discovered multicast flow. If these multicast node destinations do not exist, WAE Design infers them from the multicast flow hops that have traffic going into them, but none coming out. WAE Design also infers multicast external flow hops if traffic is leaving a node in a direction that is outside the plan. These destinations are populated in the Multicast Flow Destinations table.

To run this initializer, choose **Initializers > Multicast Demands**.

Simulated Multicast Demands

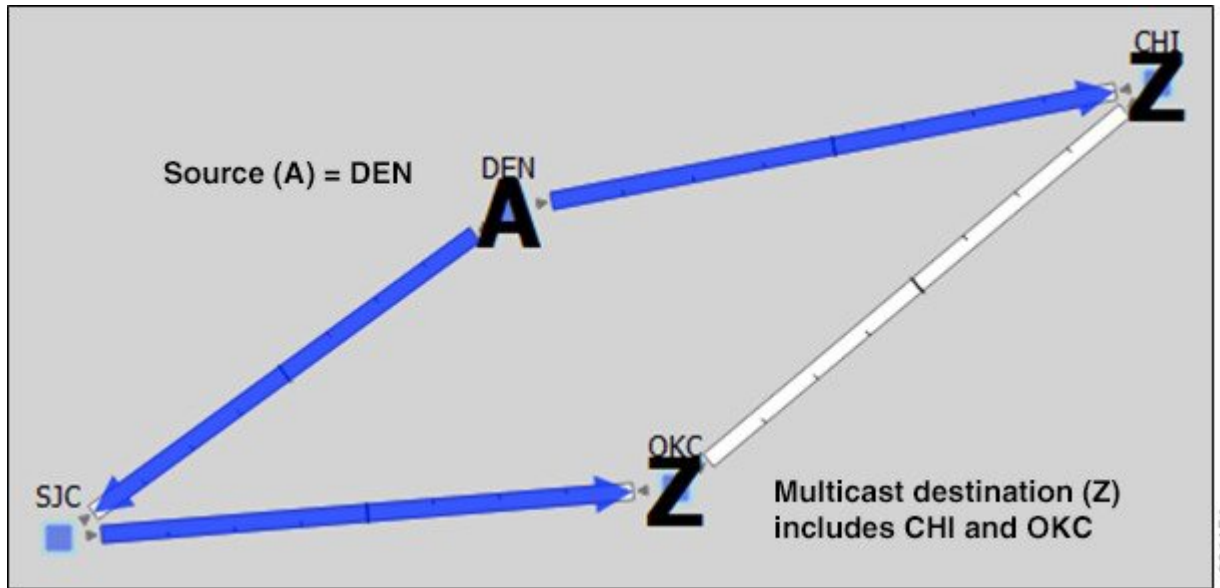
For manually inserted demands, all SSM demand traffic for the (S,G) pair (multicast flow) goes through any interface that is traversed by that demand. SSM demands are routed to take the shortest path from the destination to the source, rather than from the source to the destination, as unicast demands do. By default, multicast multipath is disabled. If two paths of equal cost exit from a node on the route back to the source, the path is chosen based on:

- The remote interface with the highest IP address is used.
- If IP addresses are not available, the router name with the lowest lexicographical name is used.

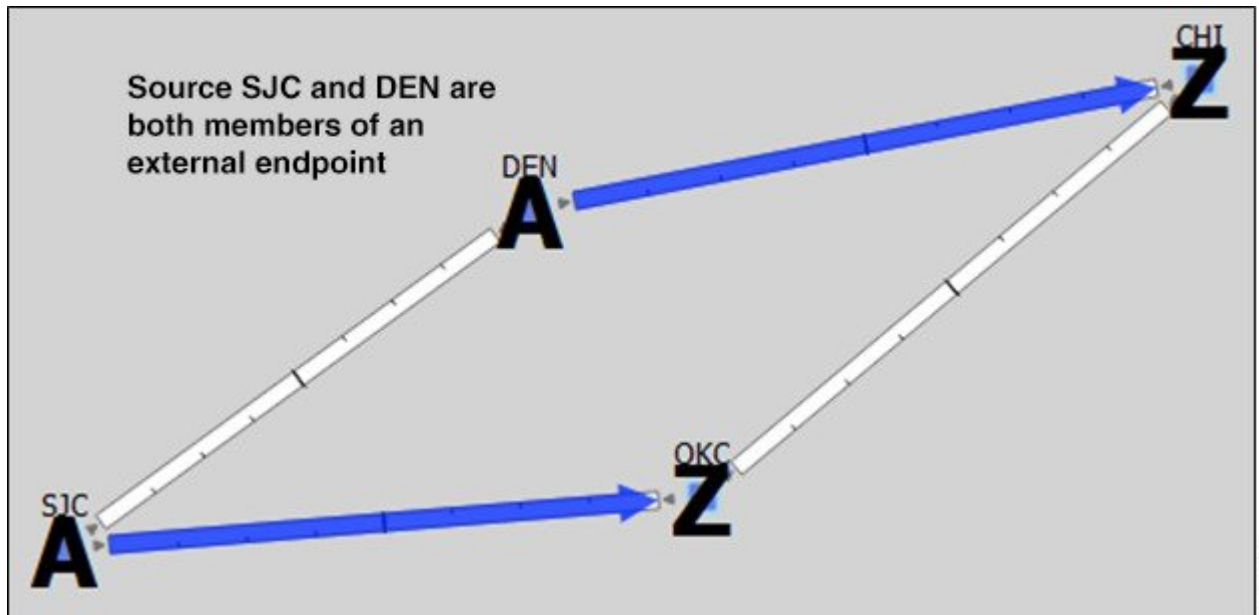
However, you can set WAE Design to use the Cisco next-hop multicast, multipath selection method. For information, see [Cisco Next Hop, on page 294](#).

The sources of these multicast demands can be nodes, interfaces, external ASes, or external endpoints. Using interfaces lets you specify the exact interface on which demand traffic is entering. Using external endpoints lets you model situations where the multicast source is outside of the nodes in the plan, and there is more than one possible entry point of the traffic flow into and through the interfaces in the plan.

In this example, the selected demand has a node source of DEN, and a multicast destination containing two nodes: CHI and OKC. The source node (S) is marked by A, and the destinations in the receiver group (G) are marked by Z.



You can model a source that is outside the plan using external endpoints. In this example, the demand's source is an external endpoint that contains SJC and DEN as members. The multicast destination includes both CHI and OKC.

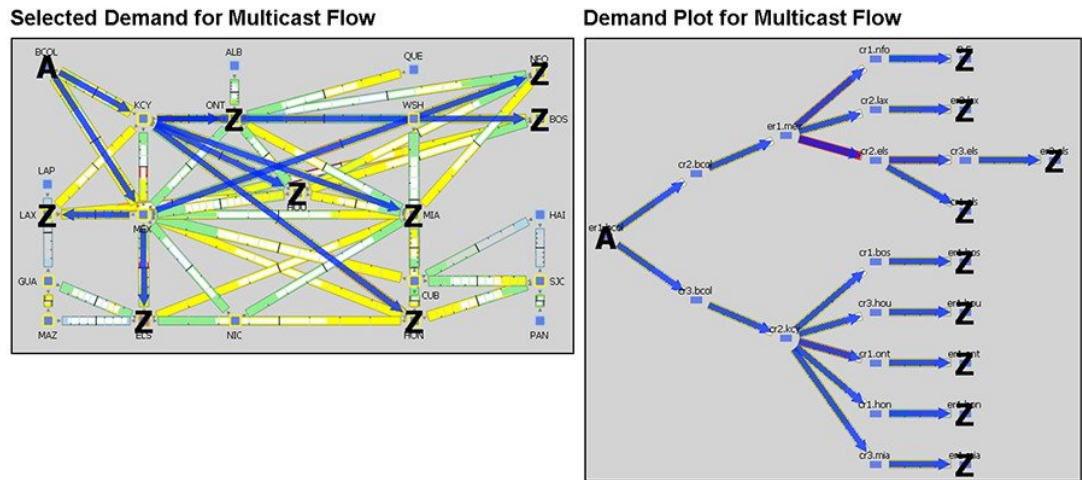


Viewing Multicast Flows

To highlight discovered multicast flows and multicast flow hops in the plot, select them from their respective tables. To view sources and destinations of multicast demands, select the demand from its table. The source is identified in the plot by A and the destinations are identified by Z.

For complex multicast flows, filter to the demand, right-click the demand, and choose **Plot Demand**. [Figure 115: Example Demand Plot of Multicast Flow, on page 297](#) shows an example of a demand plot for a multicast flow.

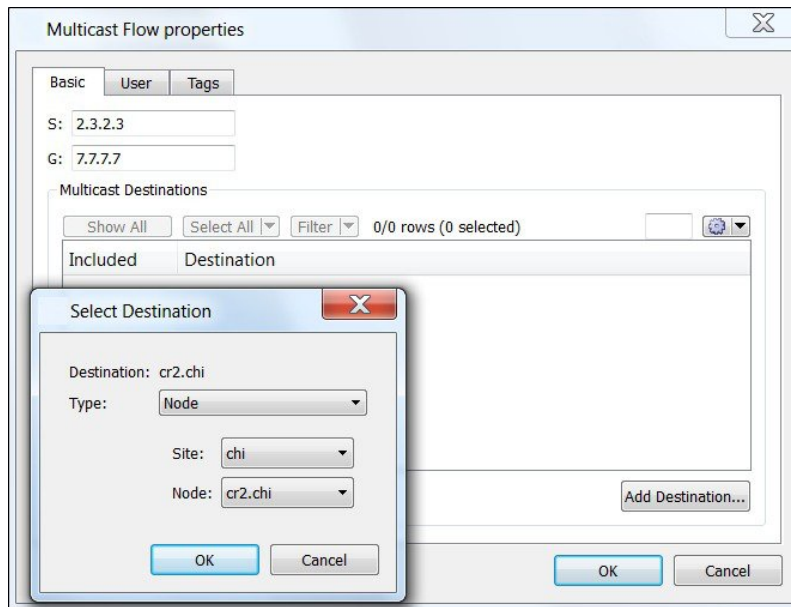
Figure 115: Example Demand Plot of Multicast Flow



To View...	Show This Table
Multicast flows, including both source and receiver names	Multicast Flows
Discovered multicast flow hops, including (S,G) name, hop node, and hop interface	Multicast Flow Hops
Discovered multicast flow external hops (destinations that are inferred as external to the plan), including (S,G) name, direction of the outbound interface	Multicast Flow External Hops
User-created multicast flow destinations, including (S,G) name and destination node	Multicast Flow Destinations
Multicast demands	Demands

Creating Multicast Flows

The recommended method of creating multicast flow destinations is to do so while creating the associated multicast flows, as follows.



-
- Step 1** Choose **Insert > Multicast Flow**, or right-click the plot and choose **New > Multicast Flow**. A Multicast Flow dialog box opens.
- Step 2** In the S field, identify the source node by entering its dot-decimal name; for example, 2.3.2.3.
- Step 3** In the G field, identify the collective name for all receiver nodes (destinations) by entering its dot-decimal name; for example, 7.7.7.7.
- Step 4** Add destinations to the receiver group.
- Click **Add Destination**.
 - In the Type list, select either a node, interface, external AS, or external endpoint.
 - If you are adding a node to the receiver group, select both the site and node name of the destination point in the multicast flow.
 - If you are adding an interface, select the site, the node, and the interface.
 - If you are adding an external AS, select the external AS name and node for the destination point.
 - If you are selecting an external endpoint, select its name.
 - Click **OK**. These values now appear in the Multicast Flow dialog box, where you can disable or enable them from participating in the multicast flow.
- Step 5** Repeat Step 4 for each destination in the multicast flow.
- Step 6** Click **OK** to save and exit.
- After creating the multicast flow, you need to create demands for the multicast flows.
-

Creating Demands for Multicast Flows

- Step 1** Choose **Insert > Demands > Demand**, or right-click the plot and choose **New > Demands > Demand**.
- Step 2** In the Name field, enter the name of the demand.
- Step 3** In the Source area, define the source (S) of the multicast flow.
- From the Type list, select the source as a node, interface, external AS, or external endpoint.
 - For node sources, select both the site and the node.
For interface sources, select the site, node, and the interface.
For external ASes, select both the external AS and the node within the plan file through which this external traffic flows.
For external endpoints, select its name.
- Step 4** In the Destination area, define the receiver group (G) of the multicast flow.
- From the Type list, choose **Multicast Destination**.
 - From the Receiver list, choose the receiver that identifies the simulated multicast flow (S,G).
- Step 5** (Optional) Complete all other fields as needed, and then click **OK**.

toggling Multicast Flow Destinations to On and Off

- Step 1** In the Multicast Flows table, double-click the multicast flow. A dialog box opens with a list of existing destinations. Each destination has a T next to it if it is currently part of the receiver group or an F if it exists, but is not part of the receiver group.

	Included	Destination
1	<input type="checkbox"/> F	if{cr1.ams to_cr2.ams}
2	<input checked="" type="checkbox"/> T	cr1.chi

381161

- Step 2** To enable or disable the multicast flow destination, click its associated T/F value.
- Step 3** Click **OK** to save and exit.



CHAPTER 18

Metric Optimization

The [Metric Optimization, on page 302](#) tool optimizes metrics in the network core. It replaces the IGP metrics of core interfaces in a network plan with new metrics chosen to maximize the throughput achievable by the resultant routes, under a specified set of failure scenarios.

The [Tactical Metric Optimization, on page 304](#) tool lets you quickly fix local congestion by doing as few modifications as possible on selected interfaces to push utilizations under a given level.



Note The core of the network must be connected, and all nodes must be contained in one AS. That is, there must be a path between any two core nodes that passes only through other core nodes in the same AS, and not through edge nodes. If this is not the case, Metric Optimization signals an error.

This section contains the following topics:

- [Core Versus Edge, on page 301](#)
- [Metric Optimization, on page 302](#)
- [Tactical Metric Optimization, on page 304](#)
- [Reports, on page 305](#)

Core Versus Edge

WAE Design plan files allow nodes to be defined as either core nodes or edge nodes. Note that these definitions are not related to any explicit router configurations. A similar classification of circuits is determined implicitly by the nodes they connect. If a circuit is attached to one or more edge nodes, it is defined as an edge circuit. Otherwise, it is a core circuit. Edge metrics and core metrics mean metrics of edge or core circuits, respectively.

This core and edge distinction helps define certain policies followed by WAE Design tools. In Metric Optimization, these are as follows:

- Only core metrics are changed.
- The objective is to minimize utilization in the core. Utilization in the edge is not taken into account.
- The routes selected by the metrics are restricted by the tool so that any route between a source and a destination will enter the core at most once, and leave the core at most once. That is, *edge leakage* of routes is prevented.

A consequence of these policies is that larger edge metrics in the plan file to be optimized might result in better optimized core routes. With larger given edge metrics, the Metric Optimization tool has more flexibility in how it can set the core metrics without leakage of routes into the edge.

Metric Optimization

There are many trade-offs in metric optimization. The main trade-off is between performance under normal operation versus performance under failure. Metrics can be selected to do well under the former while ignoring the latter, or to try to balance performance between the two.

Metric optimization optimizes undifferentiated traffic, but does not optimize traffic per service class or per interface queue. If there are policies set on the undifferentiated traffic, WAE Design tries to minimize both the interface utilization and the policy violations.

-
- Step 1** Choose **Tools > Metric Optimization**.
- Step 2** Decide on the normal (non-resilient) optimization options to use. (See [Table 27: Metric Optimization Options](#), on page 302 for field descriptions.)
- Step 3** Decide on the failure (resilient) options to use. (See [Table 27: Metric Optimization Options](#), on page 302 for field descriptions.)
- For planning purposes, the most benefit from metric optimization is that it is able to optimize routing over a large number of failure scenarios. This capability controls that optimization, as well as controls the trade-off between normal and failure optimizations.
- Step 4** Specify the remaining options to fine-tune the optimization, and then click **OK**.
-

What to do next

Table 27: Metric Optimization Options

Field	Description
Normal (Non-Resilient) Options	
Minimize maximum interface utilization	This default is usually the primary objective, and concentrates on reducing the utilization of the interface with the highest utilization.
Minimize # of interfaces with utilization > x %	Often a small number of interfaces are bottlenecks in the network, and their utilizations cannot be reduced through rerouting. However, it is still desirable to reduce the utilization of other, under-utilized interfaces. This option tries to reduce all interface utilizations to below a specified utilization level, or, if this is not possible, to minimize the number of interfaces over this level.
Minimize average latency	If after the optimizations flexibility is available in the choice of metrics, WAE Design uses this flexibility to minimize the average latency of the routes across the network. This is not user-configurable and is always on.
Enforce latency bounds	If checked, the choice of paths for each demand is restricted, if possible, to those demands with latency below the latency bound.

Field	Description
Failure (Resilient) Options	
Minimize maximum interface utilization	If checked, WAE Design selects metrics to minimize the maximum interface utilization over all interfaces and all failure scenarios. If unchecked, only normal operation is optimized, and the remainder of the failure section is ignored.
Minimize # of interfaces with utilization > x %	The number of interfaces with utilization (under any failure scenario) above a specified percentage is minimized.
Enforce no-failure utilization bound	If unchecked, maximum failure utilization is minimized with no consideration for utilization under normal operation. It is often convenient to check this option and place a bound on the utilization under normal operation, which Metric Optimization then satisfies, if possible.
Failures to consider	<p>This option determines the failure scenarios over which optimization is performed. You can select any combination of failures.</p> <ul style="list-style-type: none"> • All unprotected circuit, node, Layer 1 (L1) node, and L1 link failures • All SRLG failures <p>This option does not consider port or port circuit failures. For information on protecting objects from being included in this analysis, see Simulation Analysis.</p> <p>To control the number of threads that WAE Design processes in parallel when examining failure scenarios, set the Maximum number of threads field.</p>
Other Options	
Optimization Type	<p>WAE Design can perform global or incremental metric optimizations.</p> <ul style="list-style-type: none"> • Incremental—Uses the current routes as a basis from which improvements can be made. The target routes are not changed as radically as when using global optimizations, and therefore the number of metric changes from the metric targets is typically much smaller using the incremental option. • Global—Creates optimal routes from scratch, and targets the old metrics afterward. Because WAE Design can choose routes globally, this option tends to perform better under some circumstances, particularly when the target routes are poorly chosen. However, global optimization typically takes longer than incremental optimization.
Set Metrics on Interfaces	By default, WAE Design sets or changes metrics on any of the core interfaces in the network to achieve its optimization goals. Alternatively, as provided by this option, Metric Optimization restricts metric changes to the following interfaces: all, core, selected in table, or tag name.
Traffic Level	The traffic level over which the optimization is performed.

Field	Description
Interfaces to Optimize	<p>By default, WAE Design optimizes all interfaces. You can restrict the optimized interfaces using this option and choosing the Optimize utilization on interfaces value: all, core, selected in table, or tag name.</p> <p>For those interfaces that are not optimized, you can set the following:</p> <ul style="list-style-type: none"> • Ignore • Minimize number of interfaces with util > ___ % • Minimize number of interfaces with util > current + ___ %
Target to Current Metrics	<p>There are many sets of metrics that will result in equivalent sets of routes. (Doubling all metrics, for example, does not change routes.) WAE Design selects between these sets of metrics by metric targeting. Metrics can be selected to match as closely as possible the current metrics in the network. This simplifies the changeover from one set of metrics to the next.</p> <p>Targeting optimized metrics to the current metrics in the plan is performed by default. Various targeting refinements are available as CLI options to specify upper and lower bounds on metric selections.</p>
Enforce Symmetric Metrics	<p>This option sets the two metrics on each interface of each circuit equal to one another. This constraint can reduce optimization performance. However, it ensures that demands from A (source) to Z (destination) and from Z to A (in the core) take the same path, and so the chances of a session between A and Z being interrupted by a failure is minimized.</p>
Prevent Edge Leakage	<p>This default option restricts how WAE Design can change demand routes. If a demand starts by routing only through the core, or from the edge through the core to the edge, this option prevents the routing from being changed so that it routes through the edge in the middle of its path. This is usually a desired routing policy restriction.</p>
Simulate after optimization	<p>This default option performs a simulation analysis after optimizing metrics. The simulation analysis allows a detailed view, for example, of the utilizations under failure resulting from the new metric settings.</p>
New Plan for Result	<p>This default option creates a new plan as a result of running Metric Optimization. Unless a name is specified, WAE Design attaches the suffix <i>-mopt</i> to the plan filename. If not selected, WAE Design changes the current plan file with the updated information.</p>

Tactical Metric Optimization

The Tactical Metric Optimization tool (**Tools > Tactical Metric Optimization**) is a reduced version of the Metric Optimization tool that minimizes the number of changes and runs faster. The options for this tool are a subset of those for the Metric Optimization tools.

The tool tries to find the most optimal solution it can. In general, the longer it runs, the better the result. The Time Limit option tells the tool to stop looking for better solutions after the specified amount of time, and use the best solution it has found so far. If the option is not set, the tool continues to run until it runs out of solutions to explore, which can take a long time. If you need all available solutions to be explored, use the [Metric Optimization, on page 302](#) tool.

Reports

Metric Optimization writes a Metric-Opt report to the Design History area of the Reports window. To access this information later, choose **Window > Reports**.

Metric Optimization Report

- Options—Summarizes the options used to call Metric Optimization. Some of these options (for example, the Output file and Report file) are only relevant when using the CLI tool.
- Number of demands—Total number of demands in the plan.
- Objectives—Summarizes the optimization objectives in order of priority.

Warnings

Warnings detected by Metric Optimization might lead to misleading or undesirable results.

- No Improvement in Optimization

```
No routing improvement found: metrics unchanged
```

Metric Optimization was unable to improve on the metrics in the network. If the incremental option was specified and latency bounds were enforced, this could be because the target metrics violated latency bounds in ways that incremental optimization could not fix.

- Optimization Exit Diagnostics

Optimization performance may be limited by low edge metrics

During the optimization, some desirable routes could not be selected because the given edge metrics would have caused edge leakage.

- Optimization Constraints Violated

- `<n>` \{intra, inter\site core metrics exceed \crlf\{intra, inter\site-metric \{upper, lower\}-bound

One or more intrasite or intersite core metric bounds have been violated.

- Maximum normal utilization exceeds specified bound

The original plan had normal utilization above the specified bound, and Metric Optimization was unable to reduce the normal utilization below this bound. If this occurs, Metric Optimization tries to set normal utilization as low as possible. No worst-case optimization is performed because reducing utilization under the normal scenario is the highest priority.

- `<n>` demands with non-zero bandwidth exceed latency bounds

Because enforcing latency bounds is the highest optimization priority, this situation should not occur unless a latency bound is less than the shortest possible latency for a demand.

- `<n>` demands with zero bandwidth exceed latency bounds

Metric Optimization does not always enforce latency bounds for zero bandwidth demands. These warnings can occasionally occur even if it is possible to find lower latency paths for the signaled demands.

- Unrouted Demands

- `<n>` unroutable demands under normal operation

There are `<n>` demands for which no route exists between source and destination, even under the No Failure scenario.

- `<n>` unroutable demands under `<m>` circuit failure scenarios

There are `<n>` demands for which no route exists between source and destination under `<m>` circuit failure scenarios. (Different demands might be unroutable under different circuit failures.)

- `<n>` unroutable demands under `<m>` SRLG failure scenarios

There are `<n>` demands for which no route exists between source and destination under `<m>` SRLG failure scenarios. (Different demands might be unroutable under different SRLG failures.)

- `<n>` unroutable demands under `<m>` source/dest node failure scenarios

There are `<n>` demands whose source and/or destination nodes fail under `<m>` failure scenarios. These demands clearly cannot be routed in these circumstances. (Different demands might be unroutable under different node failures.)

- `<n>` unroutable demands under `<m>` non-source/dest node failure scenarios

There are <n> demands for which no route exists between source and destination under <m> node failure scenarios, where the nodes that fail are intermediate nodes in the path of the demands, and are not source/destination nodes of the demands. (Different demands might be unrouted under different node failures.)

Routing Summary

Summary statistics are displayed of the routes in network, both before (in parentheses) and after the optimization.

- Core/Edge Max Utilization—The maximum percentage utilization over any interface in the core or edge network. The normal utilization is under the normal (no-failure) scenario, and the worst-case utilization is maximized over all failure scenarios in the failure set.
- Latency—Median and average latencies over all routed demands, both in milliseconds and as a percentage of the latency of the smallest latency route possible.

The latency of each demand's routing is calculated as a percentage of the latency of the shortest possible route, by latency, for that demand. Under the percentage column, the median and average of these percentages are displayed.

- Num of routes away from shortest path—The number of routes (out of all the demands routed) that do not follow the shortest path, by latency. This statistic is an indicator of how much the routes have been affected by utilization minimization as the primary criterion for optimization.

Metrics

The metric targeting choice is shown, together with the number of metrics different from the target metrics. Lists of interfaces for which metrics differ from the target metrics are displayed, together with both the target metric and the optimized metric. The metrics are separated into those on intrasite and intersite interfaces.



CHAPTER 19

LSP Optimization

This section contains the following topics:

- [LSP Disjoint Path Optimization, on page 309](#)
- [LSP Loadshare Optimization, on page 316](#)
- [LSP Setup Bandwidth Optimization, on page 320](#)

LSP Disjoint Path Optimization

LSPs and LSP paths are *disjoint* if they do not route over common objects, such as interfaces, nodes, or L1 links. The LSP Disjoint Path Optimization tool (**Tools > LSP Optimization > LSP Disjoint Path Opt**) creates disjoint LSP paths for RSVP LSPs and SR LSPs and optimizes these paths based on user-specified constraints.



Note LSP Disjoint Path Optimization tool supports both Inter-Area and Inter-AS functionalities.

One common use case for this optimization is that disjoint LSPs can ensure services are highly resilient when network failures occur. For example, this tool lets you route LSPs to have disjoint primary and secondary paths that are optimized to use the lowest delay metric possible.

If it is not possible to achieve the optimization as defined by the routing selection, path requirements, and constraints, the tool provides the best disjoint paths and optimization possible.

Upon completion, WAE Design tags the LSPs with *DSJOpt* and generates a new plan file with an *-DSJOpt* suffix. WAE Design also writes a report containing the results of the optimization.



Note While the optimizer applies to both RSVP LSPs and SR LSPs, only one of these types of LSPs can be optimized at a time.

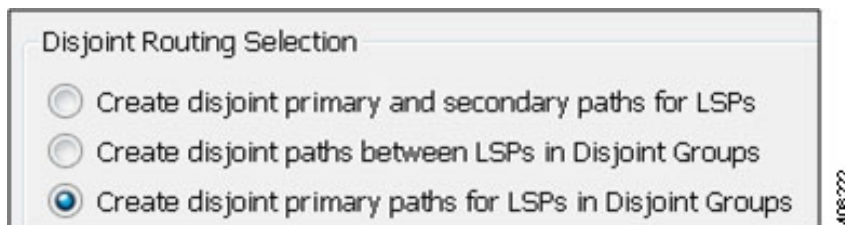
Disjoint Routing Selection

Only existing LSP paths are rerouted. New LSP paths are not created.

- Explicit hops are modified or created for RSVP LSPs.

- Segment list hops are modified or created for SR LSPs. The final hop is either a node hop or interface whose remote node is the destination of the LSP.

Segment lists are created only for LSP paths, and only LSP path segment lists are updated. If a segment list is associated with an LSP (rather than an LSP path), that LSP segment list is removed.



Routing options include:

- Create disjoint primary and secondary paths for LSPs—For all LSPs, whether they are in a disjoint group or not, route all LSP paths so that they are disjoint from all other paths belonging to that LSP. This disjointness extends beyond primary and secondary paths to include all other path options (for example, tertiary).
- Create disjoint paths between LSPs in disjoint groups—For all LSPs that are in disjoint groups, route all LSP paths so that they are disjoint from all other paths belonging to LSPs in that disjoint group. This disjointness extends beyond primary and secondary paths to include all other path options (for example, tertiary).

