



## Exporting Routes

---

The ability to export routes and tables facilitates the exchange of information in a text-editable format. It also supplements the analysis of routes and plan file objects. For instance, you could export demand routes using L1 link to determine which L1 links are used by demands. Since you can export routes that changed due to failure states, you can fine-tune your analysis of how failures impact the network.

- [Export Routes](#)—Routes of demands, shortest IGP, TE, and latency paths, LSPs, LSP paths, circuits, Layer 1 (L1) circuits, and L1 circuit paths. Most of these have options for specifying the hop type.
- `<Xref>`Export Explicit LSP Path Settings—Exports all explicit LSP path settings.
- [Export L1 Link Wavelength Utilization](#)—Exports a file containing a `<LambdaUtil>` table that identifies all of the L1 circuit path Lambda Sim values that are routed on each L1 link.
- [Export Current Table](#)—Exports the current table that is showing in the WAE Design GUI.

## Export Routes

You can export routes using the WAE Design GUI or using `export_routes` CLI tool.

This tool creates one of the following tables, depending on what you are exporting.

- `<CircuitHops>`
- `<DemandHops>`
- `<L1CircuitHops>`
- `<L1CircuitPathHops>`
- `<LSPHops>`
- `<LSPPathHops>`
- `<ShortestIGPPathHops>`
- `<ShortestLatencyPathHops>`
- `<ShortestTEPathHops>`

The table generated is output to a .txt file containing only this table. Their key columns are a combination of the key column identified for their object type plus a Step column. The Step column is an integer that identifies the sequence of this hop in the route path. For example, the `<Demands>` table has key columns of Name, Source, Destination, and ServiceClass. Therefore, the `<DemandHops>` table has these four columns to uniquely identify the demand, plus a Step column to identify the hop sequence.

Each exported hops table includes the following columns to specify details of the hop and its position with respect to other hops in the route. Note that the routes of demands might be split to account for ECMP routing or multicast. So demand hops might not be listed sequentially as a hop could be followed by two next hops as the route and traffic split into two.

- **UnresolvedHop**—The hop type and complete definition. For example, `if{cr1.atl|to_cr1.hst}` means this hop is an interface with a source node of `cr1.atl` and an egress interface of `to_cr1.hst`. For a complete list of these object notations, see the [Plan Files and Tables](#) chapter.
- **PreviousStep**—Identifies the previous step or steps in the route. As such, this identifies the previous hop in the route. For example, if you are on the row corresponding to Step 4, and the PreviousStep is 2, then the previous hop is contained in the row corresponding to Step 2.  
If the PreviousStep column is empty, then this is the first hop in the route.
- **NextStep**—Identifies the next step in that table. As such, this identifies the next hop in the route.  
If the NextStep column is empty, then this is the last hop in the route.
- **TrafficProportion**—The proportion of traffic on this hop compared to the other hops in the route. Although this value is usually 1, in the case of ECMP routing, it could be less than 1.

Optionally, you can fail objects prior to exporting their routes and then export only the routes that changed due to that failed state.

For More Information ...	See ...
Export routes through WAE Design GUI	<a href="#">Export Routes Using the GUI</a>
Export routes using CLI tool	<code>export_routes</code> Help output
Key column definitions per table	<code>\$CARIDEN_HOME/docs/table_schema.html</code> file; key columns are highlighted in orange
Key columns	<a href="#">Plan Files and Tables</a> chapter
WAE Design tables	<a href="#">Plan Files and Tables</a> chapter
Object notation	<a href="#">Plan Files and Tables</a> chapter
Fail objects prior to exporting routes	<i>WAE Design User Guide</i>

## Example Export Route Tables

[Table 5-1](#) is an example `<LSPHops>` table that was created using active LSPs with interface hops. The `cr2.par_cr1.fra` LSP has no interface hops in its route. In the table, it has only one step and no previous or next steps. The `cr1.ams_cr1.par` LSP has three interface hops (steps 1, 2, and 3) between source (`cr1.ams`) and destination (`cr1.par`).

**Table 5-1 Example <LSPHops> Table**

Name	Source	Destination	Step	UnresolvedHop	Previous Step	Next Step	Traffic Proportion
<code>cr2.par_cr1.fra</code>	<code>cr2.par</code>	<code>cr1.fra</code>	1	<code>if{cr2.par to_cr1.fra}</code>			1
<code>cr1.ams_cr1.par</code>	<code>cr1.ams</code>	<code>cr1.par</code>	1	<code>if{cr1.ams to_cr2.lon}</code>		2	1

Name	Source	Destination	Step	UnresolvedHop	Previous Step	Next Step	Traffic Proportion
cr1.ams_cr1.par	cr1.ams	cr1.par	2	if{cr2.lonl{to_cr1.lon}}	1	3	1
cr1.ams_cr1.par	cr1.ams	cr1.par	3	if{cr1.lonl{to_cr1.par}}	2		1

Table 5-2 is an example <DemandHops> table that was created using demands with interface hops (Figure 5-1). The demand route in this example splits into multiple ECMP paths.