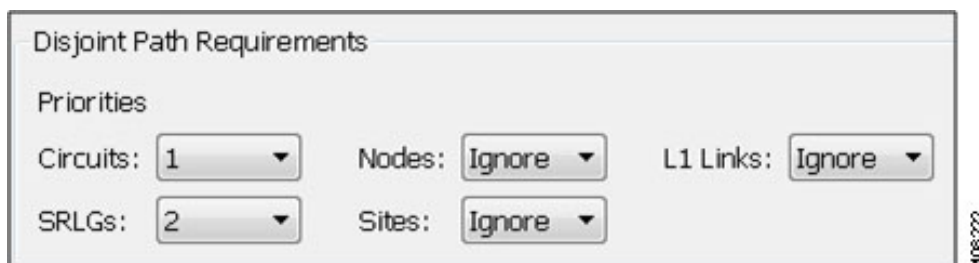
Example: All LSPs in disjoint group East are rerouted to be disjoint from each other. All LSPs in disjoint group Southeast are rerouted to be disjoint from each other. However, LSP paths in the East group are not rerouted to be disjoint from those in the Southeast group.

- Create disjoint primary paths for LSPs in disjoint groups—For all LSPs that are in disjoint groups, reroute only their primary paths so that they are disjoint from each other.

Disjoint Path Requirements

The disjoint path requirements identify the priority for creating disjointness across a path. Disjointness priorities 1, 2, 3, and Ignore are available for circuits, SRLGs, nodes, sites, and L1 links. The tool tries to create disjointness for all objects that have a priority set other than Ignore. If full disjointness cannot be achieved, the tool prioritizes disjointness based on these values.

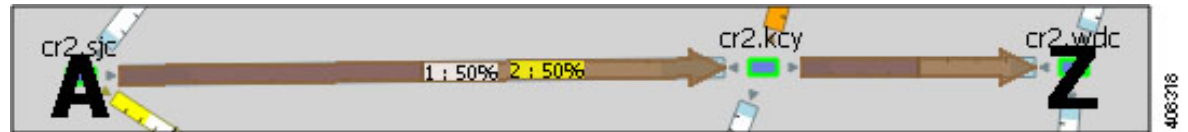
Example: Circuits have a priority of 1, SRLGs have a priority of 2, and the other objects are ignored. If the tool cannot achieve full disjointness across both circuits and SRLGs, it prioritizes the disjointness of circuits over SRLGs.



Example

This example shows how disjoint routes can be created for primary and secondary RSVP LSP paths, and how those routes differ, depending on the path requirements set.

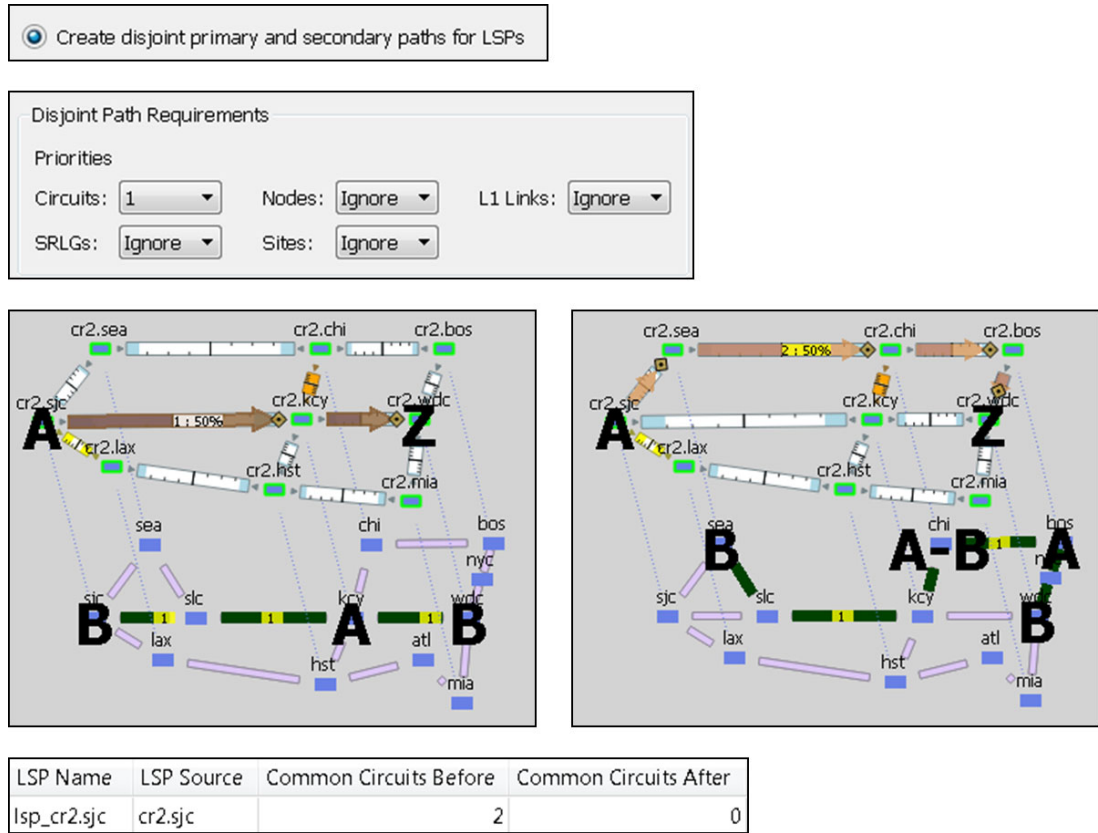
The LSP has a primary and secondary LSP paths that use the same route from cr2.sjc to cr2.wdc. The LSP is not a member of a disjoint group.



In both [Figure 116: Primary and Secondary Paths Based on Disjoint Circuit Requirements, on page 312](#) and [Figure 117: Primary and Secondary Paths Based on Disjoint Circuit and L1 Link Requirements, on page 313](#), the option is selected to create disjoint primary and secondary paths for LSPs. However, they show different secondary path routes.

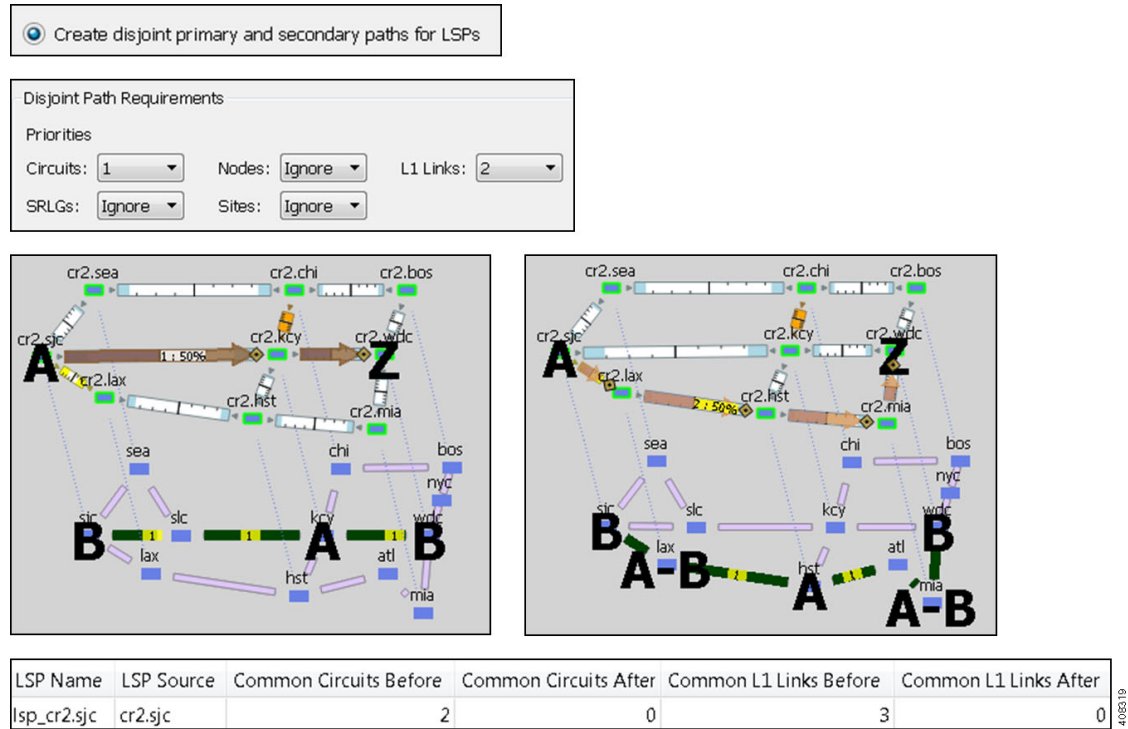
- [Figure 116: Primary and Secondary Paths Based on Disjoint Circuit Requirements, on page 312](#) shows explicit interface hops set for the secondary LSP path that force it to take a different (northern) route from cr2.sjc to cr2.wdc. Because the disjoint path requirement is only circuits, the primary and secondary paths route across different circuits, as indicated in the resulting report. However, they share a common L1 link between slc and kcy.
- [Figure 117: Primary and Secondary Paths Based on Disjoint Circuit and L1 Link Requirements, on page 313](#) is based on setting circuits as the first priority for disjoint requirement and L1 links as the second priority. The result is a different set of explicit hops for the secondary LSP path, routing it through southern disjoint circuits that are mapped to different L1 links. The resulting report shows there are no longer any common L1 links.

Figure 116: Primary and Secondary Paths Based on Disjoint Circuit Requirements



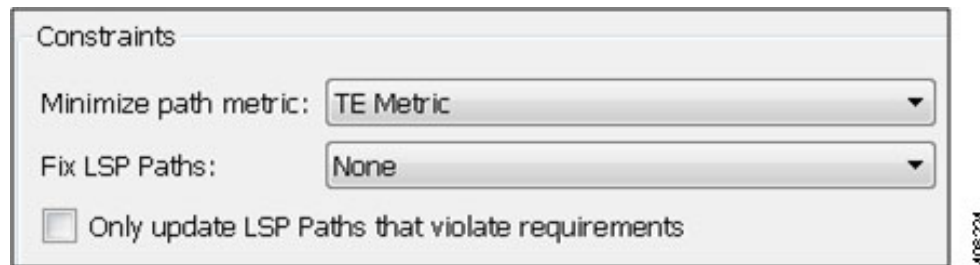
408321

Figure 117: Primary and Secondary Paths Based on Disjoint Circuit and L1 Link Requirements



Constraints

- Minimize path metric—Paths are optimized to minimize the sum of the metrics along the path with respect to delay, TE, or IGP metrics. All of these properties are configurable from an interface Properties dialog box, and delays can also be set in a circuit Properties dialog box.
- Fix LSP Paths—Selected or tagged LSP paths are not rerouted. This constraint is useful when you have previously optimized specific LSPs within the network and want to maintain their routes.
- Only update LSP Paths that violate requirements—Paths are modified only if they violate the requirements specified in the area of the [Disjoint Path Requirements, on page 310](#) dialog box.

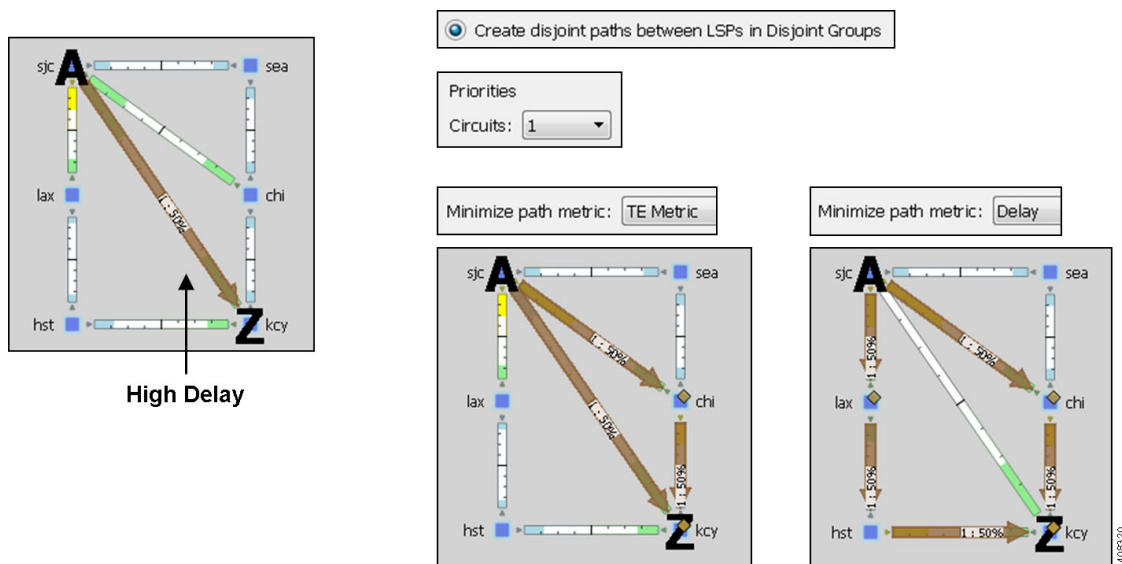


Example

This example shows how disjoint routes can be optimized for two SR LSPs using the same source (sjc) and destination (kcy).

- Both LSPs belong to the same disjoint group, and both LSPs have an LSP path.
- The circuit between sjc and kcy has significantly higher delay than the other circuits.
- The only disjoint path requirement selected is circuits.
- [Figure 118: Example Routing of SR LSPs, on page 314](#) shows the following:
 - Before using the LSP Disjoint Path Optimization tool, both LSP paths use the same route. There are no segment list hops.
 - By selecting the option to create disjoint paths between LSPs in the same disjoint group and using the TE metric for the shortest path calculation, two disjoint LSP path routes are created. Both have a segment list node hop on the destination node. One has an additional segment list node hop on chi to force a different route.
 - By selecting the same disjoint option and using the Delay metric for the shortest path calculation, one LSP is moved away from the high-delay sjc-kcy circuit because traversing that circuit is not the shortest latency path.

Figure 118: Example Routing of SR LSPs



Disjoint Report

The resulting report summarizes the number of LSPs and number of updated LSP paths. Depending on the disjoint option selected, the report summarizes the uniquely distinguishing attributes, such as LSP name and disjoint group name.

The Disjoint Groups, LSP Disjointness, and Path Disjointness areas all list the number of common objects (selected as disjoint path requirements) before and after the optimization, as well as disjointness violations based on these requirements before and after the optimization. You can filter to related information from these areas.

Prerequisites

- The plan file must already contain the primary LSP paths. If using the option to create disjoint primary and secondary paths, it must also at least contain secondary LSP paths, though it can contain other path options, such as tertiary.
- If creating disjoint paths between LSPs in the same disjoint group, the LSPs must first be added to the disjoint groups.

Creating Disjoint Groups

Step 1 Right-click one or more LSPs and choose **Properties**.

Step 2 Click the **Advanced** tab.

Step 3 Enter or select the disjoint group name.

Step 4 If needed, assign priorities to LSPs within these groups. Higher priority LSPs are assigned shorter routes based on the selected metric. The higher the number, the lower the priority.

Example: There are two LSPs in the same disjoint group, each with a different disjoint priority. Run the LSP Disjoint Path Optimization tool to create disjoint paths for LSPs in the same disjoint group using TE metrics as a constraint. The LSP with a disjoint priority of 1 routes using the lowest TE metric, and the LSP with a disjoint priority of 2 routes using the second lowest TE metric.

Step 5 Click **OK**.

Optimizing Disjoint Paths

Step 1 Skip this step if using tags. You can also skip this step if you want to run the tool on all SR LSPs and there are only SR LSP types, and if you are not selecting LSP paths to fix.

If you are fixing selected LSP paths or if you are running the optimization on selected SR LSPs, select them before opening the dialog box.

- If you are selecting one or the other, do so from their respective LSP Paths or LSPs tables.
- If you are selecting both, make your selection in one table, and then press and hold the Alt key (Option key on a Mac) while making selections in another table.

If the table contains both RSVP and SR LSPs, show the Type column to easily sort and find the LSP.

Step 2 Choose **Tools > LSP Optimization > LSP Disjoint Path Opt.**

Step 3 Select whether to optimize disjoint paths for all LSPs, selected LSPs, or tagged LSPs.

Step 4 Select how to route the disjoint paths.

Step 5 Select disjoint path requirements and priorities.

Step 6 Select the constraints.

Step 7 Specify whether to create and add tags to all SR LSPs rerouted during optimization. You can edit the *DSJOpt* default.

- Step 8** Specify whether to create a new plan with the results of the optimization. Unless a name is specified, WAE Design attaches an *-DSJopt* suffix to the current plan filename. If not selected, WAE Design changes the current plan file with the updated information.
- Step 9** Click **OK**.
-

LSP Loadshare Optimization

The LSP Loadshare Optimization tool automates the process of finding and setting the most favorable loadshare ratios across parallel LSPs to balance traffic and avoid congestion. The optimizer only includes interfaces that use parallel LSPs in the optimization. You can further limit parallel LSP interfaces to only those on which you want to drive down the maximum utilization.

Upon completion, WAE Design tags the LSPs with *LSPLoadshare* and generates a new plan file with an *-Lshare* suffix. This plan file opens, showing the LSPs table that is filtered to these rerouted (and newly tagged) LSPs. Saving this plan file then simplifies the process of identifying which LSPs to reconfigure in the network. Both the tags and the new plan filenames are editable.

Also upon completion a report is generated, identifying how many LSPs and interfaces were affected, the number of bins used, the resulting maximum interface utilization, and if applicable, the number of interfaces with utilization over the specified threshold.

Running the LSP Loadshare Optimization Tool

- Step 1** Open a plan file and choose one or more LSPs. If you do not choose any LSPs, the tool uses all LSPs in the plan file.
- Step 2** Choose **Tools > LSP Optimization > LSP Loadshare Opt**. (See [Table 28: LSP Loadshare Optimization Options](#), on [page 317](#) for field descriptions.)
- Step 3** (Optional) Choose the LSPs on which you want to optimize the Loadshare value.
- Step 4** Choose the interfaces on which you want to drive down utilization:
- If you choose the default (All), the optimizer uses all interfaces in the plan file.
 - Choose only those interfaces of interest on which to drive down the maximum utilization.
- Step 5** Choose which method of optimization to use:
- Minimize max interface util
 - Minimize number of interfaces with util > ___ %
- Step 6** If distributing the traffic to bins, enter the total number of flow bins to use. (See the [Bin Example](#), on [page 320](#).)
- Step 7** (Optional) Choose which traffic level to optimize.
- Step 8** (Optional) Change the LSP tag name.
- Step 9** (Optional) Choose whether to create a new plan file based on the results, and change how to name it.
- Step 10** Click **OK**.
-

Optimization Options

The LSP Loadshare options help balance traffic and avoid congestion.

Table 28: LSP Loadshare Optimization Options

Field	Description
Set loadshare on LSPs	Sets the loadshare setting between the selected parallel LSPs in the network. Parallel LSPs are those with the same source and destination nodes. By default, the optimizer assumes you want to optimize all LSPs. However, you can also choose a subset or certain selected LSPs to optimize for load sharing purposes. The optimizer does not set a loadshare value of 0 on LSPs.
Interfaces	Specifies whether to consider all interfaces or just a subset of these interfaces on which to drive down the maximum utilization. You can choose them using tags or by choosing a subset of them from the WAE Design tables.
Minimize max interface util	Minimizes the maximum interface utilization over all interfaces on the LSP routes. WAE Design tries to minimize the number of loadshare parameter changes required to do so.
Minimize number of interfaces with util > ___%	Minimizes the number of interfaces that have a utilization greater than the specified value. This is a looser constraint than minimizing the maximum interface utilization. Therefore, WAE Design has the opportunity to modify as few loadshare values as possible, thus reducing the amount of reconfiguration required.
Number of flow bins	Routers typically cannot divide flows arbitrarily between parallel LSPs, but instead allocate them to a fixed number of “bins” of approximately equal size. The bins are then divided between the parallel LSPs. This option lets you specify the total number of bins, which in turns places a constraint on the traffic division.
Traffic Level	Specifies the traffic level to use in the optimization. This is only required if more than one traffic level is present.
Tag changed LSPs	Specifies a tag for any changed LSPs. By default, the optimizer tags upgraded circuits with the label <i>LSPLoadshare</i> .
New plan for result	Specifies a name for the new plan file. By default, the optimizer appends the input plan file with <i><plan_file_Lshare></i> .

Minimization Example

The base plan for this example, *Acme_Network*, contains two sets of parallel LSPs, each with a Loadshare value of 1 and each using strict hops on named paths to reach its destination. All of these LSPs have a Traff Sim of 3000 Mbps. The interfaces to which these LSPs filter all have a Traff Sim value of 3000 Mbps, except for *sjc-to-okc*, which has 6000 Mbps of simulated traffic ([Figure 119: Example Acme Network Before LSP Loadshare Optimization, on page 318](#)).

- Using the *Acme_Network* plan file ([Figure 119: Example Acme Network Before LSP Loadshare Optimization, on page 318](#)), if you minimize the maximum interface utilization across all LSPs, all four Loadshare parameters change, and the maximum interface utilization is 40% ([Figure 120: Example Acme](#)

Network After Minimizing Maximum Interface Utilization, on page 319). The default plan filename and default LSP tags are used, thus naming the new plan Acme_Network-Lshare and the tags as LSPLoadshare.

- Using the Acme_Network plan file (Figure 119: Example Acme Network Before LSP Loadshare Optimization, on page 318), if you minimize the number of interfaces with utilization greater than 50%, only two of the Loadshare values change (Figure 121: Example Acme Network After Minimizing Number of Interfaces with Utilization 50%, on page 319). Notice in the figure how Loadshare values are relative to the other Loadshare values for parallel LSPs, and not to all LSPs in the table. Also, only the affected LSPs are tagged.

Figure 119: Example Acme Network Before LSP Loadshare Optimization

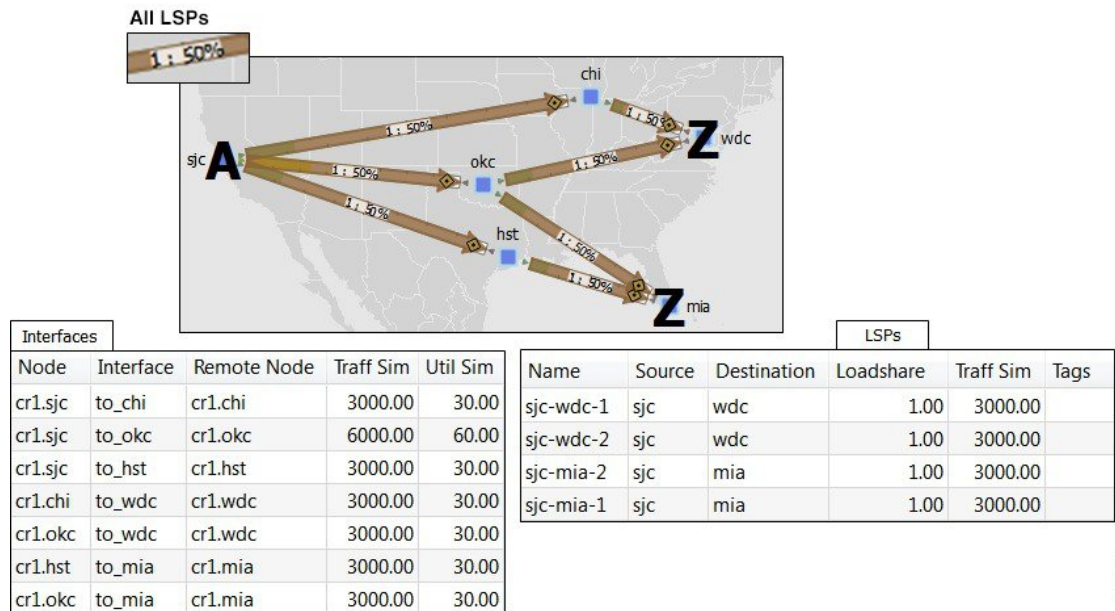
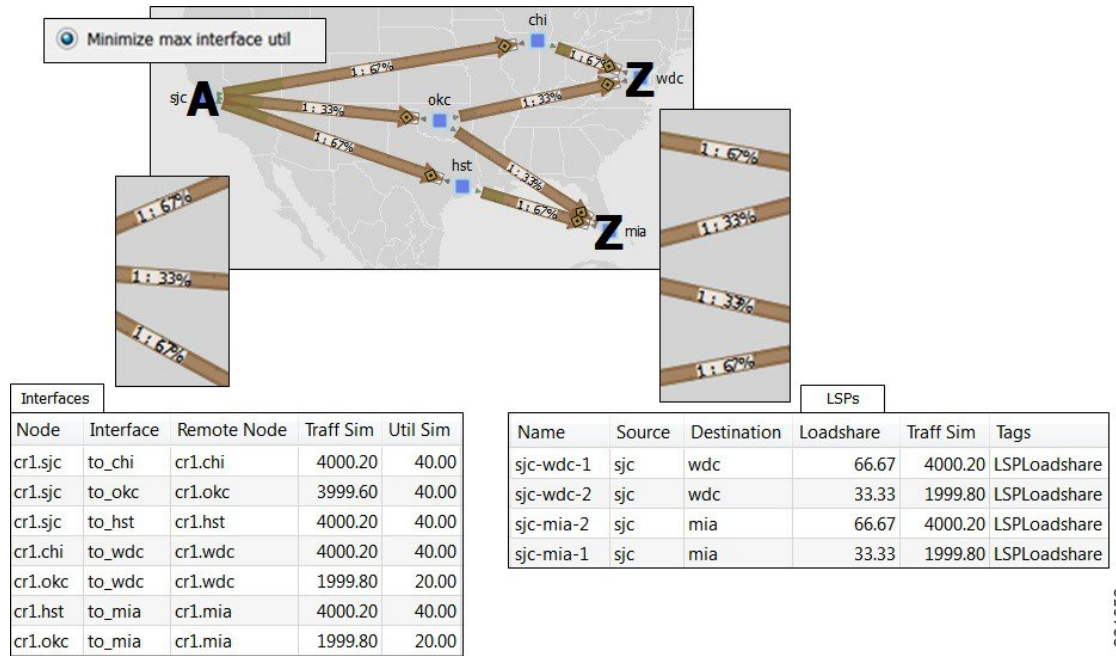
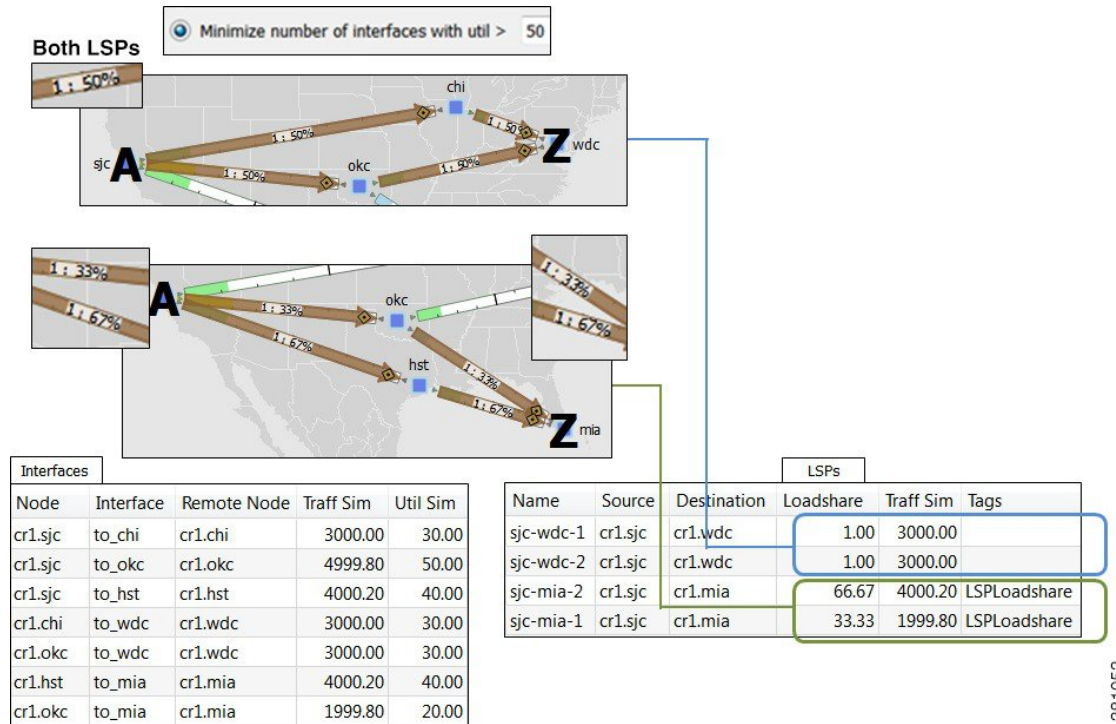


Figure 120: Example Acme Network After Minimizing Maximum Interface Utilization



381052

Figure 121: Example Acme Network After Minimizing Number of Interfaces with Utilization 50%



381053

Bin Example

In this example, a node uses a maximum of 32 bins, and the optimal traffic allocation is 45% of the traffic through LSP A and 55% of the traffic through LSP B. The node that sources these two LSPs, however, cannot do this exact split. The optimization divides the traffic into 32 bins, each with the same amount of traffic in them. Thus, each bin has 3.125% of the traffic (100% of the traffic divided by 32). The optimization also determines how the node should split the traffic. In this case, the split is to give 43.75% (which is 14 bins of 3.125% each) to LSP A and 56.25% (which is 18 bins of 3.125% each) to LSP B. Thus, it optimizes and distributes 32 (14+18) bins of traffic. This is as close as possible to the optimal 45%/55% split using 32 bins.

LSP Setup Bandwidth Optimization

As a network operator, you might need to periodically update LSPs as traffic changes in your network. You might have some fixed rules around the maximum setup bandwidth for LSPs. Restrictions on the setup bandwidth per LSP affect the number of LSPs in the network. As the setup bandwidth increases, the number of LSPs that you have to manage in your network is reduced. However, the risk that certain LSPs cannot be routed also increases. As the setup bandwidth decreases, more alternative paths can be found, allowing for better load balancing. This holds true both in the fail-free and under failures cases.

To address these LSP setup bandwidth requirements, WAE Design includes an LSP Setup BW Optimization tool (**Tools > RSVP LSP Optimization > LSP Setup BW Opt**).

Workflow

The LSP Setup BW Optimization tool lets you add or remove LSPs based on set bandwidth requirements that you specify. You can select the LSPs that you want the optimizer to consider. The optimizer then separates these LSPs into different groups. The optimizer defines an LSP group based on the fact that the LSPs within the group share common source and destination nodes. You can also specify additional custom groupings to get a finer granularity on the setup bandwidth.

The optimizer lets you create LSPs by specifying:

- The number of LSPs to create for each LSP group.
- The maximum setup bandwidth per LSP. In this case, the minimum number of LSPs is created for each group in order to meet this requirement.

Additionally, the optimizer lets you remove LSPs by specifying:

- The number of LSPs to remove for each group.
- The minimum setup bandwidth per LSP. In this case, the minimum number of LSPs is removed from each group in order to meet this requirement.

You can then perform different operations on each group to test different design scenarios.



Note The sum of the setup bandwidth of the LSPs per group is not modified by adding or removing LSPs. The setup bandwidth is evenly redistributed among the LSPs within each LSP group.

-
- Step 1** Choose **Tools > RSVP LSP Optimization > LSP Setup BW Opt**.
- Step 2** Choose the LSPs you want the optimizer to consider. The default is to operate on all LSPs. If you want to consider just certain LSPs, choose a subset of the LSPs or choose tags to specify certain LSPs.
- Step 3** Choose the manner in which LSPs are separated into LSP groups:
- Source/Destination Nodes—LSPs with a common source and destination belong in the same group.
 - Source/Destination Nodes and existing SetupBWOpt::Group Entries—LSPs with a common source and destination and common SetupBWOpt::Group entries belong in the same group. This option is useful when you want to group LSPs based on different service classes.
- Step 4** For each LSP group, provide your setup bandwidth requirements:
- Create x LSPs—Enter a positive integer. The optimizer creates a certain number of LSPs per group; for example, 1.
 - Create the minimum number of LSPs so that Setup BW $\leq x$ Mbps—Specify the minimum number of LSPs to meet your setup bandwidth rule. For example, the bandwidth rule could be 10000 Mbps.
 - Remove x LSPs—Enter a positive integer. The optimizer removes a certain number of LSPs per group; for example, 1.
 - Remove the minimum number of LSPs so that Setup BW $\leq x$ Mbps—Specify the removal of the minimum number of LSPs to meet your setup bandwidth rule; for example, 10000 Mbps.
- Step 5** (Optional) Override the defaults for how LSPs are tagged (SetupBWOpt) or for how new optimized plan files are saved.
- Step 6** Click **OK**.
-

Summary Report

After running the LSP setup bandwidth optimization, the report summarizes the total number of:

- LSPs you selected.
- LSP groups identified.
- LSPs created.
- LSPs removed.

LSP Groups Report

In the Reports window, click **LSP Groups** to view details on how the optimizer operated on LSPs:

- Group—Name of the LSP group as specified by SetupBWOpt::Group. If this group is not specified, the Group column is empty.
- LSP Source—Source node of the LSP group.
- LSP Destination—Destination node of the LSP group.

- Total Setup BW—Sum of the setup bandwidth values of the LSPs within the LSP group.
- LSP Number Before—Number of LSPs within the LSP group before running the optimization.
- LSP Number After—Number of LSPs within the LSP group after the optimization.
- LSP Setup BW Before—Average setup bandwidth within the LSP group before the optimization.
- LSP Setup BW After—Setup bandwidth within the LSP group after the optimization.



CHAPTER 20

RSVP-TE Optimization

The RSVP-TE Optimization tool (**Tools > RSVP LSP Optimization > RSVP-TE Opt**) creates explicitly routed LSP paths to optimize route properties such as distance, TE metrics, utilization, and disjointness. The routing decisions are centralized within the tool, in contrast to traditional RSVP-TE routing, in which routing decisions are distributed across the head-end routers in the network. Although it has fewer parameters, this tool is faster and has a more predictable run-time than the Explicit LSP Optimization tool and is the recommended method for explicit RSVP-TE LSP path optimization.

The primary optimization goal is to minimize path distance (which can be based on interface TE metrics, delay, or user-defined value), while not exceeding a *bandwidth bound* defined for each interface so as to avoid congestion. Unlike RSVP-TE, LSPs are routed even if they cannot avoid exceeding bandwidth bounds for certain interfaces, though these bound violations are minimized. Additionally, you can use this tool for reoptimizing and reconfiguring RSVP-TE LSPs after events, such as network failures and subsequent route reconvergence.

You can specify different treatment for different LSPs: whether the LSPs should be rerouted to shorten the path distance, only as necessary to reduce congestion, or not optimized at all. These three LSP groups are called Opt, Fit, and Fix. Using these input parameters allows for a variety of use cases, such as:

- Global optimization—Minimize the path length of all LSPs subject to specified interface bandwidth constraints.
- New LSP optimization—Minimize the path length of newly created LSPs without exceeding bandwidth bounds specified for the interfaces. Existing LSPs are kept on their current paths, being moved only if necessary to prevent new LSPs from violating bandwidth bound constraints.
- Tactical congestion mitigation—Change the routing of as few LSPs as possible to bring bandwidth on all interfaces below the specified bandwidth bound.

Upon completion, WAE Design generates a new plan file with a suffix, which by default is *RSVPTEOpt*. This suffix is configurable in the “New plan for result” field. As such, the tool is beneficial for offline planning and configuration of an explicitly routed RSVP-TE network. It is also useful for providing routing decisions for a centralized, reactive network control application, such as can be implemented on the WAE platform.

This optimized plan file opens showing the LSPs table filtered to the rerouted LSPs. Saving this plan file then simplifies the process of identifying which LSPs to reconfigure in the network.



Note RSVP-TE Optimization tool supports both Inter-Area and Inter-AS functionalities.

This section contains the following topics:

- [Optimization Input, on page 324](#)
- [Optimization Output, on page 328](#)
- [Examples, on page 329](#)
- [Workflow, on page 333](#)

Optimization Input

The RSVP-TE Optimization tool uses the LSP and interface input properties listed in [Table 29: RSVPTEOpt Input Parameters, on page 324](#) to execute its optimization calculations. These properties, which are specified in user columns in the LSPs and Interfaces tables, can be created in one of two ways:

- Use the tool to automatically create (initialize) the properties if they do not exist. The selections made in the Specify LSP Groups, LSP Parameters, and Interface Parameters areas populate the input properties.

Example: If you choose LSPs to optimize, to fit, and to fix, the tool creates the RSVPTEOpt::Group column in the LSPs table and sets this property to the appropriate values for each LSP (Opt, Fit, Fix). If the property already exists, it is updated with the new value if it has changed.

Example: If you select to set the interface bandwidth bound based on the Resv BW property, the tool creates the RSVPTEOpt::BWBound column in the Interfaces table and for each interface, copies the Resv BW value to it. If the property already exists, it is updated with the new bandwidth bound value if it has changed.

- Before running the tool, manually create the input properties as user columns in the LSPs and Interfaces tables. When running the tool, you must then select the “Use Existing RSVPTEOpt::<property_name>” option. This is useful if you want to configure specific values rather than using existing properties and values. For information on creating user-defined columns, see [User Interface, on page 5](#) or the *Cisco WAE Design Integration and Development Guide*.



Note Because the resulting plan file can potentially reset properties that can be used as input to optimizations, it is a best practice to run the optimization only on plan files or LSPs that have not yet been optimized. For example, by default WAE Design resets the LSP Setup BW property to 0 after optimizing LSPs, and this same property could be used as input to other optimizations.

Table 29: RSVPTEOpt Input Parameters

Table	Input Property	Description and Values Used to Create the Input Property
LSPs	RSVPTEOpt::Group	LSP optimization group. The input property is derived based on your entries in the Specify LSP Groups area unless you select to use the existing RSVPTEOpt::Group property. Values are Opt, Fit, or Fix. If the value is not defined, it is set to Ignore.
	RSVPTEOpt::BWReq	The bandwidth required by each LSP that is being optimized. The input property is derived from the Setup BW, Traff Meas, or Traff Sim LSP properties unless you specify to use the existing RSVPTEOpt::BWReq value.

Table	Input Property	Description and Values Used to Create the Input Property
Interfaces	RSVPTEOpt::BWBound	Sum total of LSP bandwidth routed over the interface should not exceed this value. The input property is derived from the Resv BW, Capacity, or Capacity Sim interface properties unless you specify to use the existing RSVPTEOpt::BWBound value.
	RSVPTEOpt::Metric	Metric value used in the shortest path calculations. The input property is derived from either the TE Metric or Delay property of the interface unless you specify to use the existing RSVPTEOpt::Metric value.

Selecting LSP Groups

To determine how and whether LSPs are optimized, select them from the Specify LSP Groups area. If an LSP is in multiple groups, the first group in which it resides is how the LSP is treated.

- **Opt**—Explicitly route or reroute these LSPs to minimize their path length, while respecting the bandwidth bounds specified for the interfaces. The RSVP-TE Opt tool changes the LSP path as much as needed to find the optimal route. First, it tries to set the LSP to its shortest path, which is defined by the interface Metric parameter. The LSP is moved away from the shortest path only if needed due to congestion.

Setting all LSPs to this option is useful when trying to achieve global optimization on all LSPs.

- **Fit**—Reroute these LSPs only as necessary to respect specified bandwidth bounds on the interfaces, while accommodating the Opt group. If there is currently no explicit route or if the route is incomplete, treat like Opt LSPs.

This option is useful when attempting tactical congestion mitigation, where it is desirable to move as few LSPs as possible to remove existing congestion.

A combination of Opt and Fit or a combination of Opt and Fix is useful for targeting the optimization of new LSPs.

- **Fix**—Do not reroute these LSPs, but consider them in the optimization calculations.

LSPs that are not selected are ignored, and as such, maintain their original configuration and are not considered in the optimization calculations. This can be useful if you have network regions that do not interfere with optimizations. Note, though, that if measured interface traffic is used in the optimizations, traffic contributions from the ignored LSPs to this measured traffic might affect the results.

Because you can set each of the “Set RSVPTEOpt::Group” selections to None, All, only selected, or tagged LSPs, you can section off specific LSPs for one optimization treatment or another.

Example: Tag new LSPs and set them to be optimized (Opt LSPs). Select and set existing LSPs to Fit LSPs to optimize their paths only as necessary to accommodate bandwidth bound constraints.

Setting Optimization Parameters

Optimization calculations are based on two sets of parameters that define how much bandwidth the LSPs require, how much bandwidth the interfaces have available for these LSPs, and which metric to use for shortest path. See [Table 29: RSVPTEOpt Input Parameters, on page 324](#) for selection options.

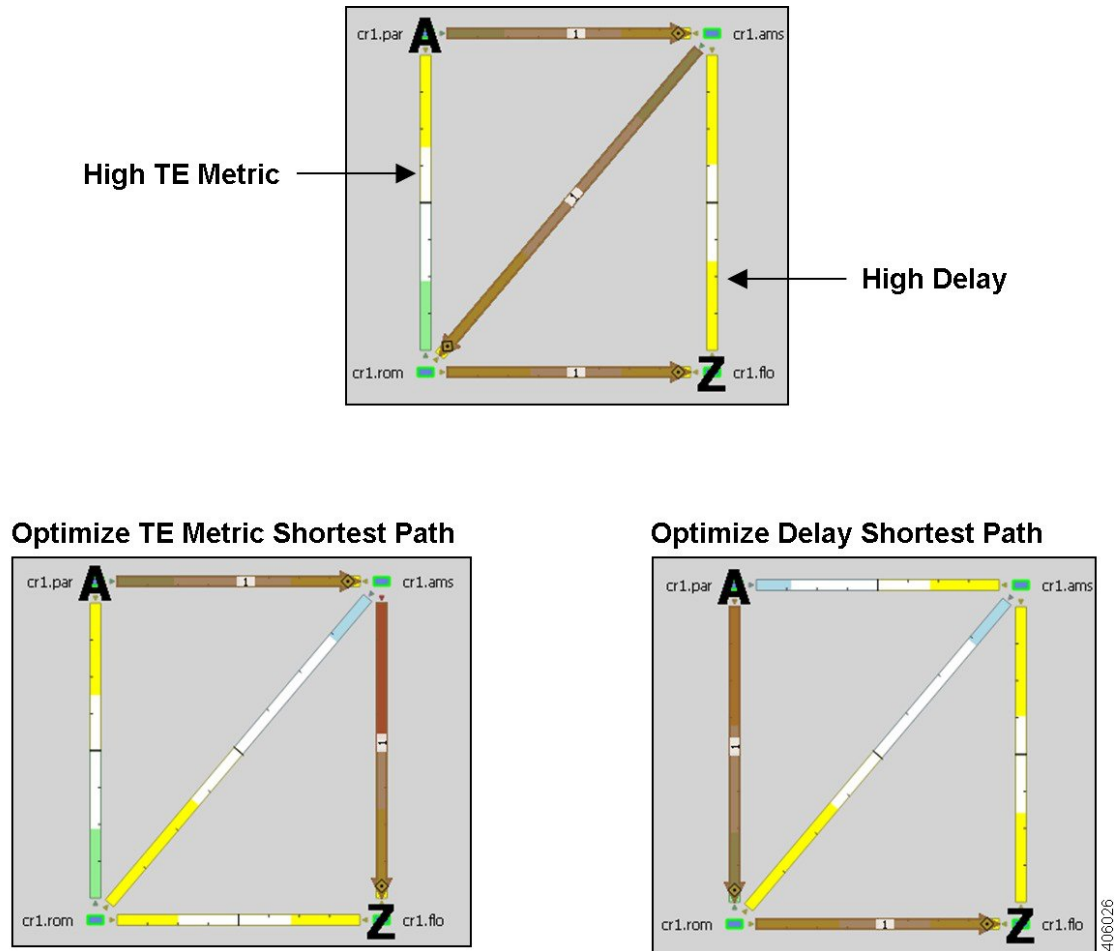
The screenshot shows a configuration window with two main sections: "LSP Parameters" and "Interface Parameters".

- LSP Parameters:** Contains a "Required BW" dropdown menu currently set to "Setup BW".
- Interface Parameters:** Contains two dropdown menus:
 - "BW Bound" is set to "Resv BW".
 - "Metric" is currently set to "TE Metric", and its dropdown menu is open, showing options: "TE Metric" (highlighted), "Delay", and "Use existing RSVPTEOpt::Metric".

A vertical ID number "406031" is visible on the right side of the configuration window.

- LSP Required BW—Specify how to determine the bandwidth required for the primary LSP paths.
- Interface parameters:
 - BW Bound—Specify how much bandwidth the interfaces have available to carry LSPs. WAE Design tries to route LSPs without surpassing this bound, but exceeds it if necessary.
 - Metric—Specify which metric to use for shortest path calculations. [Figure 122: Example Shortest Path Optimization, on page 327](#) shows a sample LSP path that is optimized once based on the TE Metric and once based on the Delay.

Figure 122: Example Shortest Path Optimization



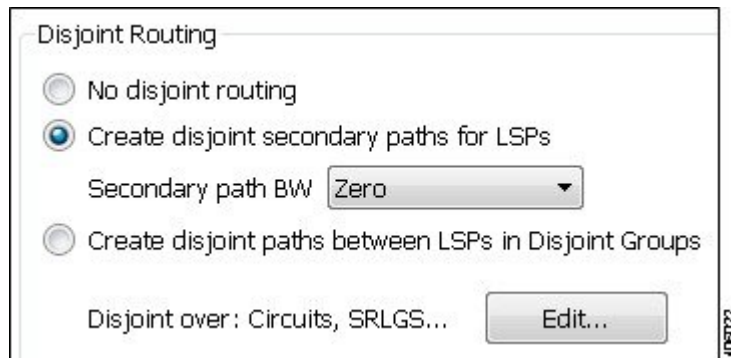
Setting Disjoint Groups

In the Advanced tab, you can create disjoint paths or use existing disjoint paths as follows. Note that two LSP paths are *disjoint* if they do not route over common objects. These objects are configurable and can be circuits, nodes, sites, SRLGs, or L1 links.

- No disjoint routing—Do not create disjoint secondary paths and do not enforce any disjointness on primary paths.
- Create disjoint secondary paths for LSPs—Create secondary LSP paths that are disjoint from the primary LSP paths.

Additionally, specify whether to include secondary paths in the optimization. If set to Zero, their bandwidths are set to 0 and are not included. Otherwise, their required bandwidths are set to be the same as the primary paths, and they are included in the optimization.

- Create disjoint paths between LSPs in Disjoint Groups—Create disjoint paths between LSPs that are in disjoint groups.



Setting Avoidance Constraints

Avoid Objects section in the Advanced tab lets you to select Nodes, Interfaces, L1 Links, L1 Nodes, SRLGs to be avoided when optimizing LSPs.

- Nodes - LSPs are created to route away from the Nodes selected. Default is None.
- Interfaces - LSPs are created to route away from the Interfaces selected. Default is None.
- L1 Links - LSPs are created to route away from the L1 Links selected. Default is None.
- L1 Nodes - LSPs are created to route away from the L1 Nodes selected. Default is None.
- SRLGs - LSPs are created to route away from the SRLGs selected. Default is None.

Setting Post-Optimization Parameters

The Post Optimization options identify how to reset the Setup BW for modified LSPs:

- Reset Setup BW to 0.
- Reset Setup BW to be the same as the resulting RSVPTEOpt::BWReq value.
- Keep the Setup BW the same.

Optimization Output

Properties Created

In addition to creating the input parameters, as described in [Optimization Input, on page 324](#), the tool generates the properties listed in [Table 30: RSVPTEOpt Parameters Used as Optimization Input, on page 329](#). These properties provide further insight into the optimizations.

Table 30: RSVPTEOpt Parameters Used as Optimization Input

Table	Property	Description
LSPs	RSVPTEOpt::Action	Identifies whether the LSP's path was updated (optimized).
	RSVPTEOpt::PathMetric	Sum of interface metrics on the LSP path.
	RSVPTEOpt::ShortestPath	Identifies whether the LSP is now taking the shortest path.
	RSVPTEOpt::ShortestPathMetric	Sum of interface metrics on the shortest LSP path.
Interfaces	RSVPTEOpt::BWTotal	Sum of the bandwidth required for all LSPs routed on the interface.
	RSVPTEOpt::BWBoundExceeded	Identifies whether the bandwidth bound of the interface was exceeded.

Optimized LSPs Reconfigured

The optimized LSPs are reconfigured with the following parameters:

- Metric Type—Autowrite
- Metric and Hop Limit—na
- Setup BW—0, the same value as the resulting RSVPTEOpt::BWReq, or unchanged, depending on how you configured the Post Optimization option in the Advanced tab
- Primary paths Standby—T (true)
- Status—Active
- FRR Enabled—T (true)
- LSP entry in the Actual Paths table is removed

Reports

WAE Design writes a report containing the results of the optimization. To access this information, choose **Window > Reports**.

The report summarizes the results of the optimization, such as how many LSPs there are total compared to how many were optimized, as well as how they were optimized. It also summarizes how disjoint priorities were handled and the number of paths that exceed the interface's bandwidth bound.

Examples

Optimizing New LSPs Example

[Figure 123: Non-Optimized LSP Routes, on page 330](#) shows three LSPs: LSP_A is configured with 4000 Mbps of Setup BW, LSP_B with 5000 Mbps of Setup BW, and LSP_C, which represents a newly added LSP, with 6000 Mbps of Setup BW. Each interface has a Resv BW value of 10,000 Mbps. Without optimization,

this newly added LSP_C is using three hops to reach cr1.bos because this is the only path with sufficient bandwidth.

Figure 123: Non-Optimized LSP Routes

Interfaces			
Node	Interface	Resv BW Sim	Remote Node
cr1.chi	to_cr1.nyc	10000.00	cr1.nyc
cr1.nyc	to_cr1.chi	10000.00	cr1.chi
cr1.wsh	to_cr1.bos	10000.00	cr1.bos
cr1.bos	to_cr1.wsh	10000.00	cr1.wsh
cr1.chi	to_cr1.wsh	10000.00	cr1.wsh
cr1.wsh	to_cr1.chi	10000.00	cr1.chi
cr1.nyc	to_cr1.wsh	10000.00	cr1.wsh
cr1.wsh	to_cr1.nyc	10000.00	cr1.nyc
cr1.nyc	to_cr1.bos	10000.00	cr1.bos
cr1.bos	to_cr1.nyc	10000.00	cr1.nyc

LSPs			
Name	Source	Destination	Setup BW
LSP_A	cr1.chi	cr1.bos	4000.00
LSP_B	cr1.chi	cr1.bos	5000.00
LSP_C	cr1.chi	cr1.bos	6000.00

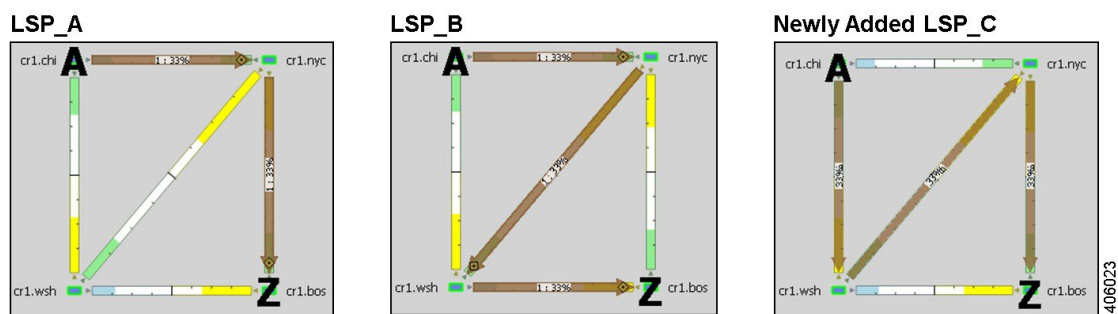


Figure 124: Example: No Optimization Improvements When Setting New LSP_C to Opt and All Other LSPs to Fix, on page 331 shows the results of prioritizing the optimization of the new LSP by setting it to Opt, and setting the other two LSPs to route “as is” by setting them to the Fix LSP group. The LSP Required BW parameter is Setup BW, and the interface BW Bound is Resv BW. While a primary path with explicit hops is created for LSP_C, all LSPs remain on the same route. This demonstrates that LSP_C cannot go on a two-hop path without exceeding bandwidth and it cannot alter this behavior by shifting the other two LSPs around because they are set to Fix. The interface bandwidth bound, however, is not exceeded, which is shown in the RSVPTOpt::BWBoundExceeded column in the Interfaces table. The LSPs table shows that only LSP_C was updated in the RSVPTOpt::Action column, and that neither LSP_B, nor LSP_C are on their shortest path. Notice also in the LSPs table that the Setup BW values are reset to 0, which is the default behavior, and that the RSVPTOpt::Group shows how each LSP was considered during the optimization (Fix or Opt).

Figure 125: Example: Improved Global Optimization When Setting All LSPs to Opt, on page 331 shows the results of an alternative optimization where all three LSPs are set to Opt. The resulting plan file shows both LSP_B and LSP_C were optimized to route on shortest paths without exceeding bandwidth bounds.

Figure 124: Example: No Optimization Improvements When Setting New LSP_C to Opt and All Other LSPs to Fix

Specify LSP Groups

Set RSVPTEOpt::Group with selections

Opt: Selected in table (1/3)

Fit: None

Fix: All

Ignore: Remaining LSPs

Use existing RSVPTEOpt::Group entries

LSP Parameters

Required BW Setup BW

Interface Parameters

BW Bound Reservable BW

LSPs				
Name	Setup BW	RSVPTEOpt::Group	RSVPTEOpt::Action	RSVPTEOpt::ShortestPath
LSP_A	0.00	Fix	na	T
LSP_B	0.00	Fix	na	F
LSP_C	0.00	Opt	Update	F

Interfaces			
Node	Interface	RSVPTEOpt::BWBound	RSVPTEOpt::BWBoundExceeded
cr1.chi	to_cr1.nyc	10000.0	F
cr1.wsh	to_cr1.bos	10000.0	F
cr1.chi	to_cr1.wsh	10000.0	F
cr1.nyc	to_cr1.wsh	10000.0	F
cr1.wsh	to_cr1.nyc	10000.0	F
cr1.nyc	to_cr1.bos	10000.0	F

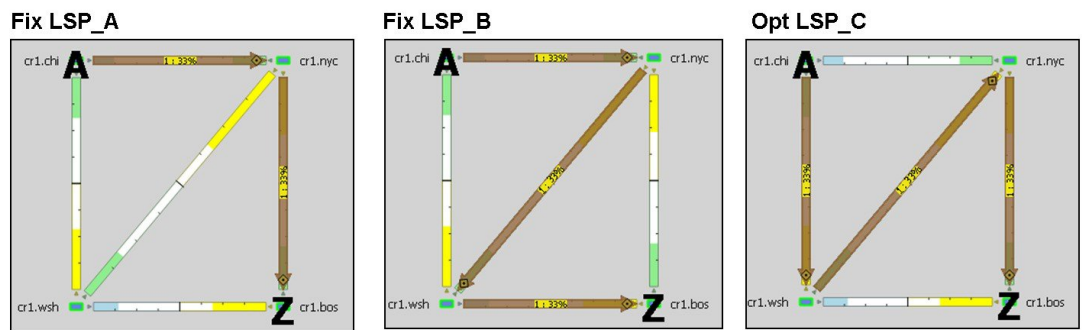


Figure 125: Example: Improved Global Optimization When Setting All LSPs to Opt

Specify LSP Groups

Set RSVPTEOpt::Group with selections

Opt: All

Fit: None

Fix: None

Opt LSP_A

Opt LSP_B

Opt LSP_C

Tactical Mitigation Example

Figure 126: Unroutable LSP Due to Insufficient Bandwidth, on page 332 shows three non-optimized LSPs: LSP_A and LSP_B have 4000 Mbps and 5000 Mbps of Setup BW, respectively. They route the same as in Figure 123: Non-Optimized LSP Routes, on page 330. LSP_C has 12,000 Mbps of Setup BW, leaving it unable to route because all interfaces have a Resv BW of 10,000 Mbps.

Figure 126: Unroutable LSP Due to Insufficient Bandwidth

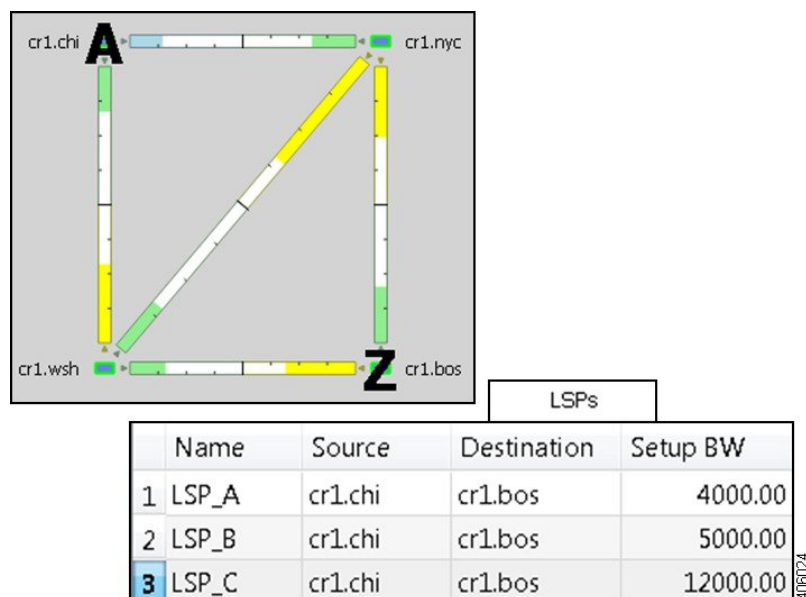


Figure 127: Example: Congestion Occurs When Settings LSPs to Fix, on page 333 shows an optimization run to set this high-Setup BW LSP_C to Opt and keep the other LSPs set to Fix. This results in congestion and leaves three interfaces exceeding their bandwidth bound.

An alternative is shown in Figure 128: Example: Tactical Congestion Mitigation Using Fit LSPs, on page 333, where congestion is reduced by running the optimization to set LSP_C to Opt, while setting the other to LSPs to Fit. This lets WAE Design adjust the routes of the other LSPs to achieve the best optimization possible (lowest shortest-path metric with the least congestion). While both LSP_B and LSP_C route on two hops now, two interfaces still exceed their bandwidth bound, though the amount of congestion is significantly less. Note that the same result would be achieved if all three LSPs had been set to Opt.

Figure 127: Example: Congestion Occurs When Settings LSPs to Fix

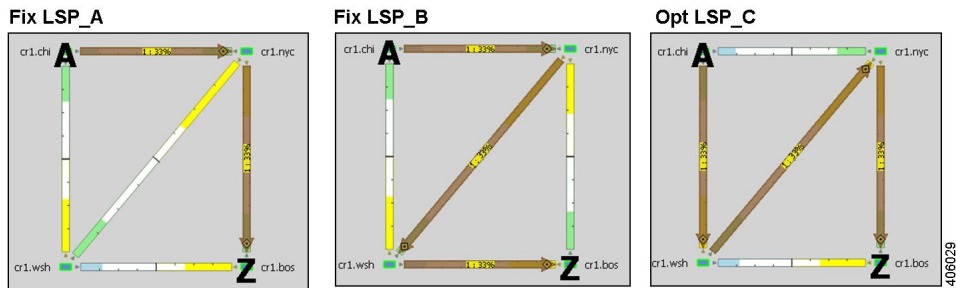
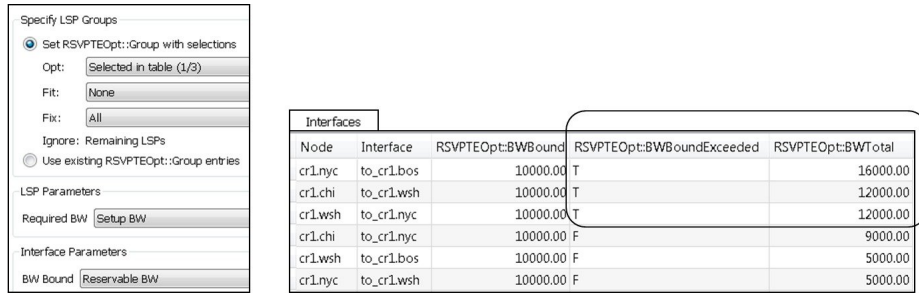
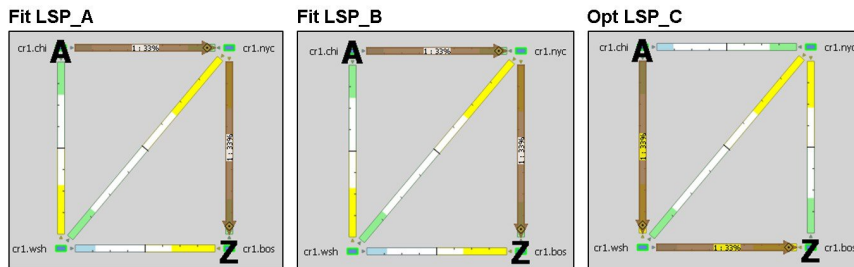
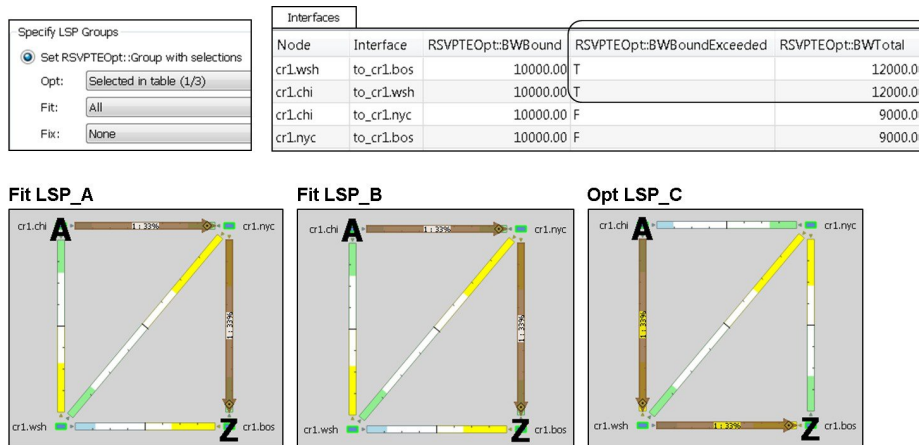


Figure 128: Example: Tactical Congestion Mitigation Using Fit LSPs



Workflow

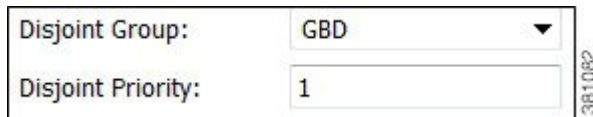
Step 1

If this is the first time using the tool and if you want to use your own preconfigured RSVPTEOpt properties, you must manually edit the LSPs and Interfaces tables to create them.

- LSP RSVPTEOpt::Group—Place LSPs into the Opt, Fit, or Fix groups. Any remaining LSPs are placed in the Ignore group.
- LSP RSVPTEOpt::RequiredBW—Define an exact amount for the sum total of LSP bandwidth that is using the interface.

- Interface RSVPTEOpt::BWBound—Define an exact amount of bandwidth that the interface can hold.
- Interface RSVPTEOpt::Metric—Define an exact amount to use in shortest path calculations.

Step 2 If creating disjoint paths between LSPs in the same disjoint group, the LSPs must first be added to the disjoint groups in the LSP Properties dialog box. Here, you can also assign priorities to LSPs within these groups. Higher priority LSPs are assigned shorter routes based on the metric being used for shortest path calculations. The higher the number, the lower the priority.



The screenshot shows a dialog box with two fields. The first field is labeled "Disjoint Group:" and has a dropdown menu with "GBD" selected. The second field is labeled "Disjoint Priority:" and has the value "1" entered. To the right of the dialog box, there is a vertical label "381082".

Step 3 If using tagged LSPs, create those tags. For information on how to create tags, see [Plan Objects](#), on page 23.

Step 4 If running the optimization based on selections, select those LSPs.

Step 5 Choose **Tools > RSVP LSP Optimization > RSVP-TE Opt.**

Step 6 In the Specify LSP Groups area, make selections for optimizing LSPs based on Opt, Fit, or Fix LSP groups, or based on those defined in the RSVPTEOpt::Group property.

Step 7 In the Required BW field, set bandwidth requirements for the primary LSP paths.

Step 8 In the BW Bound field, set how much bandwidth the interfaces can carry.

Step 9 In the Metric field, specify on which property to base the shortest path.

Step 10 (Optional) Specify the name of the resulting plan file.

Step 11 Click the **Advanced** tab.

Step 12 Identify whether to create disjoint LSP paths and if so, how you want them defined.

Step 13 Specify any objects you want to avoid (like Nodes, Interfaces, L1 Links, L1 Nodes, and SRLGs) during optimization in the Avoid Objects section.

Step 14 Specify whether to reset the LSP bandwidth after the optimization, and if so, how so.

Step 15 Click **OK**.



CHAPTER 21

Explicit and Tactical RSVP-TE LSP Optimization

The Explicit RSVP-TE LSP Optimization tool minimizes congestion by optimizing the placement of primary and secondary paths for selected RSVP LSPs. By default, WAE Design minimizes the utilization across primary paths under normal operation and creates disjoint secondary paths so that a single failure cannot disrupt both paths simultaneously.

The default is to optimize utilization on all interfaces using all RSVP LSPs on those interfaces. Another default is to remove CSPF constraints, such as affinities and hop limits. For example, WAE Design sets the setup bandwidth to 0, thus providing the greatest flexibility when setting these explicit paths.

Upon completion, WAE Design tags the RSVP LSPs with *exopt* and generates a new plan file with an *-eopt* suffix. This plan file opens, showing the LSPs table that is filtered to these rerouted (and newly tagged) RSVP LSPs. Saving this plan file then simplifies the process of identifying which RSVP LSPs to reconfigure in the network.

WAE Design also writes a report containing the results of the optimization. To access this information later, choose **Window > Reports**.

This section contains the following topics:

- [Running the RSVP-TE LSP Optimization Tool, on page 335](#)
- [Tactical Explicit RSVP-TE LSP Optimization, on page 340](#)

Running the RSVP-TE LSP Optimization Tool

-
- Step 1** Choose **Tools > RSVP LSP Optimization > Explicit Opt**. (See [Table 31: Explicit RSVP-TE LSP Optimization Options, on page 336](#) for field descriptions.)
- Step 2** Specify objectives for primary, secondary, and tertiary paths.
- Step 3** Identify which interfaces to optimize.
- The default is to optimize RSVP LSP paths on all selected interfaces, and by default, all interfaces are considered selected. You can, however, optimize a limited number of interfaces or specify a set of RSVP LSPs by preselecting them or by using tags.
- Step 4** Click **OK**.
-

What to do next

Table 31: Explicit RSVP-TE LSP Optimization Options

Field	Description
Primary Path Options	
Primary paths	<p>Define whether to reroute primary RSVP LSP paths:</p> <ul style="list-style-type: none"> • Optimized—Create optimized explicit primary paths based on the objectives selected. • Keep—Route RSVP LSPs along the existing primary paths. <p>Primary paths are optimized using the following three objectives in order of priority. The first two objectives move RSVP LSPs away from their shortest latency path in an attempt to reduce the utilizations of the most highly utilized interfaces in the network.</p>
1. Minimize # of interfaces with utilization > x %	Specify a percentage and minimize the number of selected interfaces with utilizations over that percentage under normal operation (default).
2. Minimize maximum interface utilization	Route primary paths so that the maximum interface utilization over all selected interfaces is minimized under normal operation.
3. Balance across equal latency paths	<p>Balance utilizations over lower utilized interfaces. For example, use this option to balance utilizations on parallel interfaces between two nodes if the two interfaces have the same latency.</p> <ul style="list-style-type: none"> • Utilization threshold—Keep the number of interfaces with utilization greater than this value as low as possible without significantly increasing the latencies of the primary paths. • Latency tolerance—Permit this percentage of additional latency.
Enforce latency bounds	Enforce latency bounds that can be specified for some or all demands in the plan file. If checked, this option takes precedence over all of the preceding objectives.
Secondary Path Options	
Secondary paths	<p>Define whether and how to route secondary paths.</p> <ul style="list-style-type: none"> • Optimized—Create optimized explicit secondary paths based on the objectives selected. • Dynamic—Route secondary paths dynamically. No explicit hops will be created for the path. • None—No secondary paths are created; existing paths are removed. <p>For optimized secondary paths, the objectives are used in order of priority listed.</p>
Hot Standby	Set the secondary path to be a <i>hot</i> standby, which means it is established at the same time as the primary path, rather than after the primary path fails.

Field	Description
1. Maximize primary/secondary path disjointness with respect to	<p>Define primary and secondary paths for each RSVP LSP that are disjoint with respect to circuit, SRLG, node, and Layer 1 (L1) link failures, depending on what is selected.</p> <ul style="list-style-type: none"> • Circuits—No circuit is used by both the primary and secondary paths (default = 1). • SRLGs—No SRLG is used by both the primary and secondary paths (default = 2). • Nodes—No node is used by both the primary and secondary paths (default = 3). • L1 Nodes—No L1 node is used by both the primary and secondary paths (default = 3). • L1 Links—No L1 link is used by both the primary and secondary paths (default = 3). • Traffic disjointness only—A path is acceptable even if it uses similar circuits, SRLGs, or nodes as other paths provided there is no traffic routed over the RSVP LSP when failures occur. <p>You can specify the degree of disjointness of primary and secondary paths. The lower the number, the higher the disjointness priority. For example, if it is important that the paths are node disjoint and the SRLG disjointness are less important, you could change the setting to circuits 1, nodes 2, and SRLGs 3.</p> <p>Note that the network topology sometimes makes it impossible to fulfill all the selected disjointness requirements. In this case, paths are selected that are maximally disjoint. That is, they are disjoint for as many circuits, SRLGs, nodes, L1 nodes, and L1 links as possible.</p>
2. Minimize # of interfaces with utilization > ___ %—	<p>Options 2 and 3 both operate on the selected failure scenarios (circuits, SRLGs, nodes, L1 nodes, and L1 links) listed in the “Failures to consider” options listed under 3. These choices are the failure scenarios over which the simulation is performed at the end of the optimization. Note that this selection of failure scenarios is distinct from the failure scenarios selected for the disjointness objective 1.</p> <p>Use option 2 to minimize the number of interfaces with utilizations over the specified percentage across all selected failure scenarios.</p>
3. Minimize maximum interface utilization	Use option 3 to minimize the maximum interface utilization over all interfaces and over all selected failure scenarios.
Tertiary Path Options	
Tertiary paths	<p>Define whether to create tertiary paths:</p> <ul style="list-style-type: none"> • Dynamic—Create dynamic tertiary paths. • None—No tertiary paths are created.
Hot Standby	Set the tertiary path to be a hot standby path, which means it is brought up with the primary path, rather than after the primary path fails.
Other Options	
Optimized Interfaces	Identify which interfaces to optimize: all interfaces, those selected prior to opening the dialog box, or those with specific tags.

Field	Description
Using RSVP LSPs	Specify which LSPs are candidates for rerouting. The default is all the LSPs currently routed on the selected interfaces.
Tag changed LSPs	Create and add tags to all LSPs rerouted during optimization. Using tags makes it easier to find the LSPs that need to be reconfigured in the actual network according to the optimization results. By default, newly optimized RSVP LSPs are tagged as <i>expopt</i> .
New plan for result	This default option creates a new plan with the results of the optimization. Unless a name is specified, WAE Design attaches an <i>-eopt</i> suffix to the current plan filename. If unchecked, WAE Design changes the current plan file with the updated information.
Details button	See Setting Optimization Details, on page 338 .

Setting Optimization Details

Step 1 Choose **Tools > RSVP LSP Optimization > Explicit Opt**. (See [Table 31: Explicit RSVP-TE LSP Optimization Options, on page 336](#) for field descriptions.)

Step 2 Click the **Details** button to identify less frequently used parameters, such as traffic levels and how to treat non-optimized interfaces.

- Non-optimized interfaces—You can specify whether to ignore non-optimized interfaces or set the acceptable level of utilization for them.

If setting an acceptable utilization level and if both options are selected, WAE Design uses the higher of the two. These settings are calculated on a per-interface basis.

- Acceptable utilization of optimized interfaces: ___%—This value is the same as the utilization threshold set in the Primary Path area (Minimize # of interfaces with utilization > ___%, where the default is 80). To change this value, you must change it in the Primary Path area.

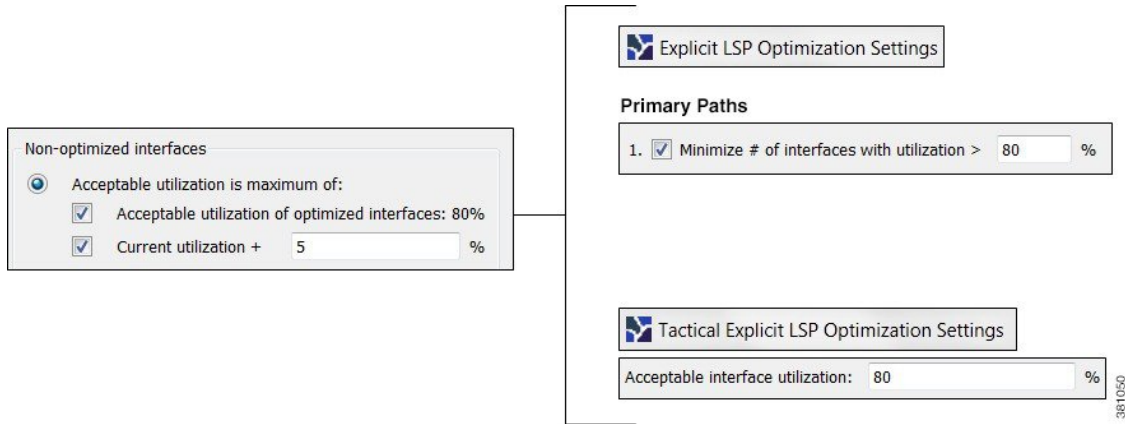
If using the Tactical Explicit RSVP-TE LSP Optimization tool, this field is equivalent to, and thus only changeable in, the Acceptable Utilization ___% field.

- Current utilization + ___%—Current utilization of non-optimized interfaces plus the added percentage.

Example: There are two non-optimized interfaces: cr1.chi_cr1.mia has a utilization of 60% and cr2.sjc_cr2.okc has a utilization of 78%.

The acceptable utilization settings for non-optimized interfaces are:

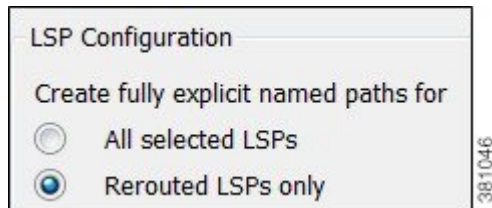
- The primary path utilization threshold is 80%.
- The current utilization has 5% added to it.



Result: The maximum utilization for each interface is individually calculated. The acceptable level of utilization for cr1.chi_cr1.mia is 80%, and the acceptable level of utilization for cr2.sjc_cr2.okc is 83% (78 + 5).

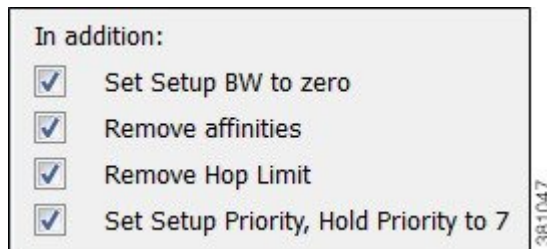
- LSP Configuration > Create fully explicit named paths for—By default, WAE Design creates named paths across the RSVP LSPs that were selected if a reroute is required to achieve the optimization objectives. For example, if a selected dynamically routed RSVP LSP has an acceptable route, WAE Design does not create a named path for it.

To change this default, click **All selected LSPs**. WAE Design then creates and routes fully explicit named paths for all selected RSVP LSPs.



- LSP Configuration > In addition—By default, WAE Design sets the Setup BW to zero, providing the greatest flexibility when creating explicit routes. As well, all affinities and hop limits are removed, and the setup and hold priorities are set to 7. These changes apply only to the RSVP LSPs with newly created or changed explicit named paths.

You can turn off these defaults, and you can also set them individually after the optimization is performed using the LSP Paths Properties dialog box. (In the LSP Paths table, right-click an RSVP LSP path and choose **Properties**.) If these defaults are turned off, the original parameters are preserved.



- Traffic Level—Choose the traffic level you want used in the utilization calculations and optimizations. For information on traffic levels, see [Traffic Demand Modeling](#).
- Rerouting Preference—By default, preferences on which RSVP LSPs to reroute are not based on traffic volume. You can select high-traffic or low-traffic options to sequentially give priority to RSVP LSPs with higher and lower traffic.

Step 3 Click **OK** to close the Details dialog box.

Step 4 Click **OK**.

Tactical Explicit RSVP-TE LSP Optimization

The Tactical Explicit RSVP-TE LSP Optimization tool is a reduced version of the Explicit RSVP-TE LSP Optimization tool. This tool optimizes only primary paths using the minimum number of path changes required to bring utilizations below an acceptable level. It is useful when you need to reduce congestion in a specific area of the network with a limited number of RSVP LSP reconfigurations.

If you need to target specific interfaces or RSVP LSPs to confine the optimization to problem areas, select those interfaces or RSVP LSPs first. Then choose **Tools > RSVP LPS Optimization > Tactical Explicit Opt.**



CHAPTER 22

Segment Routing Optimization

The [SR-TE Optimization, on page 341](#) tool creates or updates segment lists to minimize the sum total of user-specified metrics for selected SR LSPs, including inter-AS LSPs, using the fewest number of segment list hops as possible. The tool sets the routes based on traffic engineering criteria and optimizes a sequence of node or adjacency hops for the SR LSP paths to follow.

The [SR-TE Bandwidth Optimization, on page 347](#) tool lets you reduce the traffic utilization of selected interfaces to below a specified threshold. This ability to mitigate congestion is useful when planning for increased traffic and when determining if service-level agreements can be met in the event of congestion. Additionally, you can use this tool for reoptimizing bandwidth after events, such as network failures and subsequent route reconvergence.

This section contains the following topics:

- [SR-TE Optimization, on page 341](#)
- [SR-TE Bandwidth Optimization, on page 347](#)
- [Compressing Segment Lists, on page 352](#)

SR-TE Optimization

You can use the SR-TE Optimization tool to design, capacity plan, and manually configure networks to meet the following objectives:



Note SR-TE Optimization tool supports both Inter-Area and Inter-AS functionalities.

- **TE Metric or Delay Minimization**—Minimize distance between hops with respect to metrics other than IGP metrics. These can be either TE metrics configured on interfaces (which can be set proportional to circuit latency) or latencies (delays) for each circuit. An example application is a differentiated service in which latency-sensitive network traffic is routed on shortest latency paths, while the bulk of the traffic routes over cost-optimized paths.
- **Avoidance**—Create or optimize segment lists so they avoid routing through specified objects (nodes, interfaces, L1 links, L1 nodes, or SRLGs). An example application is routing pairs of LSPs, each over a different plane in a dual-plane network. The same traffic is routed over both LSPs simultaneously, thus improving availability.

While there is an option to specify a maximum number of hops, doing so might not achieve the lowest possible latency. In this case, the best achievable solution is provided.

You can additionally avoid unnecessary LSP churn by specifying boundaries (bounds) on the path length and margins within which the shortest path must be optimized.



Note Unless qualified with “TE” or “IGP,” the term *metric* in this chapter applies to IGP metric, TE metric, or delay.

Optimizing SR-TE LSPs

Choose **Tools > SR LSP Optimization > SR-TE Opt**. Then identify which SR LSPs to optimize: all SR LSPs, those selected before opening the dialog box, or those with specific tags. The default is to optimize all selected SR LSPs, and by default, all SR LSPs are considered selected.

- Parameters for optimizing explicit LSP paths are specified in the Bound, Margin, and Constraints areas of the dialog box. If you do not specify a bound, margin, or constraint, WAE Design generates the best possible SR LSP routes for the given path metric.
- A segment list is created or updated for all SR LSP paths that can be optimized using the minimum number of segment list hops possible.
 - If an LSP has a directly associated segment list, the segment list is deleted.
 - If an LSP has no LSP paths, an LSP path with a path option of 1 is created.
 - If an LSP has no segment list, but it has LSP paths, a segment list is created for the active path if it exists and if not, for the lowest path option (where lowest means the smallest value).
 - If an LSP has associated LSP paths and a segment list associated with either the active path or lowest path option, the segment list hops in the segment list are updated.

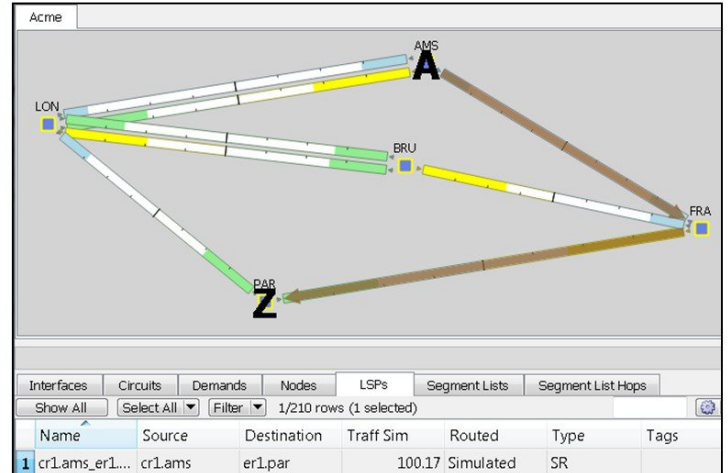
Example: [Figure 129: Example SR LSP Optimization, on page 343](#) shows an SR LSP that normally routes from AMS-FRA-PAR based on IGP metrics. It has no LSP paths. All circuits have a Delay Sim value of 50 except for the AMS-FRA circuit, which has a Delay Sim value of 60. Optimizing this SR LSP based on delay results in rerouting the SR LSP over the shortest achievable delay path, creating an LSP path with a path option of 1, and creating a segment list node hop on the cr.lon node.

- The final hop is either a node hop or interface whose remote node is the destination of the LSP.
- If it is not possible to achieve the lowest possible path without exceeding the specified maximum number of segment list hops, WAE Design provides the best achievable solution within this restriction of maximum permissible hops.
- If it is not possible to avoid all specified nodes for an SR LSP, that SR LSP is not updated.
- If an existing segment list would be better optimized by removing all segment list hops, those hops are removed, though the segment list remains.
- If a previously optimized SR LSP does not need further optimization, it is not changed and its previously set SROpt tag is removed.

- If the source and destination nodes are in different IGP areas, WAE Design creates an explicit hop on the most optimal ABR for each IGP area that the SR LSP traverses.

Figure 129: Example SR LSP Optimization

SR LSP Routed Based on IGP Metrics Before Optimization



SR LSP Route Optimized for Best Path Based on Delay

Minimize Path Metric Delay

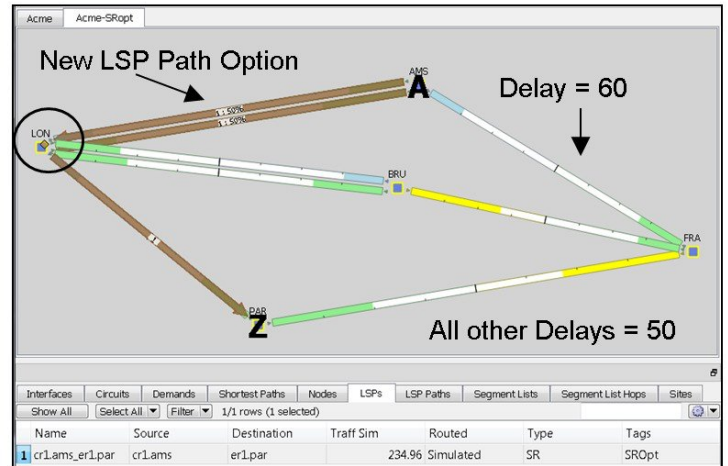
Bound on path length

Fixed

Margin above shortest path

Fixed

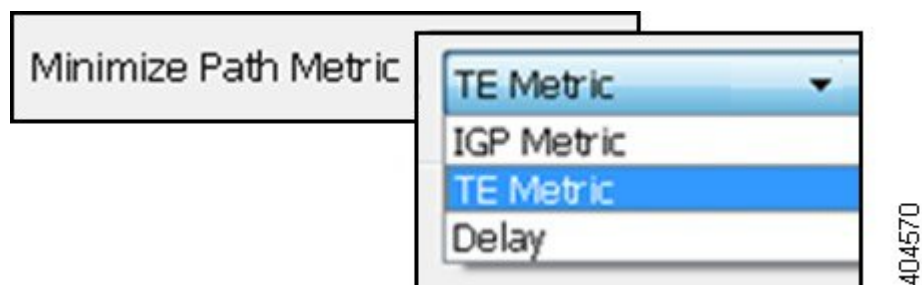
Percentage



404671

Optimizing the Path Metric

The Minimize Path Metric selection defines whether to optimize SR LSPs based on interface IGP metrics, interface TE metrics, or circuit delays. This minimization is for the sum of the metrics along the path. Note that for inter-AS SR LSPs, these metrics are calculated end-to-end for the LSP, not per AS. These properties are configurable from an interface Properties dialog box, and delays can also be set in a circuit Properties dialog box.



Bound and Margin

Bounds and margins identify which paths to optimize, as well as when to stop optimizing a given path. If you enter a value for more than one restriction, WAE Design uses the strictest limitation as the optimization target. If there is no bound or margin specified, WAE Design optimizes LSP paths to the best possible solution (lowest total metrics for the LSP path).

Bound: Maximum Acceptable Path Length

The Fixed Bound entry lets you set the maximum path metric that is acceptable. WAE Design tries to optimize LSP paths with metrics that exceed this bound. If a solution adhering to this bound cannot be found, the best possible solution is provided and bound violations are listed in the report. LSP paths that are less than or equal to this bound are not optimized.

Example: If you select to optimize LSP paths based on TE metrics, the value entered is 50, and the sum of TE metrics on the LSP path is 51, that LSP path is optimized.

Enter a number based on the selected path metric. A TE metric is a property values whose total sum for the LSP path cannot be exceeded. The delay is also a property value, but it is in milliseconds (ms). If you enter “50” and you have selected delay as the metric to optimize, this represents 50 ms as the maximum acceptable delay for the LSP path.

Margin: Maximum Acceptable Metric Above Shortest Path

The Margin entries let you identify the acceptable deviation above the shortest achievable path metric. Any existing LSP path with a metric that is less than or equal to the shortest path metric plus the margin is not optimized.

- Fixed—The amount by which a metric must be surpassed before it is optimized.

Example: If an SR LSP route has a delay of 110 (as in [Figure 129: Example SR LSP Optimization, on page 343](#)), a fixed margin set to 10, and the shortest achievable delay path is 100, the current SR LSP is within the margin and will not be updated ([Figure 130: Example SR LSPs Optimized with Margins, on page 345](#)). If the fixed margin is set to 9, the SR LSP would be optimized.

100 (shortest path) + 10 (fixed margin) = 110 , so all paths greater than 110 are optimized

- Percentage—The amount by which a metric must be surpassed, expressed as a percentage of the shortest path, before it is optimized.

Example: If an existing SR LSP route has a TE metric of 210, a percentage margin set to 10%, and the shortest achievable TE metric path is 200, the current SR LSP is within the margin and will not be updated. If the current SR LSP had a metric of 225, it would be optimized.

200 (shortest path) \times $.10$ (percentage margin) = 20 , so all paths greater than 220 must be optimized

Example: If an SR LSP route has a delay of 110 (as in [Figure 129: Example SR LSP Optimization, on page 343](#)), a fixed bound is set to 120 , a fixed margin is set to 15 , a percentage margin is set to 5% , and the shortest achievable delay path is 100 , the strictest of these restrictions takes precedence, and the SR LSP is optimized ([Figure 130: Example SR LSPs Optimized with Margins, on page 345](#)).

Fixed Bound = 120

100 (shortest path) + 15 (fixed margin) = 115

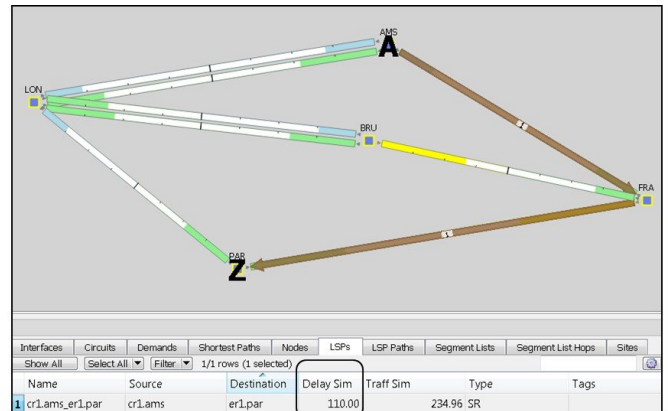
100 (shortest path) \times $.05$ (percentage margin) = 5 , so all paths greater than 105 are optimized because it is the strictest margin

Figure 130: Example SR LSPs Optimized with Margins

Shortest Delay = 100
 SR LSP Delay = 110

SR LSP Is Not Optimized

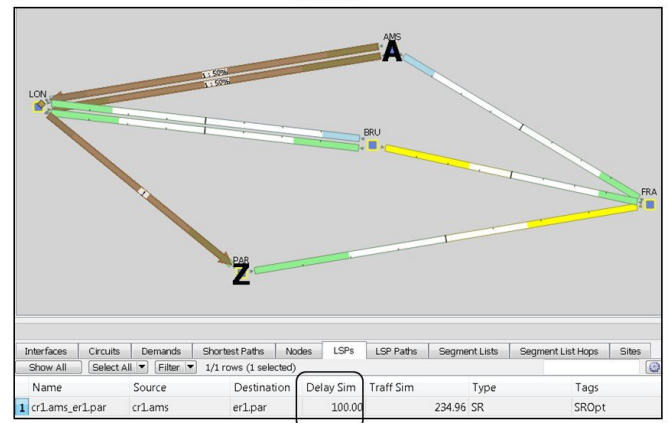
Minimize Path Metric	Delay
Bound on path length	
Fixed	
Margin above shortest path	
Fixed	10
Percentage	



SR LSP Is Optimized

Minimize Path Metric	Delay
Bound on path length	
Fixed	120
Margin above shortest path	
Fixed	15
Percentage	5

Strictest →



404572

Constraints

The constraints let you specify restrictions for the optimizations.

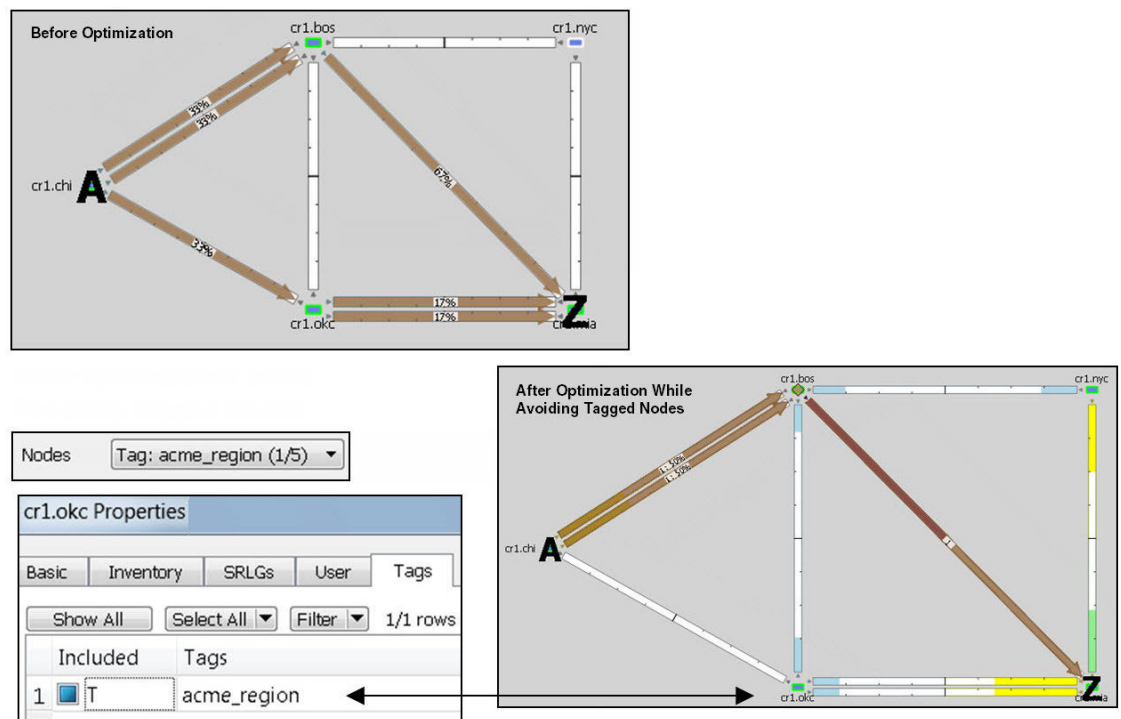
- **Maximum Segment List Hops**—The maximum number of segment list hops that any given segment list can contain after optimization. If no value is specified, WAE Design creates as many hops as needed to optimize the SR LSP.

- Avoid—Do not permit optimized segment lists to route through the selected or tagged objects (nodes, interfaces, L1 links, L1 nodes, or SRLGs). This constraint is useful when modeling dual-plane topologies that route disjoint LSPs.

Example: [Figure 131: Example of Avoiding Tagged Nodes During SR-TE Optimization, on page 346](#) shows an SR LSP from cr1.chi to cr1.mia that is routed based on its shortest IGP route. It also shows the optimized SR LSP path if it were to avoid cr1.okc based on its tag.

- Restrict segment nodes to core nodes—Segment list node hops must be core nodes (nodes that have their Function property set to “core”), and the local node of segment list interface hops must be a core node. An SR LSP could still route using edge nodes if they are not used as hops.

Figure 131: Example of Avoiding Tagged Nodes During SR-TE Optimization



Using SR Tags and Creating New Plan Files

These options identify how optimized SR LSPs are tagged, and whether to create new plan files with the results:

- Tag updated LSPs with—Create and add tags to all SR LSPs rerouted during optimization. Using tags makes it easier to find the changed SR LSPs that are deployed to the network or to find those SR LSPs that need to be reconfigured in the actual network according to the optimization results. By default, newly optimized LSPs are tagged as *SROpt*.
- New plan for result—This default option creates a new plan with the results of the optimization. Unless a name is specified, WAE Design attaches an *-SROpt* suffix to the current plan filename. If not selected, WAE Design changes the current plan file with the updated information. Saving this plan file then simplifies the process of identifying which LSPs to reconfigure in the network.

Optimization Report

Upon completion, WAE Design writes a report containing the results of the optimization, as well as verifications that the results met the criteria for avoiding nodes and meeting the specified bounds. To access this information later, choose **Window > Reports**.

SR-TE Bandwidth Optimization

The goal when reducing traffic is to reroute as few demands as possible. This is achieved by creating private SR LSPs for demands. The SR-TE BW Optimization tool also creates LSP paths and segment lists for these LSP paths with the fewest node or interface segment list hops possible, with a maximum of three hops. The final hop is either a node hop or interface whose remote node is the destination of the LSP.

The tool operates on a set of constraints that determine which paths are selected, whether demands and LSPs can be rerouted, and whether demand latency bounds can be surpassed. If the traffic cannot be reduced to below the threshold given these constraints, private SR LSPs are still created to route demands to reduce congestion as much as possible. If interface utilization surpasses the threshold prior to the optimization, it does not go higher. Utilization of non-congested interfaces might increase, though these will not surpass the threshold.

The tool reroutes existing SR LSPs only if they are private to a demand.



Note SR-TE Bandwidth Optimization tool supports Inter-Area functionality.

Operating Modes

You can run the SR-TE BW Optimization tool in two modes:

- **Analysis**—Performs multiple optimizations, one for each failure scenario within the specified failure set. The results of these optimizations are aggregated in a report.
- **Operation**—Performs an optimization using a specified set of constraints.

An output plan file is produced with the optimization results. Only a single state of the network is considered. For example, if no objects are failed in the input plan file, the optimization is performed for normal operation. If a circuit is failed in the input plan file, the optimization is performed considering this particular failure scenario.

Constraints

Both the analysis and operation modes use the same dialog box for specifying bandwidth constraints:

- **Path Selection**—When choosing demand routes, the SR-TE BW Optimization tool uses one of two criteria:
 - **Maximize Available BW**—Paths are optimized to achieve the highest available bandwidth on interfaces (Capacity Sim - Traff Sim).

- Minimize Metric—Paths are optimized to minimize the sum of the metrics along the path with respect to delay, TE, or IGP metrics. All of these properties are configurable from an interface Properties dialog box, and delays can also be set in a circuit Properties dialog box.
- Create Interface Segment Hops—If checked, the tool considers interface hops when trying to drive down utilization.
- LSPs:
 - Create new LSPs—If checked, new private SR LSPs with optimized routing can be created. If unchecked, new LSPs are not created.
 - Fix LSPs—Controls whether or not LSP routes can be modified. This constraint is useful if you want to reroute existing LSPs to mitigate congestion.
- Rerouting Demands:
 - Fix Demands—Selected or tagged demands are not rerouted. This constraint is useful, for example, when you have previously optimized specific LSPs within the network and want to maintain the route of one or more existing demands.
 - Maximum Demand Split—Use this option to split demands into smaller demands. If 'n' is the value entered, the demands are split into 'n+1' smaller demands. The default value is 0.
 - Enforce latency bounds—If checked, demands cannot surpass their configured latency bound. (This property is set in a demand's Properties dialog box.) If the latency of rerouted demands exceeds the configured bound, the tool does not reroute demands on congested interfaces.

Bandwidth Optimization Report

Upon completion of a basic optimization, the SR-TE BW Optimization tool generates a summary report to identify key changes due to the optimization. The tool tags the LSPs with *SRBWopt* and generates a new plan file with an *-SRBWopt* suffix. This plan file opens, showing the LSPs table that is filtered to these rerouted (and newly tagged) LSPs. Saving this plan file then simplifies the process of identifying which LSPs to reconfigure in the network. Both the tags and the new plan filenames are editable.

After running the bandwidth optimization, the optimizer creates a summary report of the:

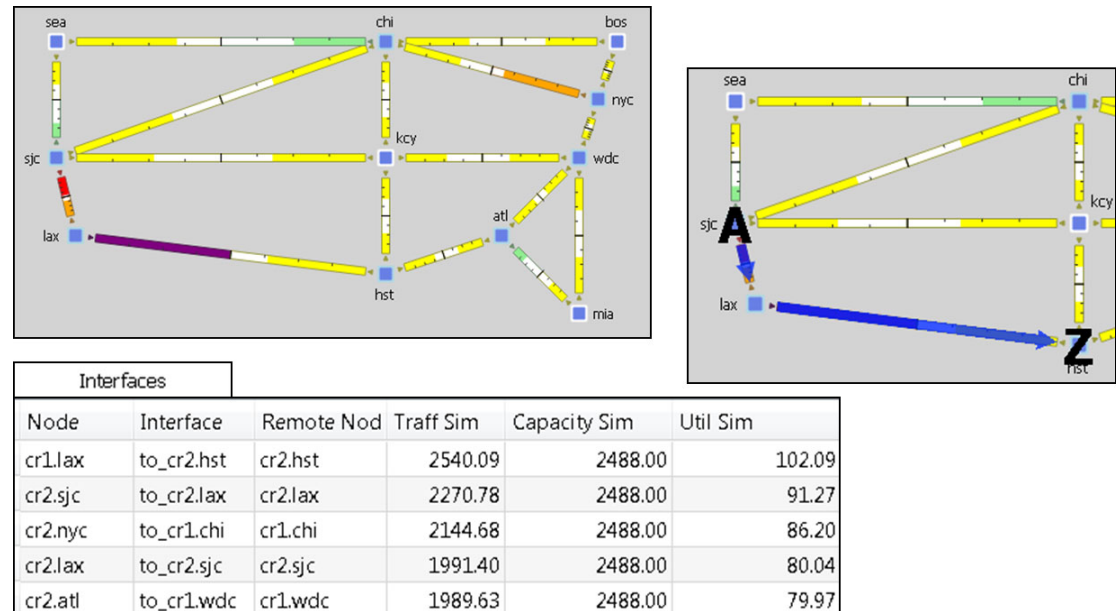
- Number of interfaces that exceeded the utilization both before and after the optimization.
- Maximum utilization both before and after the optimization.
- Number of demands that were rerouted.
- Number of LSPs that were created and rerouted.
- Average segment list length.
- Number of latency bound violations both before and after the optimization.

Examples

[Figure 132: Network Before Optimization, on page 349](#) shows a network before bandwidth optimization. It has no LSPs, and there is congestion between San Jose (sjc) and Los Angeles (lax), as well as between Los

Angeles (lax) and Houston (hst). Sorting the Util Sim column shows the top five highest utilizations with two of them higher than 90%. Notice that the demand from sjc to hst routes directly over these highly used interfaces.

Figure 132: Network Before Optimization

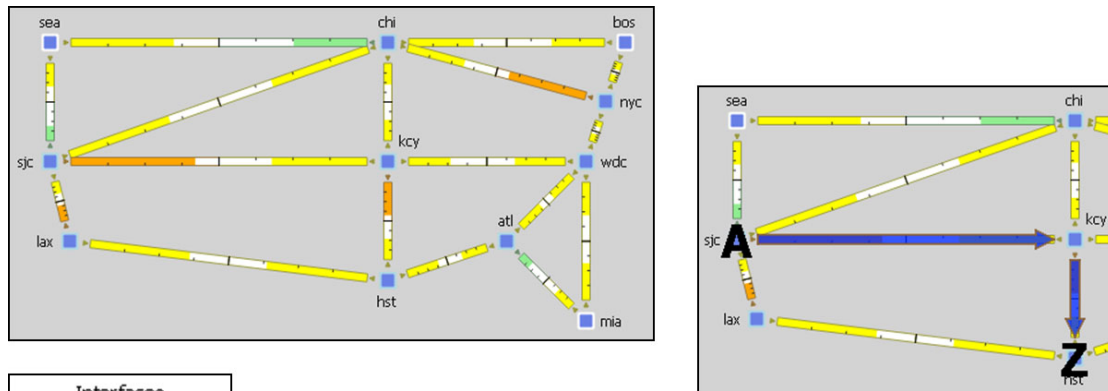


To optimize the maximum available bandwidth to be below 90% and to enforce latency bounds, the following parameters are set:

- Congestion threshold = 90%.
- Base the rerouted paths on maximizing available bandwidth.
- Enforce latency bounds.

The result of this optimization is that no interfaces have utilization over 90%, and the interfaces with the highest utilization are different interfaces than prior to the optimization. [Figure 133: Network After Reducing Utilization to Below 90%, on page 350](#) shows this redistribution of traffic across the network. The only demand rerouted was the one between sjc and hst, which you can compare to [Figure 132: Network Before Optimization, on page 349](#).

Figure 133: Network After Reducing Utilization to Below 90%



Interfaces					
Node	Interface	Remote Node	Traff Sim	Capacity Sim	Util Sim
cr2.nyc	to_cr1.chi	cr1.chi	2144.68	2488.00	86.20
cr2.kcy	to_cr1.hst	cr1.hst	2124.55	2488.00	85.39
cr1.sjc	to_cr1.kcy	cr1.kcy	2088.27	2488.00	83.93
cr2.lax	to_cr2.sjc	cr2.sjc	1991.40	2488.00	80.04
cr2.atl	to_cr1.wdc	cr1.wdc	1989.63	2488.00	79.97

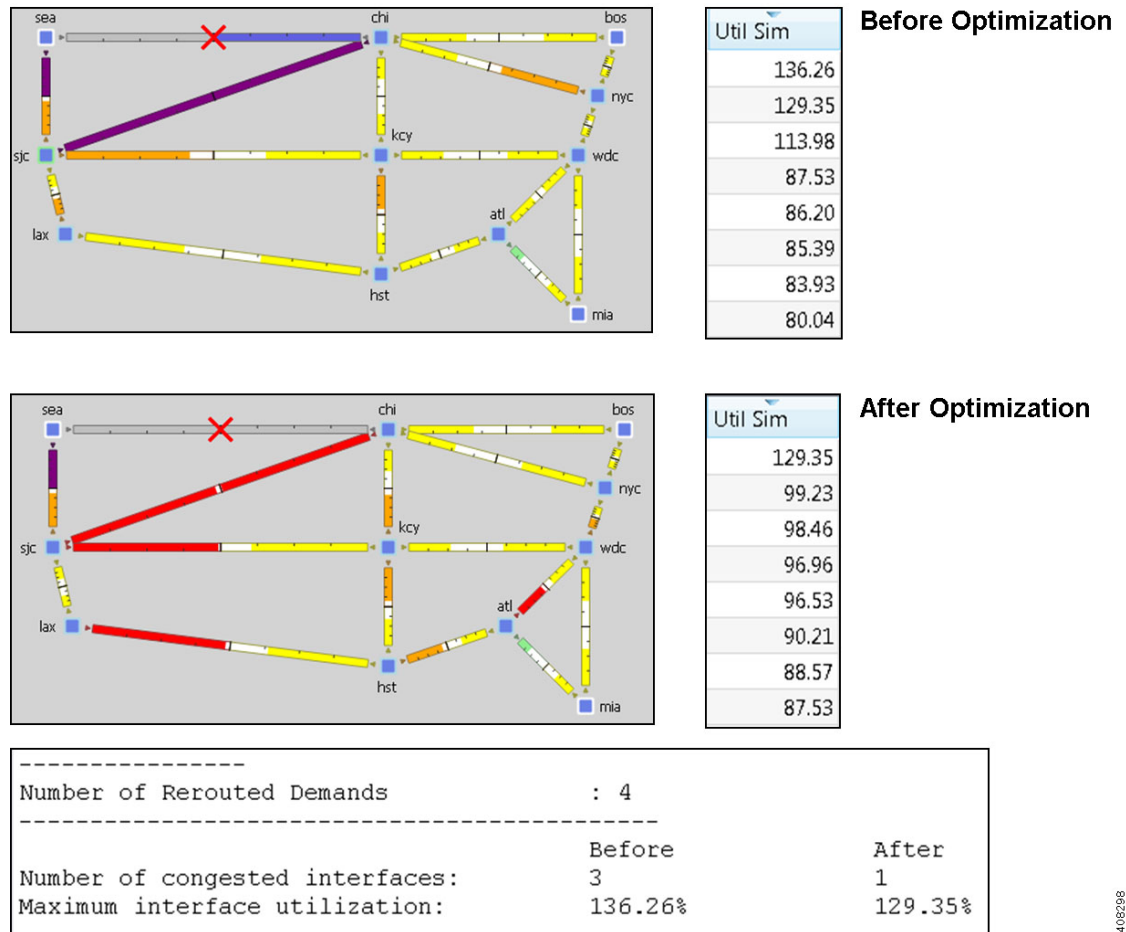
Number of Rerouted Demands	: 1	

	Before	After
Number of congested interfaces:	2	0
Maximum interface utilization:	102.09%	86.2%

Figure 134: Optimized Network Under Failure, on page 351 shows this same network, but with a failed circuit between Seattle (sea) and Chicago (chi). There are three interfaces with over 100% utilization. Here, the SR-TE BW Optimization tool was rerun using the same parameters as above except the congestion threshold was set to 100. Although congestion cannot be completely mitigated, the tool minimizes congestion as much as it can, leaving only one interface congested (over 100% utilized).

4/8/2017

Figure 134: Optimized Network Under Failure



408298

Optimize Bandwidth

Step 1 Skip this step if you are using tags. You can also skip this step if you want to run the tool on all interfaces, and if you are not selecting demands to fix.

If you are fixing selected demands or running the optimization on selected interfaces, select them before opening the dialog box.

- If you are selecting one or the other, do so from their respective Demands or Interfaces tables.
- If you are selecting both, make your selection in one table, and then press and hold the Alt key (Option key on a Mac) while making selections in another table.

Step 2 Choose **Tools > SR LSP Optimization > SR-TE BW Opt > Operation**.

Step 3 Select whether to optimize bandwidth for all interfaces, selected interfaces, or tagged interfaces.

Step 4 Specify constraints. See [Constraints, on page 345](#).

Step 5 Specify whether to create and add tags to all SR LSPs rerouted during optimization. You can edit the *SRBWopt* default.

- Step 6** Specify whether to create a new plan with the results of the optimization. Unless a name is specified, WAE Design attaches an *-SRBWopt* suffix to the current plan filename. If unchecked, WAE Design changes the current plan file with the updated information.
- Step 7** Click **OK**.
-

Analyzing Congestion Under Different Failure Sets

You can perform multiple optimizations, one for each failure scenario within a specified failure set. The results of these optimizations are aggregated in a report. This way, you can evaluate the ability of the optimizer to mitigate congestion under different failure scenarios.

- Step 1** Choose **Tools > SR LSP Optimization > SR-TE BW Opt > Analysis**. By default, the analysis tool presents you with the Basic bandwidth optimization options.
- Step 2** Click the **Failure Set** tab.
- Step 3** Select failure sets that you want the optimizer to consider (circuits, nodes, sites, and so on).
Entries appear dimmed if they are not available in your design plan.
- Step 4** (Optional) Specify the maximum number of threads.
By default, the optimizer tries to set this value to the optimal number of threads based on the available cores.
- Step 5** Click **OK**.
This runs a simulation analysis and creates the optimization report.
-

Compressing Segment Lists

Some routers might not support long segment lists; however, long segment lists might be required to achieve the desired route.

The Compress Segment Lists tool takes LSPs as input and creates new segment lists that adhere to a maximum length. Any segment lists that exceed the maximum length are replaced by new LSPs, LSP paths, and segment lists. A newly created LSP is set as the final segment list hop of the previous LSP; this process repeats until the initial LSP reaches its destination. The tool also creates new LSPs when the route crosses an AS boundary.

- Step 1** Choose **Tools > SR LSP Optimization > Compress Segment Lists**.
- Step 2** Choose which segment lists to compress: all (the default), those selected before opening the dialog box, or those with specific tags.
- Step 3** In the Maximum Segment List Length field, enter a value of at least 3 hops. The default limit is 9 hops.
- Step 4** Check the **Remove Orphaned Segment Lists** check box if you want to remove segment lists that are not used by other LSP paths or SR FRRs as a result of the optimization.

- Step 5** In the **Tag updated LSPs with** field, enter a tag to add to all segment lists that are compressed during optimization. Using tags makes it easier to find the segment lists that need to be compressed in the actual network according to the optimization results. By default, newly compressed segment lists are tagged as *SLOpt* .
- Step 6** Check or uncheck the **New plan for result** check box. If checked, WAE Design generates a new plan file with the results of the optimization and appends the suffix *SLOpt* to the new plan filename. (You can change the filename.) If unchecked, WAE Design changes the current plan file with the updated information.
- Step 7** Click **OK**.
-



CHAPTER 23

Capacity Planning Optimization

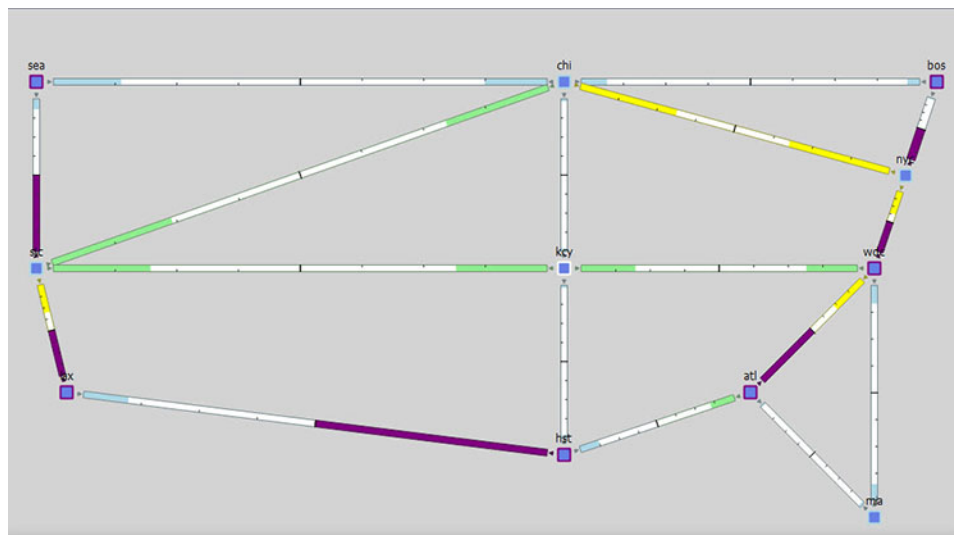
As demand grows on your network you need a way to address the additional traffic and congestion. To alleviate congestion, you can:

- Upgrade existing circuits by adding more capacity to them
- Augment these circuits with associated port circuits
- Add parallel circuits to existing circuits
- Specify new adjacencies between nodes that were not initially connected

To perform these planning optimization investigations, WAE Design includes a capacity planning optimization tool. The goal of a capacity planning optimization is to minimize the addition of any required capacity to be installed on the network. The capacity planning optimizer operates on both Layer 1 and Layer 3 network elements, as well as across failure sets, so you can use a combination of elements in a layered fashion to reach an acceptable solution to meet maximum utilization requirements.

[Figure 135: Design Before Optimization](#), on page 355 shows an example of a network design before optimization, where extra demands were placed on the network. As a result, you can see congested interfaces in purple.

Figure 135: Design Before Optimization



This section contains the following topics:

- [Workflow, on page 356](#)
- [Reports, on page 358](#)
- [Layer 3 Optimization Options, on page 359](#)
- [Layer 1 Optimization Options, on page 361](#)
- [Advanced Optimization Options, on page 362](#)

Workflow

You can set optimization parameters to relieve network congestion and augment capacity. You start by defining a maximum utilization threshold for a specified set of interfaces and then layering additional options that operate on Layer 3, Layer 1, and across failure sets.



Note Some of the optimizer options are mutually exclusive; for example, if you tell the optimizer to create port circuits, it cannot create parallel circuits at the same time. Other options are complementary; for example, you can specify the creation of parallel circuits and new adjacencies together.

Step 1 Choose **Tools > Capacity Planning Optimization**

Layer 3 | Layer 1 | Advanced

Maximum Interface Utilization %

Capacity

Capacity Increment Mb/s

Use Capacity of Existing LAG Members

Upgrade Existing Circuits

Upgrade Circuits

Create Port Circuits (LAGs)

Create Parallel Circuits

Create New Adjacencies

Restrict New Adjacencies between Nodes

Maximum Number of New Adjacencies

Set IGP Metric of New Interfaces to

Shortest IGP Metric minus Decrement

Fixed Metric

Tag Upgraded Circuits with:

Tag New Objects with:

New plan for result:

OK Cancel

Step 2 In the Layer 3 tab, define the maximum utilization threshold that you want to use for all interfaces in the network.

Step 3 Decide on the Layer 3 optimization options to use. (Options are described in [Table 32: Layer 3 Optimization Options , on page 360](#).)

- Specify the capacity parameters to use when upgrading circuits, creating port circuits, or creating parallel circuits.

Note You can specify both the capacity increment and the existing capacity of LAG circuits in tandem.

- Specify circuit upgrade parameters.

You can upgrade all existing circuits or a subset of these circuits. If you want to allow the upgrade of just certain circuits, indicate those circuits to be modified. This is useful if you want to limit your upgrades to a certain geographical location in the network.

To create new port circuits or parallel circuits, choose one of the following options:

- Create new port circuits (LAGs). In this case, the optimizer augments the existing circuits with associated port circuits (LAGs) with additional port circuits. The optimizer converts non-LAG circuits to LAGs.
- Create new parallel circuits. In this case, the optimizer creates new circuits that are parallel to existing circuits.
- Create new adjacencies.

By default, the optimizer does not create new adjacencies. If you specify a set of candidate nodes for it to use, the optimizer proposes new adjacencies. The optimizer restricts a new adjacency between the specified candidate nodes. For example, you might require that only core nodes be directly connected. In this case, specify only core nodes as your candidate nodes. Additionally, you can impose a limit on the maximum number of new adjacencies that the optimizer creates.

To create new adjacencies:

- Choose all the nodes in the table or just a subset of the nodes that you want to optimize.
- Specify the Maximum Number of New Adjacencies threshold.

Note You can apply the IGP metric to new adjacencies, but not to parallel circuits.

- Specify the IGP metric type for new interfaces, which can be either:

Decrement (default): The metric of new interfaces is equal to the shortest path minus the value specified in the metric option.

Fixed: The metric of new interfaces is equal to the value specified in the Fixed Metric field.

Step 4 (Optional) On the Layer 1 tab, check **Create L1 Circuits**. See for the Layer 1 options and [Table 33: Layer 1 Optimization Options , on page 361](#) see [Creating Layer 1 Circuits, on page 362](#) for more information.

Step 5 (Optional) On the Advanced tab, choose options to provide additional capacity planning and describe the cost for the various elements in your design.

Step 6 (Optional) On the Layer 3 tab, override the defaults for how circuits are tagged or for how new optimized plan files are named.

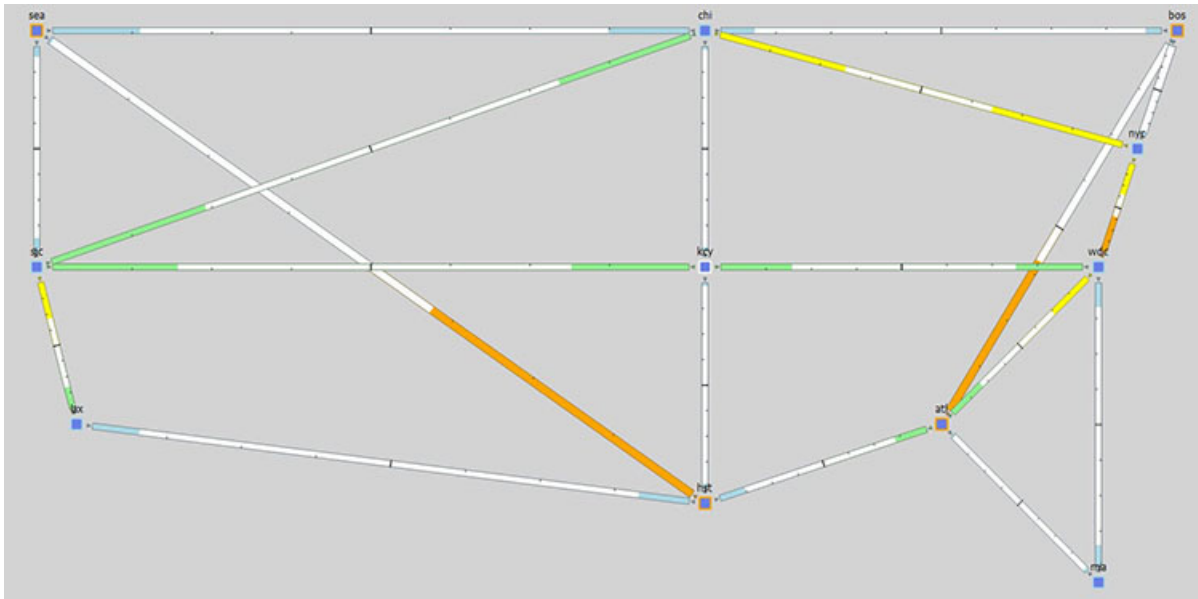
Step 7 Click **OK** to create the capacity planning optimization reports.

WAE Design routes the traffic and looks at the utilization threshold you specified, and any other optimization parameters you specified. After running an optimization, you can look at the summary report to see what the optimizer did to remove congestion. The key metric to look at is the Total Capacity Added (Mb/s).

[Figure 136: Design After Optimization, on page 358](#) shows the design after optimization using the parameters specified in Step 1. In this case, the optimizer proposes to set up two new adjacencies:

- One between Seattle (sea) and Houston (hst)
- One between Atlanta (alt) and Boston (bos)

Figure 136: Design After Optimization



Reports

The Reports window provides a summary report and a capacity upgrades table.

Summary Report Example

After running the optimization, the summary reports provide useful metrics at a glance. Using the previous design as an example with the default utilization threshold of 100%, the decrement interface metric type, and unbounded adjacencies, this report provides details on the:

- Total capacity added as a result of the optimization (in this example, 34832 Mb/s)
- Number of new adjacencies (in this example, 2)
- Number of upgraded circuits (in this example, 4)
- Number of new circuits (in this example, 2)
- Number of new ports (in this example, 36)
- Number of new port circuits (in this example, 28)
- Number of new L1 ports (in this example, 20)
- Number of new L1 circuits (in this example, 10)

- Number of new L1 circuit paths (in this example, 10)

Figure 137: After Optimization Summary Report Example

CAPACITY PLANNING OPTIMIZATION	
Options	
Plan File	:
Output File	:
Number of candidate Nodes	: 22
Interface metric type	: decrement
Interface metric	: 1
Utilization threshold	: 100%
Bandwidth increment	: 2488
Auto Bandwidth increment	: false
Number of Circuits to upgrade	: 50
Maximum adjacencies	: unbounded
Feasibility limit	:
Failure Sets	: none
Number of threads	: 4
New Object tag	: CapacityOpt::New
Upgraded Circuits tag	: CapacityOpt::Upgraded
Parallel Circuits	: false
Create L1 Circuits	: true
Create Explicit Primary L1 Circuit Paths	: false
Create Secondary L1 Circuit Paths	: false
Total Capacity Added (Mb/s)	: 34832
Number of New Adjacencies	: 2
Number of Upgraded Circuits	: 4
Number of New Circuits	: 2
Number of New Ports	: 36
Number of New Port Circuits	: 18
Number of New L1 Ports	: 20
Number of New L1 Circuits	: 10
Number of New L1 Circuit Paths	: 10

Capacity Upgrades Report Example

Click **Capacity Upgrades** for details on how the optimizer upgraded circuits. Using the previous optimized design as an example and using the default utilization threshold of 100%, the decrement interface metric type, and unbounded adjacencies, this report provides details on the six circuit upgrades that it suggests to meet your optimization parameters.

Figure 138: After Optimization Capacity Upgrades Report Example

Action	Name	Node A	Interface A	Node B	Interface B	New Adjacency	NumNewPortCircuits	NumNewL1Circuits
1 Circuit Upgrade	er1.atl-cr2.atl	cr2.atl	to_er1.atl	er1.atl	to_cr2.atl	F	2	
2 Circuit Upgrade	er1.bos-cr2.bos	cr2.bos	to_er1.bos	er1.bos	to_cr2.bos	F	2	
3 Circuit Upgrade	er1.hst-cr2.hst	cr2.hst	to_er1.hst	er1.hst	to_cr2.hst	F	2	
4 Circuit Upgrade	er1.sea-cr1.sea	cr1.sea	to_er1.sea	er1.sea	to_cr1.sea	F	2	
5 Circuit Create		cr2.atl	to_cr2.bos[1]	cr2.bos	to_cr2.atl[1]	T	5	
6 Circuit Create		cr1.sea	to_cr2.hst[1]	cr2.hst	to_cr1.sea[1]	T	5	

In this example, out of the six upgrades listed, there were two new port circuits created for each upgraded circuit.

Layer 3 Optimization Options

You have several Layer 3 options for capacity planning. These parameters tell the optimizer what your preferences are in terms of utilization thresholds, capacity increments, and whether to upgrade existing circuits, create new adjacencies between nodes, create port circuits, or create parallel circuits.

Table 32: Layer 3 Optimization Options

Option	Description
Maximum Interface Utilization	Defines the congestion threshold for all interfaces in the network. By default, the capacity planning optimizer sets the threshold at 100% congestion. However, for future planning purposes, you might want to set the utilization to a lower number, such as 90%.
Capacity Increment	Defines the bandwidth increment value (that is, the allowed capacity increment). You can think of this value as the capacity of the ports. The default is 100000 Mb/s.
Use Capacity of Existing LAG Members	Augments capacity based on the capacity already defined in the existing LAG Members instead of using the Capacity Increment.
Upgrade Circuits	Upgrades all circuits, or just a subset of circuits or those circuits with certain tags to reduce congestion. By default, the optimizer assumes that you want to upgrade all circuits. To indicate that you want to use a subset of these circuits, you can choose them using tags or by choosing a subset of them from the WAE Design tables.
Create Port Circuits (LAGs)	Creates port circuits to reduce congestion. If a non-lag circuit is upgraded, the optimizer creates a port circuit with the initial capacity (existing L1 circuits are reassigned to this port circuit) and new port circuits are created.
Create Parallel Circuits	Creates parallel circuits to reduce congestion. If you tell the optimizer to create parallel circuits, you cannot also tell it to create port circuits and vice versa.
Restrict New Adjacencies between Nodes	By default, the optimizer does not create new adjacencies. If you specify a set of candidate nodes for it to use, the optimizer proposes new adjacencies. The optimizer restricts a new adjacency between the specified candidate nodes. For example, you might require that only core nodes be directly connected. In this case, specify only core nodes as your candidate nodes.
Maximum Number of New Adjacencies	Specifies the maximum number of adjacencies to create between candidate nodes. By default, this is an unbounded number.
Set IGP Metric of New Interfaces to Shortest IGP Metric minus Decrement	Sets the IGP metric of the interfaces corresponding to new adjacencies to be the shortest IGP metric minus the decrement of 1. This is how the IGP metric of such interfaces is set by default. The IGP metric of new interfaces due to parallel circuit upgrades is the same as the metric of the parallel interfaces.
Fixed Metric	Defines a fixed IGP metric to use when creating new interfaces. Enter a positive integer as a value in this field.
Tag Upgraded Circuits with	Defines tags any upgraded circuits. By default, the optimizer tags upgraded circuits with the label <i>CapacityOpt::Upgraded</i> .
Tag New Objects with	Defines tags for any new objects the optimizer creates. By default, the optimizer tags upgraded circuits with the label <i>CapacityOpt::New</i> .
New plan for result	Specifies a name for the new plan file. By default, the optimizer appends the input plan filename with <i>-CapPlanOpt</i> .

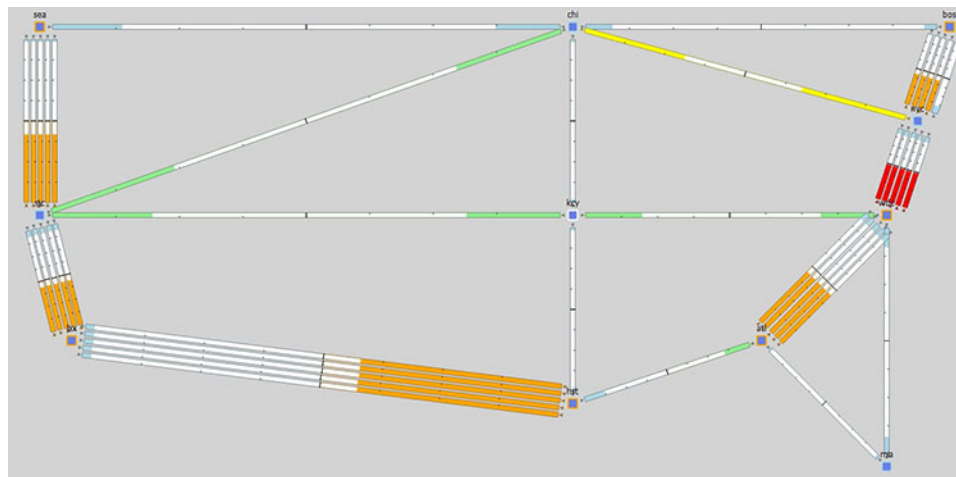
Parallel Circuits Design Example

Figure 139: Adding Parallel Circuits to Increase Capacity, on page 361 shows the first design in this chapter with parallel circuits added to increase capacity. In this case, the optimizer proposes five sets of parallel circuits:

- One between Houston (hst) and Los Angeles (lax)
- One between Los Angeles (lax) and San Jose (sjc)
- One between San Jose (sjc) and Seattle (sea)
- One between Boston (bos) and New York City (nyc)
- One between Atlanta (atl) and Washington, D.C. (wdc)

Because the utilization threshold was set to 100%, utilizations over 90% (shown in red) are acceptable solutions.

Figure 139: Adding Parallel Circuits to Increase Capacity



Layer 1 Optimization Options

The Layer 1 options tell the optimizer to create new L1 circuits.

Table 33: Layer 1 Optimization Options

Option	Description
Match Explicit Routes of Parallel Primary L1 Circuit Path	Assigns L1 circuit path hops to new primary L1 circuit paths to match the explicitly specified routes of existing parallel L1 circuits. By default, paths are dynamically routed.
Create non-standby Secondary L1 Circuit Path	Creates secondary L1 circuit paths to simulate optical restoration capabilities.
Feasibility Limit	Ignores new L3 adjacencies with no possible route in the L1 topology that is lower than the specified feasibility limit. The default is unbounded.

Option	Description
Acceptance Threshold	<p>If the total feasibility metric on the L1 links on the L1 circuit path exceeds the feasibility limit, WAE Design rejects the path. The metric on the L1 link is not entirely fixed; it might contain some noise. You can specify how much noise to take into account when testing the path metric against the metric limit.</p> <p>When WAE Design proposes new L1 circuit paths between two L1 nodes, it's important that the feasibility metric limit is not violated. Therefore, you can specify what acceptance threshold to use when considering new L1 circuit paths. Each L1 circuit path has its own feasibility threshold.</p>

Creating Layer 1 Circuits

If you know exactly where you want to place L1 circuits, you can use explicit routing. To simulate optical restoration capabilities, you can create secondary L1 circuit paths.

-
- Step 1** In the **Capacity Planning Optimization** tool, click the **Layer 1** tab.
- Step 2** Check **Create L1 Circuits**.
- Step 3** (Optional) Specify whether you want to use explicit routing, create secondary L1 circuit paths, or both. See [Table 33: Layer 1 Optimization Options](#), on page 361 for the Layer 1 options.
- Step 4** (Optional) In the Feasibility Limit field, enter a positive number.
- The optimizer takes into account the *feasibility limit* of each Layer 1 circuit path; that is, how far it can travel and what impairments it can suffer.
- Because of the feasibility limit, the optimizer might, for example, recommend adding five 10-GB circuits instead of a single 100-GB circuit.
- Step 5** (Optional) In the Acceptance Threshold field, enter a positive number. Lower thresholds increase the risk that the L1 circuit path is not feasible in the network. The default is 3.
- Step 6** Click **OK**.
-

Advanced Optimization Options

Use the advanced options to:

- Choose whether to optimize for capacity or cost.
- Define what failure scenarios to consider.

-
- Step 1** In the Capacity Planning Optimization tool, click the **Advanced** tab.
- Step 2** Click one of the following radio buttons:
- **Minimize Capacity**—(Default) Minimizes the capacity that is added to your network.

- **Minimize Cost**—Runs the optimization with the objective of reducing the overall cost when adding new capacity. Click **Insert** to manually enter a cost unit for capacity, L3 and L1 ports, and the feasibility limit. Alternately, click **Import** to import cost values from a file.

When you run an optimization, the tool generates a report on the overall cost—including the cost of the added ports—and recommends the most cost-efficient solution.

- Step 3** In the Failure Sets area, choose options that you want the optimizer to consider (circuits, nodes, sites, and so on). Entries appear dimmed if they are not available in your design plan.
- Step 4** (Optional) Specify the maximum number of threads. By default, the optimizer tries to set this value to the optimal number of threads based on the available cores.
- Step 5** Click **OK**.

This action creates the capacity planning optimization report showing the results of the increased capacity. The optimizer takes into account whether it makes sense to upgrade existing circuits or set up new adjacencies, and considers capacity increments, cost options, and feasibility limits.



CHAPTER 24

L1 Circuit Path Optimization

One common use case for L1 circuit path optimization is that disjoint L1 circuit paths can ensure services are highly resilient when network failures occur. For example, this tool lets you route L1 circuit paths to have disjoint primary and secondary paths that are optimized to use the lowest delay possible.

L1 circuit paths are *disjoint* if they do not route over common objects. This is useful, for example, when planning L1 circuit paths, particularly when multiple L1 circuit paths connect the same pair of L1 nodes. Using disjointness on these L1 circuit paths can help improve availability.

If any L1 circuit path is unrouted, no L1 circuit path hops are created for it. If the simulated central frequency ID of an L1 circuit path is different than the user-specified central frequency ID on the path, the initializer sets the central frequency ID value of the L1 circuit path to the simulated central frequency ID. You can choose to retain existing L1 circuit path hops.

If it is not possible to achieve the optimization as defined by the routing selection, path requirements, and constraints, the tool provides the best disjoint paths and optimization possible.

This section contains the following topics:

- [Running the L1 Circuit Path Optimization Tool, on page 365](#)
- [Reports, on page 366](#)

Running the L1 Circuit Path Optimization Tool

Step 1 Choose **Tools > L1 Circuit Path Optimization**.

Step 2 From the L1 Circuit Paths drop-down list, choose which L1 circuit paths to optimize: all, those selected in the table, or those with specific tags. The default is to optimize all selected L1 circuit paths, and by default, all L1 circuit paths are considered selected.

Step 3 From the Minimize Path Metric drop-down list, choose whether to optimize L1 circuit paths with respect to the:

- L1 link metric
- Delay of the L1 link
- Distance of the L1 link
- Hop count

This minimization is for the sum of the metrics along the L1 circuit path.

- Step 4** Choose one of the following disjoint path constraints:
- None—Do not create disjoint paths and do not enforce any disjointness.
 - Create disjoint paths between L1 circuits in disjoint groups—For all L1 circuits that are in disjoint groups, route all L1 circuit paths so that they are disjoint from all other paths belonging L1 circuits in that disjoint group. This disjointness extends beyond primary and secondary paths to include all other path options (for example, tertiary).
 - Create disjoint primary and secondary paths for L1 circuits—For all L1 circuits, whether they are in a disjoint group or not, route all L1 circuit paths so that they are disjoint from all other paths belonging to that L1 circuit. This disjointness extends beyond primary and secondary paths to include all other path options.
- Note** If disjoint path constraints are selected, L1 Diversity Constraints involving the selected L1 Circuits are set to be inactive in the output plan.
- Step 5** If you chose to create disjoint paths, click **Edit** and specify priorities for the type of disjointness. Then, click **OK** in the Edit Disjoint Priorities dialog box.
- Step 6** (Optional) To retain existing L1 circuit path hops and consider them in the optimization, check **Respect existing L1 Circuit Path Hops**. To remove existing L1 circuit path hops and actual L1 circuit path hops, leave the check box unchecked.
- For example, if you check this check box and a loose node hop is specified for an L1 circuit path, the created fully explicit path must contain a strict hop to the node.
- Step 7** In the **Tag Updated L1 Circuit Paths with** field, enter a tag to add to all paths rerouted during optimization. Using tags makes it easier to find the L1 circuit paths that need to be reconfigured in the actual network according to the optimization results. By default, newly optimized L1 circuit paths are tagged as *L1Opt*.
- Step 8** Check or uncheck the **New plan for result** check box. If checked, WAE Design generates a new plan file with the results of the optimization and appends the suffix *L1Opt* to the new plan filename. (You can change the filename.) If unchecked, WAE Design changes the current plan file with the updated information.
- Step 9** Click **OK**.
-

Reports

Upon completion, WAE Design writes a report containing the results of the optimization. (To access this information later, choose **Window > Reports**.)

The Disjoint Groups and Path Disjointness reports list the number of common objects (selected as disjoint path requirements) before and after the optimization, as well as disjointness violations based on these requirements before and after the optimization.

Example

```
LAYER 1 CIRCUIT PATH OPTIMIZER
=====
Options
-----
Plan File                :
Output File              :
Existing L1 circuit hops : remove
```



```
Routing selection                : Current
Disjoint priorities
  (L1 Link, L1 Node, Site, SRLG) : 1,ignore,ignore,ignore
L1 Circuit Paths
-----
Number of L1 Circuit Paths selected : 20
Number of Disjoint Groups           : 2
Disjoint Groups with common
L1 Links                            : 0
L1 Nodes                            : 2
Sites                               : 2
SRLGs                               : 0
(See Path Disjointness Report for details)
```

Example



CHAPTER 25

Changeover

WAE Design lets you model the topology, routings, and utilizations of a currently operational network. It also allows exploration of modifications to that network. For example, interface metrics can be modified or explicit LSP routings changed to obtain a new routing configuration.

The Changeover tool provides a step-by-step sequence of routing configuration changes to move a network safely from an initial configuration to a prespecified final configuration. WAE Design selects the order of these changes so that there is as little as possible congestion in the network during intermediate configurations, and so that this congestion lasts for as few intermediate steps as possible.

Only certain configuration changes are allowed between the initial and final plan:

- Changes to interface metrics
- Changes to LSP configurations
- Changes of circuits and node from the active to the inactive state, and vice versa

An individual step in the changeover sequence consists of one of the following:

- A single metric change on a specific interface
- LSP configuration changes on all LSPs sourced from a specific node
- The activation or inactivation of a specific circuit or node

This section contains the following topics:

- [Running the Changeover Tool, on page 369](#)
- [Reports, on page 370](#)

Running the Changeover Tool

- Step 1** Open both the initial and final network plans in WAE Design, and choose the desired plan to view.
- Step 2** Choose **Tools > Changeover**.
- Step 3** Decide on the changeover options to use. (See [Table 34: Changeover Options](#), on page 370 for field descriptions.)
- Step 4** Click **OK**.
-

What to do next

Table 34: Changeover Options

Field	Description
Initial Plan	Name of the initial plan, selected from open plan files.
Final Plan	Name of the final plan, selected from open plan files.
Traffic Level	Changeover monitors maximum interface utilization levels for all steps in the sequence. The utilizations are calculated using this traffic level.
Acceptable Utilization	The percentage of maximum acceptable utilization of any interface at any step during the changeover sequence. Changeover tries to keep utilizations below this level, although it is not always possible. Often the utilization spikes upward during the last few steps of the sequence, if, for example, new ECMP paths must be put in place. WAE Design tries to keep the number of high-utilization steps to a minimum.
Group Interface Metric Steps	Changeover Tool allows you to group Interface Metric changes together into steps. You can choose from the following options: <ul style="list-style-type: none"> • None - Each Interface Metric change is treated as a single step. • Parallel - Parallel Interfaces are grouped into a single step. • Source Node - Interfaces sourced from the same node are grouped into a single step if there is no impact on maximum utilization.
Group LSP steps by source node	Specify whether to group the LSP steps by source node.

Reports

Changeover creates a report that contains Summary, Steps, and Utilizations sections.

- To access this information later, choose **Window > Reports**.
- To export the steps taken to change over to a new plan file, right-click **Steps** in the Reports window, and choose **Export Tables**.
- To export the utilization values to a table, right-click **Utilizations** in the Reports window, and choose **Export Tables**.

Summary

The Summary section contains a list of the options used to run the changeover. It also contains a summary of the differences between the initial and final plans, the number of steps taken, and the number of steps resulting in intermediate configurations with utilizations in excess of the acceptable utilization.

Steps

The Steps section contains the following columns, some of which you might need to show because they are hidden:

- Step—Steps of the changeover sequence.
- Action—Action taken at the step.
 - MetricChange—An interface metric is changed.
 - CapacityChange—An interface capacity is changed.
 - TunnelsFromRouter—LSPs from a specified router are to be reconfigured.
 - ActivateNode—A node (router) is activated.
 - DeactivateNode—A node (router) is deactivated.
 - ActivateInterface—An interface is activated.
 - DeactivateInterface—An interface is deactivated.
- Node
 - If Action is MetricChange, the node containing the interface of the metric to change.
 - If TunnelsFromRouter, the node at which the LSPs are to be reconfigured.
 - If ActivateNode or DeactivateNode, the relevant node.
 - If ActivateInterface or DeactivateInterface, the node containing the relevant interface.
- RemoteNode—If Action is MetricChange, the remote node containing the interface of the metric to change.
- Circuit—If Action is MetricChange, the circuit for the metric change.
- Interface
 - If Action is MetricChange, the interface name.
 - If TunnelsFromRouter, ActivateNode, or DeactivateNode, this is na (not applicable).
 - If ActivateInterface or DeactivateInterface, the relevant interface name.
- MaxUtil—The maximum interface utilization at this step.
- UtilLocation—The interface on which the maximum utilization occurs.
- ChangeTo—If Action is MetricChange, the original metric. Otherwise, this is set to “na.”
- ChangeFrom—If Action is MetricChange, the metric to be configured at this step. Otherwise, this is set to “na.”

Utilizations

The Utilizations section contains all utilizations for all interfaces in the network, for each step in the changeover sequence.



CHAPTER 26

Patch Files

A patch file is a compact way to represent differences between plan files. These differences or *patches* can be applied to other plan files or deployed to the network.

You can create, apply, view, and send patches by choosing **Tools > Patches**.

A common WAE Design use case is to create a plan file for deployment. One potential workflow for doing this is the following:

1. Open a plan file from a WAE Automation server. (**File > Open from > WAE Automation Server**.)
2. Duplicate the plan file by saving it as a different name.
3. Make LSP modifications as needed in the duplicated plan file.
4. Open both plan files, and create a patch file. (**Tools > Patches > Create**.)
5. Perform a dry run to test the patch before deploying it. (**Tools > Patches > Send, Dry Run** option.)
6. Deploy the patch file to the network. (**Tools > Patches > Send**.)

This section contains the following topics:

- [Opening a Remote Plan File, on page 373](#)
- [Creating Patch Files, on page 375](#)
- [Applying Patch Files, on page 376](#)
- [Viewing Patch Files, on page 376](#)
- [Sending Patch Files, on page 377](#)

Opening a Remote Plan File

From the WAE Design GUI, you can open plan files residing on WAE Archive, WAE Automation Server, WAE Modeling Daemon, and Design Archive to make modifications before deploying LSPs.

Prerequisites:

- You must know the username and password of the server, or its authentication must be disabled.
- The server must be configured for the type of deployment you are using.

Table 35: Opening a Remote Plan File

To...	Do This...
open a file from WAE Archive	<ol style="list-style-type: none"> 1. Choose File > Open from > WAE Archive. 2. Identify the remote server. Enter or choose the hostname or IP address of the server. Specify how to connect to the server by entering or choosing the port number and network. To automatically populate the Network field with the list of available networks, click Refresh List of Networks. 3. If authentication is enabled, enter the same username and password. If authentication is disabled, the username and password fields must be empty. 4. Choose the latest plan or select a plan from the archive. 5. Click OK.
open a file from WAE Automation Server	<ol style="list-style-type: none"> 1. Choose File > Open from > WAE Automation Server. 2. Identify the remote server. Enter or choose the hostname or IP address of the server. Specify how to connect to the server by entering or choosing the port number and network. 3. If authentication is enabled, enter the same username and password. If authentication is disabled, the username and password fields must be empty. 4. Click OK.
open a file from WAE Modeling Daemon	<ol style="list-style-type: none"> 1. Choose File > Open from > WAE Modeling Daemon. 2. Identify the remote server. Enter or choose the hostname or IP address of the server. Specify how to connect to the server by entering or choosing the port number. 3. If authentication is enabled, enter the same username and password. If authentication is disabled, the username and password fields must be empty. 4. Click OK.

To...	Do This...
open a file from Design Archive	<ol style="list-style-type: none"> 1. Choose File > Open from > Design Archive. 2. Identify the remote server. Enter or choose the hostname or IP address of the server. Specify the protocol and the port. 3. Select the Archive. 4. If authentication is enabled, enter the same username and password. If authentication is disabled, the username and password fields must be empty. 5. Enter the UTC Offset. 6. Select Ignore SSL Options to display the SSL errors as warnings. 7. Choose the latest plan or select a plan from the archive or select a template. 8. Click OK.

Creating Patch Files

The patch created identifies how to get from the plan file identified in the From Plan field to the plan file identified in the To Plan field. These patch files only include differences in new, modified, and deleted objects that are supported by the WAE YANG model.

The screenshot shows a dialog box for creating patch files. It is divided into three main sections: 'From Plan', 'To Plan', and 'Patch'.
 - **From Plan:** Contains a radio button for 'Plan File' (selected) with a dropdown menu showing 'Edge-AK', and another radio button for 'Get Plan from WAE' with a 'Location...' button.
 - **To Plan:** Contains a text field for 'Plan File' with the value 'Edge-EP'.
 - **Patch:** Contains a text field for 'Name' with the value 'Edge-AK-Edge-EP.plp', and a text field for 'Location' with the value '|Windows-x86_64' and a 'Browse...' button.
 At the bottom, there are 'OK' and 'Cancel' buttons.

Examples:

- If From Plan Edge-AK contains 33 LSPs and To Plan Edge-EP contains 23 LSPs, the patch file identifies which 10 LSPs to delete from plan Edge-AK to create plan Edge-EP.
- If From Plan Edge-AK contains 10 LSPs and To Plan Edge-EP contains 15 LSPs, the patch file identifies which 5 LSPs to add to plan Edge-AK to create plan Edge-EP.
- If both plan files contain the same LSP, but in From Plan Edge-AK the LSP has a SetupBW property equal to 100, whereas in the To Plan Edge-EP it does not, the patch file identifies the need to modify the SetupBW property of this LSP in plan Edge-AK so that it matches plan Edge-EP.

-
- Step 1** Open the two plan files that you are using for the patch file creation.
- Step 2** From the plan file used as the To Plan, which is the plan file you want to achieve using the patch file, choose **Tools > Patches > Create**.
- Step 3** In the From Plan field, choose one of the following options:
- Plan File—Choose the plan file from which you are determining required changes to achieve the plan file identified in the To Plan field.
 - Get Plan from WAE—Obtain the plan file from WAE with a specified location.
- Step 4** The default patch filename combines the From and To plan filenames. If needed, change the filename. It must have a .plp extension. Unless you want the patch file to be overwritten, the patch filename must be unique to its location.
- Step 5** Enter or browse to the location where you want the resulting patch file to reside.
- Step 6** Click **OK**.
-

Applying Patch Files

You can apply patch files to plan files.

-
- Step 1** Open the plan file to which you want to apply the patch.
- Step 2** Choose **Tools > Patches > Apply**.
- Step 3** Enter or browse to the location where the patch file resides.
- Step 4** Enter a unique name for the plan file you want to create as a result of this applied patch.
- Step 5** Specify whether you want to stop or continue in case of error.
- Step 6** To verify the patch file's contents before applying it, click **View Patch**. (After viewing the patch, click **Close**.)
- Step 7** Click **OK**.
-

Viewing Patch Files

Viewing patch files lets you ensure they meet your needs before applying or deploying them.

Step 1 Choose **Tools > Patches > View**.

Step 2 Browse to and choose the patch file, and then click **Open**.

Step 3 View the patch and then click one of the following:

- **Send Patch**—To deploy the patch as is.
 - **Edit Patch**—To edit the patch before sending it. A dialog box opens, displaying the contents of the patch as XML text. You can save or discard the patch text edits. The tool warns you if you try to save invalid XML syntax. You can right-click in the patch text dialog and choose **Undo** to undo changes. Other standard text editing (cut, copy, paste, and so on) is also supported.
 - **Close**—To close the dialog box without deploying the patch.
-

Sending Patch Files

When sending a patch file, it is applied to the specified network, and related deployment actions are triggered.

Step 1 Choose **Tools > Patches > Send**.

Step 2 Browse to or enter the full path to the patch file.

Step 3 Identify the remote server:

- Enter or choose the hostname or IP address of the server.
- Specify how to connect to the server by entering or choosing the port number and network.

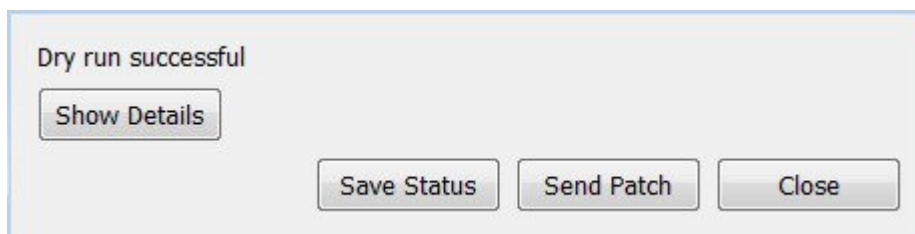
To automatically populate the Network field with the list of available networks, click **Refresh List of Networks**.

- If authentication is enabled, enter the same username and password. If authentication is disabled, the username and password fields must be empty.

Step 4 (Optional) Check **Dry Run**.

Best practice is to perform a dry run to verify the patch file can be successfully applied. If you choose this option, a status popup shows whether the deployment of this patch would be successful, unsuccessful, or only partially successful. To save the status of the dry run, click **Save Status**.

Tip If you check **Dry Run** or set the Test Option to **Test Only**, and the patch dry run succeeds, you can deploy the patch easily. Click **Send Patch** in the status popup; you then return to the Send Patch dialog box with the patch preselected.



Step 5 Specify what to do in case of an error: roll back, stop, or continue.

- Step 6** Click **View Patch** to ensure the patch meet your needs before deploying it. (After viewing the patch, click **Close**.)
- Step 7** Click **OK**.
-

Saving and Clearing Settings

- Step 1** To save all settings for use the next time you open the Send Patch dialog box, check **Save Settings**. This saves all settings whether the fields are empty or not.
- Step 2** To save the authentication password, check **Save Password**.
- Step 3** To clear all current user-specified settings from the fields, click **Clear Settings**. Note that this does not clear the default port value, although any manual entries are cleared.
- Step 4** To delete all user-specific settings so the next time you open the Send Patch dialog box all fields are empty (except for the default port value), follow these steps:
- Click **Clear Settings**.
 - Ensure that **Save Settings** is checked.
 - Click **OK** to exit.
 - At the prompt to continue, click **Yes**.
-



CHAPTER 27

Reports

WAE Design lets you generate reports that compare IP/MPLS topology and configuration information, demand routings, LSP routings, or traffic between two plan files. You can use these reports to:

- Plan for upgrades—Compare plan files before and after topology and configuration changes are made to a current network. For example, you could compare the original plan file to a proposed new plan in which circuits or nodes are added or upgraded.
- Mitigate congestion—Compare plans before and after making configuration changes to mitigate congestion due to a failure or planned maintenance, such as metric changes or LSP additions and reroutings.
- Validate simulations—Compare the simulated traffic under failure in one plan to the measured traffic after a failure has occurred in another plan to validate the accuracy of the simulation.
- Audit plans—Review any changes between two plan files.
- Identify demand and LSP reroutes—Compare plans to determine which demands or LSPs experienced routing changes, for example as the result of an interface metric change or capacity upgrades.

Upon comparing plan files, a report of the results opens automatically. You can access this report later by choosing **Window > Reports**.

For More Information...	See...
Reports on LSP path routes	MPLS Simulation, on page 113
Reports on L1 circuit routes	Layer 1 Simulation, on page 191
Reports on demand groupings and demand grouping costs	Traffic Forecasting, on page 91 Cost Modeling, on page 389
Reports on infrastructure costs	Cost Modeling, on page 389

This section contains the following topics:

- [Plan Comparison Reports, on page 380](#)
- [Traffic Comparison Reports, on page 387](#)
- [Viewing or Exporting Reports, on page 388](#)

Plan Comparison Reports

Comparison reports let you compare objects between two plan files to determine:

- Which objects are present in one plan, but not in the other.
- Which objects are present in both plans and which, if any, have different properties.

Plan 1: us_netwk

Plan 2: us_wan_L1

Report

IP/MPLS topology and configuration

Demand routings

LSP routings

Complete plan comparison

For each object

Only show properties with differences

Show all properties compared

381186

You can run four types of plan comparison reports:

- IP/MPLS topology and configuration—Objects describing the topology (such as nodes, circuits, and interfaces), configured objects (such as LSPs and LSP paths), and related properties (such as IGP metrics and capacity).
- Demand routings—Demand paths and the properties that indicate routing changes, such as Path Length and Max Latency. For a complete list of compared properties, see [Table 36: Plans Comparison Report: IP/MPLS Topology and Configuration](#), on page 381.
- LSP routings—Properties of LSPs that indicate routing changes, such as TE Path Metric and Active Path Sim. For a complete list of compared properties, see [Table 37: Plans Comparison Report: Demand Routings](#), on page 384.
- Complete plan comparison—All tables in the table schema are compared, except for internal simulation caching tables. For a complete list of these tables, refer to the \$CARIDEN_HOME/docs/table_schema file.

Report Columns

Each WAE Design table has key columns that uniquely identify objects. For example, the key columns in the Interfaces table are the Node and Interface columns, which represent the name of the node containing the interface and the name of the interface itself. In a Plan Comparison report, an object in one plan is identified with an object in another plan only if the key columns of the two objects match. That is, key columns determine if an object is present in both plans or in only one of the plans.

The Plan Comparison report displays three other types of columns. [Table 36: Plans Comparison Report: IP/MPLS Topology and Configuration](#), on page 381, [Table 37: Plans Comparison Report: Demand Routings](#), on page 384, and [Table 38: Plans Comparison Report: LSP Routings](#), on page 385 list the properties reported upon and their associated column type.

- Information only (Info)—There are no comparisons made. Information is reported for the plan file from which you are running the report (Plan 1).
- Differences (Diff)—Specific properties are compared for each object that has matching key columns across the two plan files. Objects are then identified as belonging in either Plan 1 only (file from which you are running the report), in Plan 2 only (plan file to which you are comparing), or in both Plan 1 and Plan 2. If the object exists in both plan files, but has different properties, a Diff column shows T (true) if there are any differences or F (false) if there are none.

Differences are based on the current state of the open plan files. For example, if a circuit has failed, the demand routings change to route around the failure.

- Summary Differences (Summ Diff)—These are not columns within WAE Design tables. They are T/F (true or false) values, depending on some differences in the table objects that are not represented in the table columns. For example, if a common LSP path is configured differently between plans, this difference is represented in a summary difference column in the LSP section. If demand routings differ, this is represented in a summary difference column in the Demands section.



Note For easy reference, the following tables list columns in the order of type first: Key, Info, Diff, and Summ Diff. Properties within each column type are alphabetical.

Table 36: Plans Comparison Report: IP/MPLS Topology and Configuration

Compared Table	Compared Columns	Comparison Column Type	Summary Difference Description
Interfaces	Name	Key	
	Node	Key	
	Remote Node	Info	
	Affinities	Diff	
	Area	Diff	
	Capacity	Diff	
	Circuit	Diff	
	Description	Diff	
	FRR Protect	Diff	
	IGP Metric	Diff	
	IP Address	Diff	

Compared Table	Compared Columns	Comparison Column Type	Summary Difference Description
	PC Min BW	Diff	
	PC Min Links	Diff	
	Resv BW	Diff	
	TE Metric	Diff	
Circuits	InterfaceA	Key	
	InterfaceB	Key	
	NodeA	Key	
	NodeB	Key	
	Name	Info	
	Active	Diff	
	Capacity	Diff	
	Delay	Diff	
Nodes	Name	Key	
	Active	Diff	
	AS	Diff	
	BGP ID	Diff	
	ECMP Max	Diff	
	IP Address	Diff	
	IP Manage	Diff	
	Model	Diff	
	OS	Diff	
	Vendor	Diff	
LSPs	Node	Key	
	Source	Key	
	Active	Diff	
	Destination	Diff	
	Exclude	Diff	
	FRR Link Protect	Diff	

Compared Table	Compared Columns	Comparison Column Type	Summary Difference Description
	Hold Pri	Diff	
	Hop Limit	Diff	
	Include	Diff	
	Include Any	Diff	
	Load Share	Diff	
	Metric	Diff	
	Metric Type	Diff	
	Setup BW	Diff	
	Setup Pri	Diff	
	Unresolved Destination	Diff	
	LSPs Path Diff	Summ Diff	T (true) if there are any differences in the LSP Paths table for this LSP. If two LSPs have differences in their named path hops or LSP paths, these differences are propagated to this column.
LSP Paths	Node	Key	
	Path Option	Key	
	Source	Key	
	Exclude	Diff	
	Hold Pri	Diff	
	Hop Limit	Diff	
	Include	Diff	
	Include Any	Diff	
	Path Name	Diff	
	Setup BW	Diff	
	Setup Pri	Diff	
	Standby	Diff	
	Named Path Diff	Summ Diff	T (true) if there are any differences in the Named Paths table for this LSP path.
Named Paths	Active	Key	

Compared Table	Compared Columns	Comparison Column Type	Summary Difference Description
	Name	Key	
	Source	Key	
	Named Path Hops Diff	Summ Diff	T (true) if there are any differences in the Named Path Hops table for this named LSP path.
Named Path Hops	Name	Key	
	Source	Key	
	Step	Key	
	Interface	Diff	
	IP Address	Diff	
	Node	Diff	
	Type	Diff	
	Unresolved Hop	Diff	

Table 37: Plans Comparison Report: Demand Routings

Compared Table	Compared Columns	Comparison Column Type	Summary Difference Description
Demands	Destination	Key	
	Name	Key	
	Service Class	Key	
	Source	Key	
	Destination Site	Info	
	Source Site	Info	
	Active	Diff	
	ECMP Min %	Diff	
	Max Latency	Diff	
	Path Metric	Diff	
	Path Diff	Summ Diff	T (true) if there are any differences in routing of the demand.

Table 38: Plans Comparison Report: LSP Routings

Compared Table	Compared Columns	Comparison Column Type	Summary Difference Description
LSPs	Name	Key	
	Source	Key	
	Active Path Sim	Diff	
	Destination	Diff	
	Routed	Diff	
	Shortest TE Path	Diff	
	TE Path Metric	Diff	
	Actual Path Diff	Summ Diff	T (true) if there are any differences in routing of the actual LSP path.
	Simulated Path Diff	Summ Diff	T (true) if there are any differences in routing of the LSP.

Report Sections

The Summary section shows the number of objects in each plan file, the number of objects in both plan files, and the number of objects with property differences ([Figure 140: Example Plan Comparison Report Summary, on page 386](#)).

Example: Nodes A, B, and C are in Plan 1, and B, C, and D are in Plan 2. Nodes B and C have matching key columns. The B nodes have identical properties, but the C nodes have different IP addresses. Therefore, in the Summary section, the In Both Plans column shows 2, and the Different Properties column shows 1.

An individual section is generated for each object that appears in only one plan file and if there are differences between the properties, a section is generated showing these differences. Thus, there is the potential of generating one to three sections per compared table: one for objects appearing only in Plan 1, one for objects appearing only in Plan 2, and one listing objects existing in both plan files, along with their differences noted with a T (true) or F (false).

Figure 140: Example Plan Comparison Report Summary

Number of	Plan 1	Plan 2	Plan 1 Only	Plan 2 Only	In Both Plans	In Both Plans, Different Properties
Interfaces	100	100	0	0	100	14
LSPs	1056	1056	0	0	1056	1056
Demands	89	95	0	6	89	0
Demand Traffic	89	95	0	6	89	89
Interface Traffic	100	99	1	0	99	99
Plot Layout Elements	57	46	11	0	46	0

Other than Summary, the report sections and columns (properties) that appear depend on the options selected when generating the report.

- If you select to show only properties with differences, only those columns with differences in properties appear in the report. If only one value is different, all objects are listed (not just the one with the different property).
- If you select to show all properties compared, all compared properties appear in the report whether there are differences between the two plan files or not.

Creating Plan File Comparison Reports

-
- Step 1** Open the two plan files you want to compare.
- Step 2** In either of these two plan files, choose **Tools > Reports > Compare Plans**.
- Step 3** Choose the plan file to which you are comparing the open plan.
- Step 4** Choose the type of comparisons you want to generate:
- IP/MPLS topology and configuration
 - Demand routings
 - LSP routings
 - Complete plan files
- Step 5** Choose whether to show only properties with differences or whether to show all properties compared.
- Step 6** Click **OK**.
-

Traffic Comparison Reports

Traffic Comparison reports compare traffic values within a single plan or between two plan files. You can compare traffic on one object at a time (nodes, interfaces, circuits, demands, LSPs, multicast flows, and flows). Each object has a different set of comparisons from which to choose. For example, interface traffic and capacity can be compared, and LSP traffic and setup bandwidth can be compared. Another example would be to compare the simulated interface traffic in a plan file before a failure to the measured interface traffic of a plan file in which a failure occurred. Such a comparison lets you determine which interfaces show the greatest differences.

For each plan file compared, the current traffic level and either the selected service class or queue are used. They can be different in each plan.

The Summary section shows a high-level summary of what is being compared. Additionally, a section of the difference on the selected object and traffic columns is generated and named accordingly.

Plan 1	us_wan	
Plan 2	us_wan_L1	
Compare	Interfaces	
	Plan 1	Plan 2
Traffic Type	TraffSim	TraffSim
Traffic Level	Default	Default
QoS	Undifferentiated	Undifferentiated
Total Traffic	108830 Mbps	58286 Mbps
Plan 1 has 50544.1 Mbps more traffic in total than Plan 2		

381188

Creating Traffic Comparison Reports

Step 1 If comparing traffic from the open plan file to another, open the other plan file.



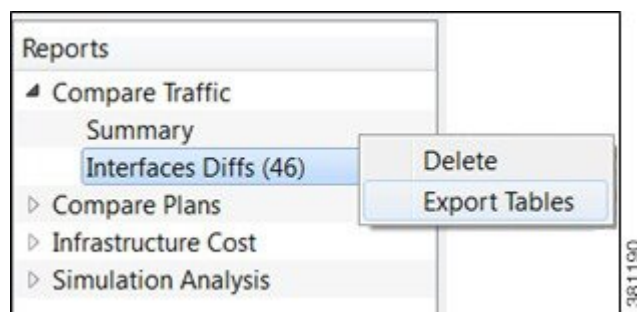
- Step 2** In each plan file, choose the travel levels and choose either the service class or queue traffic to compare.
- Step 3** Choose **Tools > Reports > Compare Traffic**. This plan file is identified as Plan 1 in the Compare Traffic dialog box that opens.
- Step 4** Choose the object type on which to compare traffic (interfaces, circuits, nodes, and so on).
- Step 5** Choose the traffic column in the current plan that you want to compare (Plan 1).
- Step 6** The default Plan 2 is the current plan file. To change this, choose a different plan file for Plan 2.
- Step 7** Choose the traffic column being compared for Plan 2.
- Step 8** Click **OK**.

Viewing or Exporting Reports

Upon generating a report, it opens automatically. From there you can select the various sections of the report from the left navigation pane.

To view a report after having closed it, choose **Window > Reports**. All reports generated on that plan file appear and are selectable from one report window.

If you want to save these reports for later use, you must save the plan file. However, running the report again overwrites the report. To preserve the information, you can export the data by report section. Right-click any of the sections in the left navigation pane and choose **Export Tables**.





CHAPTER 28

Cost Modeling

WAE Design provides basic cost modeling of network plans. Costs are specified for nodes and circuits, and these can be used to approximate infrastructure costs for the network or parts of the network. This allows for quick assessments of the costs of various upgrade alternatives, as well as for greenfield build-outs.

The specified circuit costs can also be used to estimate demand and demand grouping costs. This is the cost of routing traffic between particular locations or sets of locations in the network, taking into account the amount and cost of the circuit infrastructure used. This can be used to assess the cost of a customer, a particular service, or any other defined traffic source on the network.

This section contains the following topics:

- [Infrastructure Costs, on page 389](#)
- [Demand and Demand Grouping Costs, on page 396](#)

Infrastructure Costs

An Infrastructure Cost report can help you approximate the total monetary cost of a network design based on estimated costs assigned to nodes and circuits. These costs are calculated from initial costs and per-period costs, where per-period costs are amortized based on the specified number of periods and interest rates. The report gives an infrastructure cost based on sums of all node and circuit costs, as follows:

- Total initial cost—Sum of all initial costs for each object type.
- Total cost per period—Sum of all per-period costs for each object type before the amortization is applied.
- Total cost—Total initial cost + total cost per period after being amortized to a net present value.
- Total infrastructure cost—Sum of the total node and total circuit costs.

	Nodes	Circuits
Number	41	69
Total initial cost	211,603	73,148
Total cost per period	4,779	13,722
Number of periods	2	2
Interest rate (%)	3.75	3.75
Total cost	220,649	99,122
Total Infrastructure Cost 319,771		

380851

Cost Basis

Costs are based on any combination of Cost Initial and Cost Per Period properties that are assigned to nodes and circuits. A typical use case is to set nodes with an initial cost to identify the initial capital investment and to set circuit costs per period to analyze the operational cost of leasing circuits over time.

- Cost Initial—Static, one-time cost to which amortization and interest rates are not applied.
- Cost Per Period—Ongoing costs that are amortized based on the number of periods identified and on the interest rate specified when generating the Infrastructure Cost report.

You can set these properties manually. For circuits, you can also use the [Cost Circuit Initializer](#), on page 391 to set the costs automatically based on circuit distance and capacity.

The total cost for each node and each circuit appears in its respective Cost Initial or Cost Per Period column in the Nodes table and Circuits table. The sum total of the node and circuit costs appears in the output of the Infrastructure Cost report and is used in its calculations.

An easy way to see totals—as well as average, minimum, and maximum costs—without running the report is to right-click the Cost Initial or Cost Per Period column heading.

Cost Per Period	
1244	
5000	
1244	
5000	
1244	
5000	
1244	
1244	

Hide this Column
 Show/Hide Columns
 Adjust Width to Fit
 Save Columns settings
 76148 (Sum)
 1522.96 (Avg)
 0 (Min)
 5000 (Max)

380840

Setting Cost Properties

You can set these properties manually, as follows.

- Step 1** Choose one or more nodes or one or more circuits. Right-click one of these objects and choose **Properties**, or double-click one of these objects.
- Step 2** Click the appropriate tab:
- Nodes—Click the **Inventory** tab.
 - Circuits—Click the **Advanced** tab.
- Step 3** Enter values in the **Cost (initial)** field, **Cost (per period)** field, or both, and click **OK**.

Cost Circuit Initializer

The Cost Circuit initializer offers an automated way of setting circuit costs. For example, it lets you set realistic per-period circuit costs based on distance, capacity, or both.

The total cost for each circuit is the sum of the fixed cost plus the variable cost, where the variable cost is the result of multiplying the per-unit cost times the units selected (distance or capacity).

To access this initializer, choose **Initializers > Circuit Cost**. Options are as follows:

- **Set cost**—Set initial or per-period cost. The options that follow are applied to the choice selected here. Because initial and per-period costs are different values that do not overwrite each other, you can set one and then run the initializer again to set the other.
- **Fixed cost**—A static cost that is most often used when setting initial costs.
- **Variable cost**—A cost based on a circuit’s distance, capacity, or both. Whichever unit is selected (one or both), the value is multiplied by the “Per-unit cost” value.
 - **Per-unit cost**—Per-unit monetary cost.

- Unit distance—Distance Sim value of the circuit.

Example: If only “Unit Distance” is selected, if the Distance Sim is 700, and if the “Per-unit cost” is 0.50, the variable value for this circuit is $700 \times 0.50 = 350.00$.

- Unit capacity—Capacity Sim value of the circuit.

Example: If only “Unit Capacity” is selected, if the Capacity Sim is 400, and if the “Per-unit cost” is 0.25, the variable value for this circuit is $400 \times 0.25 = 100.00$.

Example: This example sets a per-period cost for a circuit that has a fixed cost of 250, a per-unit cost of 2, circuit distance of 850, and a capacity of 1000. Including all of these options in the calculation, the cost of this circuit is:

$$\text{Fixed} + (\text{Per-unit Cost} \times \text{Distance Sim} \times \text{Capacity Sim})$$

$$250 + (2 \times 850 \times 1000) = 250 + 1,700,250 = 1,700,250$$

Set costs on (1/69) selected circuits

Set initial cost
 Set per period cost

Fixed cost:

Cost varies by:

Unit distance
 Unit capacity

Per-unit cost:

Capacity Sim	Distance Sim	Cost Per Period
1000.00	850.00	1700250.00

360042

Infrastructure Cost Report

To create an infrastructure cost report, follow these steps.

The screenshot shows a dialog box with two main sections: 'Nodes' and 'Circuits'. Each section has a dropdown menu set to 'All' and an 'Amortization' sub-section. Under 'Amortization', there are two input fields: 'Number of periods' set to '2' and 'Interest rate (%)' set to '3.75'. A small vertical text '380850' is visible on the right side of the dialog box.

-
- Step 1** For any node or circuits you want included in the report, set their initial or per-period costs. For more information, see [Setting Cost Properties, on page 390](#).
- Step 2** If creating a report for selected nodes or circuits, select them before opening the report dialog box.
- Step 3** Choose **Tools > Reports > Cost Analysis**.
- Step 4** Nodes:
- Select whether to include all nodes, selected nodes, or specifically tagged nodes.
 - If you are calculating per-period costs, set the number of periods and the interest rate.
- Using tags is a convenient way to determine costs for a specific region of the network or for specific customers.
- Step 5** Circuits:
- Select whether to include all circuits, selected circuits, or specifically tagged circuits.
 - If you are calculating per-period costs, set the number of periods and the interest rate.
- Step 6** Click **OK**.
-

Example Infrastructure Cost Workflow

This example shows the generation of an Infrastructure Cost report based on a specific set of nodes and circuits, both tagged as the East Region. It sets initial costs on the nodes and per-period costs on the circuits based on their distance. The report is for 4 periods, and the interest rate is 5%.

-
- Step 1** Set initial node costs in the node Properties dialog box.
- Filter to the nodes containing the East Region tag.
 - In this example, there are 20 nodes with this tag. Set the 10 core nodes to an initial cost of 2000 each and the 10 edge nodes to an initial cost of 1000 each.

The Nodes table lists these costs in the Cost Initial column. Right-click this column heading to see the total Cost Initial for all of these nodes (here, 30,000). Because we are only interested in initial node costs, this is the same value that the Infrastructure Cost report shows for total initial node cost.

Node Properties

Cost (initial): 2000
Cost (per period): 0

Cost (initial): 1000
Cost (per period): 0

380848

Step 2 Run the Latency and Distance initializer to set Distance Sim values on the circuits (**Initializers > Latency and Distance**).

Set costs on (32/50) selected circuits

Set initial cost

Set per period cost

Fixed cost: 0.00

Cost varies by:

Unit distance

Unit capacity

Per-unit cost: 2.00

380847

Step 3 Set per-period circuit costs using the Circuit Cost initializer on circuits tagged with East Region.

- Filter to the circuits containing the East Region tag, and select these circuits.
- Run the Circuit Cost initializer (**Initializers > Circuit Cost**). For this example, there are no fixed costs, a per-unit cost of 2, and the initializer is only using Distance Sim.

The Circuits table lists these costs in the Cost Per Period column. Right-click this column heading to see the total Cost Per Period for all of these circuits. Here, it is 35,063.46, and this is the number that is amortized in the Infrastructure Cost report based on 4 periods with a 5% interest rate.

Nodes: Tag: East Region (20/33) ▼

Amortization

Number of periods: 1

Interest rate (%): 0

Circuits: Tag: East Region (32/50) ▼

Amortization

Number of periods: 4

Interest rate (%): 5

380846

Step 4

Now that costs are set up for the region, generate the Infrastructure Cost report.

- Choose **Tools > Reports > Cost Analysis**.
- Select the East Region nodes and circuits.
- Because there are no per-period costs for nodes, no further node options are set.

Set the circuits “Number of periods” to 4, and the “Interest rate (%)” to 5.

What to do next

The resulting report performs these calculations:

- Sums the total initial costs for all nodes tagged with East Region to determine the total node cost.
- Sums the per-period costs for all circuits tagged with East Region to determine the total per-period circuit cost.
- Amortizes the per-period circuit costs across 4 periods using a 5% interest rate to determine the total circuit cost.
- Adds the total node cost to the total circuit cost to determine the total infrastructure cost.

	Nodes	Circuits
Number	20	32
Total initial cost	30,000	0
Total cost per period	0	35,063
Number of periods	na	4
Interest rate (%)	na	5
Total cost	30,000	124,333
Total Infrastructure Cost 154,333		

380845

Demand and Demand Grouping Costs

Two estimates of the monetary costs of demands are automatically derived when you set per-period costs for circuits, as follows. These appear as the following column headings in the Demands table:

- **Cost by Capacity**—Total of the Cost Per Period for the circuits using the demand as routed, scaled by proportion of demand traffic to the total interface capacity.

It is divided by two to share the circuit cost between the costs of traffic in two directions.

(Sum Cost Per Period of All Circuits Using the Demand) x (Demand Utilization on the Interface / 2)

Where demand utilization on the interface = (Interface Utilization / Capacity)

- **Cost by Util**—Total of the Cost Per Period for the circuits using the demand as routed, scaled by proportion of demand traffic to the total traffic through the circuit.

(Sum Cost Per Period of All Circuits Using the Demand) x (Demand Utilization on the Interface) / (Sum of the Utilizations on the Two Interfaces)

Where demand utilization on the interface = (Interface Utilization / Capacity)

If there is only one demand through the circuit, the full cost of that circuit is allocated to that demand.

Demand grouping costs are calculated when you run the Demand Grouping report (**Tools > Reports > Demand Groupings**). The Cost by Capacity and Cost by Util are the sum total of the costs of the demands within the demand grouping.

Example

This example shows four nodes connected by three circuits.

- Each circuit has 10,000 Mbps capacity.
- Each circuit has a Cost Per Period of 100.
- There are two demands:

dnv-mia has 2000 Mbps of traffic.

mia-dnv has 5000 Mbps of traffic.

- These two demands are in one demand grouping named Acme_region.



Circuits		Capacity Sim	Capacity	Cost Per Period
Node A	Node B			
dnv	chi	10000.00	10000.00	100.00
chi	nyc	10000.00	10000.00	100.00
nyc	mia	10000.00	10000.00	100.00

Demands				
Source	Destinatio	Traffic	Cost by Capacity	Cost by Util
mia	dnv	5000.00	75.00	214.29
dnv	mia	2000.00	30.00	85.71

380843

Reports	
Demand Groupings	
Demand Groupings (1)	
Dema	

Name	Traffic	Number of Demands	Cost by Capacity	Cost by Util
Acme_Region	7000....	2	105.00	300.00

380844

The Cost by Capacity calculation is as follows:

- dnv-mia demand: $(100 + 100 + 100) \times ((2000/10,000)/2) = 30$
- mia-dnv demand: $(100 + 100 + 100) \times ((5000/10,000)/2) = 75$
- Acme_region demand grouping: $30 + 75 = 105$

The Cost by Util calculation is as follows:

- dnv-mia demand: $(100 + 100 + 100) \times 2000/(2000 + 5000) = 85.71$
- mia-dnv demand: $(100 + 100 + 100) \times 5000/(2000 + 5000) = 214.29$
- Acme_region demand grouping: $85.71 + 214.29 = 300$



CHAPTER 29

Plot Legend for Design Layouts

This legend is for Design layouts used in the WAE Design GUI.



Note All traffic is displayed in Mbps. In the network plot, the traffic utilization colors represent outbound traffic. L3 means Layer 3, and L1 means Layer 1.

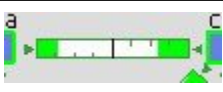
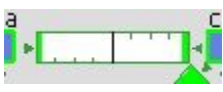
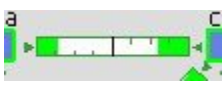


Legend	Description
Interfaces and Circuits	Interfaces and Circuits, on page 400
	Circuit Colors, on page 400
	Circuit Active, Failed, and Operational States, on page 402
Nodes	L3 Nodes, on page 404
Sites	Sites, on page 405
Demands and Routings	Demands, on page 407
	Demand Segment Lists, on page 407
External Endpoint Members	External Endpoint Members, on page 408
LSPs	LSP Paths, on page 409
	LSP Named Paths, on page 410
	LSP Actual Paths, on page 410
	Segment-Routed LSP Paths, on page 411
L1 Objects	L1 Links, on page 412
	L1 Circuit Paths, on page 413
	L1 Circuit Path Hops, on page 414
	L1 Nodes, L1 Ports, and L1 Waypoints, on page 414

This section contains the following topics:

- [Interfaces and Circuits](#), on page 400
- [L3 Nodes](#), on page 404
- [Sites](#), on page 405
- [Demands](#), on page 407
- [External Endpoint Members](#), on page 408
- [LSPs](#), on page 409
- [L1 Objects](#), on page 412

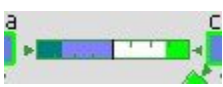
Interfaces and Circuits




Table 39: Interface and Circuit Visualization






Graphic	Description
	Circuits are rectangles divided in half by a vertical black line. Each half shows an interface. Tick marks show 25%, 50%, 75% utilization levels.
	Width of circuit shows capacity. Width increases a fixed amount for every 4x increase in capacity.
	Full-length interface style, which is the default.
	Shortened interface style, showing the same information as a full-length view, just in shorter representations that are repeated at each end of the circuit. In this example, all chi interfaces are shortened at both ends of the circuit.
	Collapsed interface view, showing the presence of interfaces as up/down arrows inside the site. The arrow color shows the highest utilization of any of the collapsed interfaces (in this example, blue for sea and yellow for chi). In this example, all chi interfaces are collapsed at both ends of the circuit.

Circuit Colors

Table 40: Interface and Circuit Colors


Graphic	Description
The traffic utilization colors are customizable. For information, see the <i>Cisco WAE Network Visualization Guide</i> .	
	Blue indicates a selected interface or circuit.






Graphic	Description
	<p>Outbound traffic utilization is depicted by colors. Default colors are light blue, light green, yellow, orange, red, and purple, and left to right, each color represents increased utilization.</p> <ul style="list-style-type: none"> • Measured Traffic and Simulated Traffic views—The color shows the percent of traffic utilized for the available interface capacity. • Worst-Case Traffic view—The color represents the worst-case utilization on each interface. This is the highest utilization that a particular interface experiences over all the failure scenarios considered when running a simulation analysis. • Failure Impact view—The color represents the failure impact of a circuit. This is defined as the highest utilization experienced by any interface if that circuit were to fail. • LSP Reservation view—The color represents the total setup bandwidth that is reserved by LSPs through the interface.
	<p>If parallel circuits are grouped for visual display, each side of the circuit shows its constituent interfaces. Here, there are two circuits (two interfaces on each side) grouped by site. If the circuits are between sites, the utilization includes the utilization of nested interfaces, if applicable.</p> <ul style="list-style-type: none"> • The color fill on each side shows the average utilization of the grouped interfaces, which is weighted against the sum of the grouped capacities. For example, the average utilization for the interfaces from chi to mia is between 30% and 50% and is shown in green. • Each interface within a grouped circuit has a thin colored border showing the utilization of that interface. For example, one interface going from mia to chi is bordered by yellow and another by blue, each showing the traffic utilization for the bordered interface.
	<p>White indicates 0% utilization.</p>

Graphic	Description
	<p>Interfaces are light gray for the following reasons:</p> <ul style="list-style-type: none"> Measured Traffic view <ul style="list-style-type: none"> The interface has no traffic measurement. This amount of interface capacity is not available due to a QoS bound. Simulated Traffic view—This amount of interface capacity is not available due to a QoS bound. Worst-Case Traffic view—No simulation analysis was performed. Failure Impact view <ul style="list-style-type: none"> No simulation analysis was performed. Failing the circuit would not increase the traffic utilization on any other circuit. Circuit is operationally down.
	QoS policy violations are indicated by red and white bars. This is displayed when utilization exceeds the QoS bound, which is set through policy or interface queue properties.
	Shortest IGP path between two selected nodes or sites.
	Shortest latency path between two selected nodes or sites.
	Shortest TE path between two selected nodes or sites.

Circuit Active, Failed, and Operational States












Table 41: Circuit Active, Failed, and Operational States

Graphic	Description
	Failed circuit.

Graphic	Description
	<p>The circuit is not operational because of one of the following reasons:</p> <ul style="list-style-type: none"> • A node connected to the circuit failed. • A site failed that contains a node connected to the circuit. • The circuit is a LAG (port channel) that does not meet its operational requirements (minimum number of available ports or minimum capacity, whichever is surpassed first) due to failure of some of its ports or port circuits. • Mapped L1 circuit cannot be routed because it traverses a failed L1 port, L1 node, or L1 link. • An SRLG failed that contains the circuit or contains one of the above objects that are required for circuit operation.
	<p>The circuit is a LAG, and one or more ports or port circuits in the LAG failed. The red strikeout line is the percent of unavailable capacity due to the failure. In the Worst-Case Traffic view, the red strikeout line is the percent of unavailable capacity under the worst-case failure.</p> <p>If you right-click the failed port or port circuit and choose Recover, the red strikeout line disappears. If you right-click again and choose Fail, the red strikeout line returns.</p>
	<p>The circuit is inactive.</p>
	<p>The circuit is not operational because of one of the following reasons:</p> <ul style="list-style-type: none"> • A node connected to the circuit is inactive. • An inactive site contains a node connected to the circuit. • The circuit is a LAG that does not meet its operational requirements (minimum number of available ports or minimum capacity, whichever is surpassed first) due to some of its ports or port circuits being inactive. • A mapped L1 circuit cannot be routed because it traverses an inactive L1 port, L1 node, or L1 link. • An SRLG is inactive, and the SRLG contains the circuit or contains one of the above objects that is required for circuit operation.
	<p>The circuit is a LAG and one or more ports or port circuits in the LAG is inactive. The gray strikeout line is the percent of unavailable capacity due to the inactive ports or port circuits.</p>

L3 Nodes






Table 42: Layer 3 Node Visualization









Graphic	Description
	A node is shown as a blue rectangle with a border. Solid blue indicates the node is selected.
	A green border surrounds the node when: <ul style="list-style-type: none"> • Simulated Traffic View—The node is the source or destination of simulated traffic that is greater than 0. • Measured Traffic View—The node is the source or destination of measured traffic that is greater than 0. • LSP Reservation View—The node is the source or destination of LSPs with nonzero bandwidth reservations.
	A white border surrounds the node when: <ul style="list-style-type: none"> • Simulated Traffic View—The node is not the source or destination of simulated traffic. • Measured Traffic View—The node has ingress and egress traffic of 0 (zero). • LSP Reservation View—The node is the source or destination of LSPs with 0 (zero) bandwidth reservations.
	A gray border surrounds the node when: <ul style="list-style-type: none"> • Measured Traffic View—The node has no ingress or egress traffic. • Worst-Case Traffic View—Nodes always have gray borders. • Failure Impact View—The node is not included in the failures considered, or failing the node would not increase the traffic utilization on any circuit.
	In the Failure Impact view, the node border color represents the failure impact of a node. This is defined as the highest utilization experienced by any circuit if that node were to fail.
	Up/down triangles show that some interfaces from the node have been collapsed.
	Failed node.
	Node is not operational because it is contained in a failed site or in a failed SRLG.
	Node is not active.
	Node is not operational because it is contained in a site or an SRLG that is not active.
	One or more selected VPN nodes exist on the node.

Sites

Site utilization colors and symbols are based on intrasite objects. Note that intrasite objects include child sites and the objects contained within them.


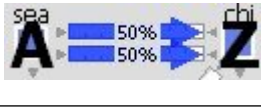
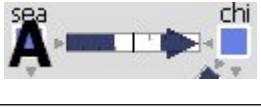



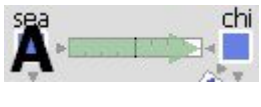

Table 43: Site Visualization

Graphic	Description
	A site is shown as a blue square with a border. Solid blue indicates the site is selected.
	<ul style="list-style-type: none"> • Layer 3 view <ul style="list-style-type: none"> • Measured Traffic and Simulated Traffic views—The border color is the maximum traffic utilization color of all intrasite interfaces within the site. • Worst-Case Traffic view—The border color is the highest utilization that the interfaces within it experience over all the failure sets selected for Simulation Analysis. • Failure Impact view—The border color is the maximum utilization level that would be caused elsewhere in the network should an L3 node or L3 circuit contained within it fail. • Layer 1 Failure Impact view—The border color is the maximum utilization level that would be caused elsewhere in the network should an L1 node or L1 link contained within it fail. • Multi-Layer Failure Impact View—The border color is the maximum utilization level that would be caused elsewhere in the network should any L1 or L3 intrasite object fail.
	<ul style="list-style-type: none"> • Layer 3 view—A white border surrounds the site if all intrasite interfaces have zero utilization. • Layer 1 view—The sites always have white borders except in Failure Impact view.
	<p>A gray border surrounds the site when:</p> <ul style="list-style-type: none"> • Layer 3 view <ul style="list-style-type: none"> • Measured Traffic view—There is no measured traffic on intrasite interfaces. • Worst-Case and Failure Impact views—No simulation analysis was performed. • Failure Impact view—Failing an intrasite L3 node or L3 circuit would not increase the traffic utilization on any L3 circuit. • Layer 1 Failure Impact view—Failing an intrasite L1 node or L1 link would not increase the traffic utilization on any L3 circuit. • Multi-layer Failure Impact view—Failing an L1 or L3 intrasite object would not increase the traffic utilization on any L3 circuit.
	The small box indicates that one or more intrasite objects is selected. In a multi-layer view, this includes both L1 and L3 intrasite objects.

Graphic	Description
	<ul style="list-style-type: none"> • Layer 3 view—Up/down triangles show that one or more egress interfaces from within the site have been collapsed. • Layer 1 view—Not applicable.
	The site failed.
	One or more intrasite objects failed. In a multi-layer view, this includes both L1 and L3 intrasite objects.
	<ul style="list-style-type: none"> • Layer 3 view <ul style="list-style-type: none"> • One or more L3 nodes or L3 circuits contained within the site are not operational because they are contained in a failed SRLG. • One or more L3 nodes or L3 circuits within the site are associated with a failed L1 port, L1 node, or L1 link. • Layer 1 view—One or more L1 nodes or L1 links contained within the site are not operational because they are contained in a failed SRLG. • Multi-layer view—One or more intrasite objects are not operational because they are contained in a failed SRLG.
	The site is inactive.
	One or more intrasite objects are inactive. In a multi-layer view, this includes both L1 and L3 intrasite objects.
	<ul style="list-style-type: none"> • Layer 3 view <ul style="list-style-type: none"> • One or more L3 nodes or L3 circuits contained within the site are not operational because they are contained in an SRLG that is not active. • One or more L3 nodes or L3 circuits within the site are associated with an inactive L1 port, L1 node, or L1 link. • Layer 1 view—One or more L1 nodes or L1 links contained within the site are not operational because they are contained in an SRLG that is not active. • Multi-layer view—One or more intrasite objects are not operational because they are contained in an SRLG that is not active.
	<ul style="list-style-type: none"> • Layer 3 view—One or more selected VPN L3 nodes exist on nodes within the site. • Layer 1 view—Not applicable.



Demands






Table 44: Demand Visualization

Graphic	Description
	"A" and "Z" mark the source and destination site/node of a demand. Blue arrows show demands routed through an interface on a circuit.
	Percentages show the proportion of a split demand (for example from ECMP) through an interface.
	Reroute of a demand under a failure.
	Demand is routed through an LSP.
	Demand is rerouted through a Fast Reroute (FRR) LSP.
	Shortest IGP path of a demand.
	Shortest latency path of a demand.
	Multicast flow and multicast flow hops.

Demand Segment Lists






Table 45: Demand Segment List Visualization

Graphic	Description
	Site containing a node segment in a demand's segment list.
	Node segment in a demand's segment list.

Graphic	Description
	Interface segment in a demand's segment list.
	Site containing the source of an LSP segment list hop in a demand's segment list.
	Source node of an LSP segment list hop in a demand's segment list.
	Site containing one or more nodes belonging to an anycast group segment in a demand's segment list.
	Node belonging to an anycast group in a demand's segment list.








External Endpoint Members

Table 46: External Endpoint Visualization

Graphic	Description
	An yellow triangle indicates that the external endpoint member selected.
If the external endpoint member is a node, this icon appears under that node in the plot. If the member is an external AS or an external AS via a node, this icon appears under all nodes in the AS.	
	Failed external endpoint member.
	External endpoint member is not operational because it is contained in a failed SRLG.
	External endpoint member is not active.
	External endpoint member is not operational because it is contained in an SRLG that is not active.

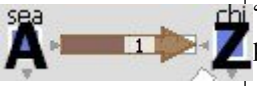

LSPs

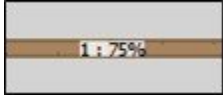
Table 47: LSP Visualization

Graphic	Description
	“A” and “Z” mark the source and destination site/node of an LSP. Brown arrows show how LSP traffic is routed.
	Rerouting of LSP traffic when a failure occurs in Convergence mode.
	FRR LSP path that would be taken if a failure required its use.
	FRR LSP that is currently rerouting LSP traffic around a failure in FRR mode.
	Shortest IGP path of an LSP.
	Shortest latency path of an LSP.
	Shortest TE path of an LSP.

LSP Paths

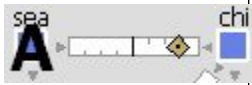





Table 48: LSP Path Visualization

Graphic	Description
	“A” and “Z” mark the source and destination site/node of an LSP path. The brown arrow indicates the LSP path is active (carrying traffic). The number in the gray rectangle (here 1) is the path option.
	A lighter brown arrow indicates the LSP path is not active and is not carrying traffic. The yellow rectangle indicates a standby path. The number in a yellow rectangle (here 2) is the path option.

Graphic	Description
	The percent value is the loadshare percentage that this LSP path is carrying for the selected LSP. In this example, the LSP loadshare percentage is 75%.

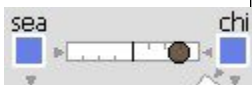
LSP Named Paths



Table 49: LSP Named Path Visualization

Graphic	Description
	“A” marks the source of a named path. The brown diamond on an interface denotes it as a named path hop. A black dot in the center denotes a strict hop.
	A brown diamond on a site denotes that the site contains a node or interface that is a named path hop. No black dot in the center denotes a loose hop. The site is shown as containing a loose hop unless all hops within the site are set to strict or exclude.
	A brown diamond on a node denotes it as a named path hop. No black dot in the center denotes a loose hop.
	“A” marks the source of a named path. A red circle with a white X on an interface denotes it as a named path hop that is set to <i>exclude</i> .
	A red circle with a white X on a site denotes that the site contains a named path hop on a node or interface and that it is set to <i>exclude</i> .
	A red circle with a white X on a node denotes it as a named path hop that is set to <i>exclude</i> .

LSP Actual Paths








Table 50: LSP Actual Path Visualization

Graphic	Description
	A dark brown circle on an interface denotes an LSP actual path hop on the interface.

Graphic	Description
	A dark brown circle on a site denotes an actual path hop on a node or interface within the site.
	A dark brown circle on a node denotes an actual path hop on the node.

Segment-Routed LSP Paths


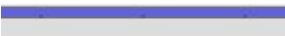







Table 51: SR LSP Path Visualization

Graphic	Description
	An interface segment in a segment-routed (SR) LSP.
	A site containing a node segment or an interface segment.
	A node segment in an SR LSP.
	Site containing the source of an LSP segment list hop in an SR LSP's segment list.
	Source node of an LSP segment list hop in an SR LSP's segment list.
	A site containing one or more nodes belonging to an anycast group.
	Node belonging to an anycast group.

L1 Objects





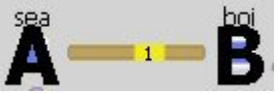
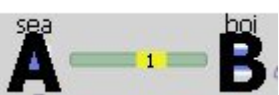
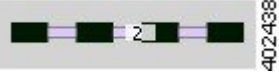

L1 Links

Table 52: Layer 1 Link Visualization

Graphic	Description
	Unselected L1 link. In the L1 Failure Impact view, indicates that failing the L1 link would not increase the traffic utilization on any L3 circuit.
	Selected L1 link.
	In the L1 Failure Impact view, the color represents the failure impact of an L1 link. This is defined as the highest utilization experienced by any L3 interface if that L1 link were to fail.
	L1 link failed.
	L1 link is not operational because of one of the following reasons: <ul style="list-style-type: none"> • An L1 node failed that is connected to the L1 link. • A site failed that contains the L1 link. • An SRLG failed that contains the L1 link. • An SRLG failed that contains an L1 node connected to the L1 link.
	L1 link is inactive.
	L1 link is inactive because of one of the following reasons: <ul style="list-style-type: none"> • An L1 node that is connected to the L1 link is inactive. • An inactive SRLG contains the L1 link.
	Multi-layer view—A blue dotted line indicates an L3-L1 link.
	Multi-layer view—A solid, thin gray line indicates a selected L3-L1 link.

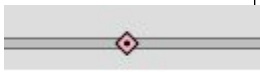







L1 Circuit Paths

Table 53: Layer 1 Circuit Path Visualization

Graphic	Description
<p>Active Sim Path</p>  <p>Sim Path</p> 	<p>Simulated L1 circuit path from L1 node “A” to L1 node “B” highlights in green when any of the following are selected:</p> <ul style="list-style-type: none"> • L1 circuit or L1 circuit path • Circuit or associated interface mapped to L1 circuit • Port circuit mapped to L1 circuit <p>The simulated active path appears in a darker green than simulated paths that are not active.</p>
<p>Standby Path</p>  <p>Non-Standby Path</p> 	<p>Path options are numerically identified within a rectangular label. Standby paths have a yellow background, and non-standby paths have a gray background. In this example, path option 2 is a standby path, and path option 3 is a non-standby path.</p>
	<p>Actual L1 circuit path.</p>
	<p>Shortest latency of an L1 circuit path.</p>
	<p>Rerouted L1 circuit path that is in use.</p>
	<p>Unrouted L1 circuit path.</p>




L1 Circuit Path Hops










Table 54: L1 Circuit Path Hop Visualization

Graphic	Description
	L1 link is a strict L1 circuit path hop.
	Site contains an L1 node that is a strict L1 circuit path hop.
	L1 node is a strict L1 circuit path hop.
	Site contains an L1 node that is a loose L1 circuit path hop.
	L1 node is a loose L1 circuit path hop.
	L1 link is an excluded L1 circuit path hop.
	Site contains an L1 node that is excluded as an L1 circuit path hop.
	L1 node is excluded as an L1 circuit path hop.

L1 Nodes, L1 Ports, and L1 Waypoints

Table 55: Layer 1 Node, L1 Port, and L1 Waypoint Visualization

Graphic	Description
	An L1 node is shown as a blue rectangle without borders.
	A selected L1 node is a darker shade of blue.
	L1 Failure Impact view—The color represents the failure impact of an L1 node. This is defined as the highest utilization experienced by any L3 interface if that L1 node were to fail.

Graphic	Description
	Indicates that failing the L1 node would not increase the traffic utilization on any L3 circuit.
	L1 node failed.
	L1 port failed.
	L1 node is not operational because it is contained in a failed site or SRLG.
	L1 node is inactive.
	L1 port is inactive.
	L1 node is not operational because an SRLG containing it is inactive.
	L1 link waypoint.
	Selected L1 link waypoint.