Figure 5-1 Example Demand Route Shown in <DemandHops> Table

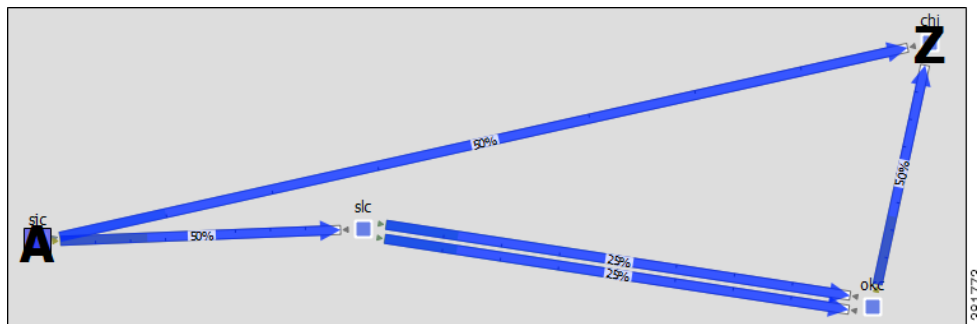


Table 5-2 Example <DemandHops> Table with ECMP Route

Name	Source	Destination	Service Class	Step	UnresolvedHop	Previous Step	Next Step	Traffic Proportion
	sjc	chi	Default	1	if{sjcl{to_slc}}		3;4	0.5
	sjc	chi	Default	2	if{sjcl{to_chi}}			0.5
	sjc	chi	Default	3	if{slcl{to_okc}}	1	5	0.25
	sjc	chi	Default	4	if{slcl{to_okc}[1]}	1	5	0.25
	sjc	chi	Default	5	if{okcl{to_chi}}	3;4		0.5

If you were to fail the circuit between sjc and slc and export only routes that changed during a failed state, the resulting exported route table would be as shown in Table 5-3.

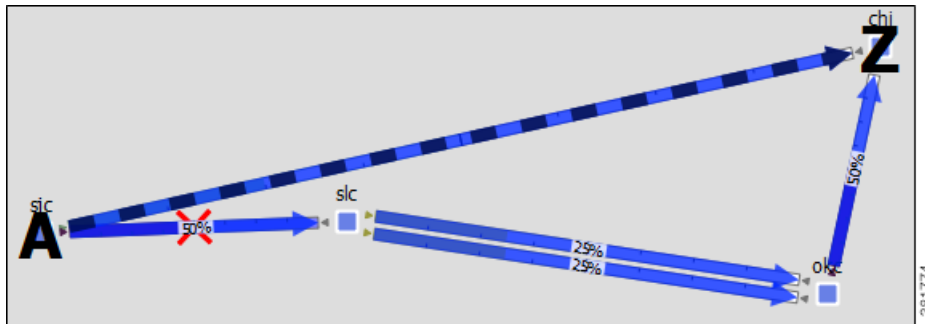


Table 5-3 Example <DemandHops> Table After Failing a Circuit

Name	Source	Destination	Service Class	Step	UnresolvedHop	Previous Step	Next Step	Traffic Proportion
	sjc	chi	Default	1	if{sjc {to_slc}}			1

## Export Routes Using the GUI

From the WAE Design GUI, follow these steps to export routes to the hops tables. Hops tables are created for the object that is selected in the dialog box. For example, if you select Circuits, WAE Design creates a <CircuitHops> table containing all circuits in the plan file.

If you want to export routes only for those objects whose route changed due failure state, fail all relevant objects before exporting the routes and select the “Only export routes changed by failure state” option. Alternatively, you could run Simulation Analysis and then fail one or more objects to the worst-case failure. For more information, see the *WAE Design User Guide*.

When exporting circuits, only L3 circuits using L1 hops are exported.

When exporting L3 circuits and L1 circuits, the Active Path Sim property in the L1 Circuits table is considered. This property identifies which L1 circuit path is currently being used by the L1 circuit. If an L1 circuit path is not available due to failure, the L1 circuit’s Active Path Sim is updated to the L1 circuit path that is being used. When exporting L1 circuit paths, all routed paths are exported.

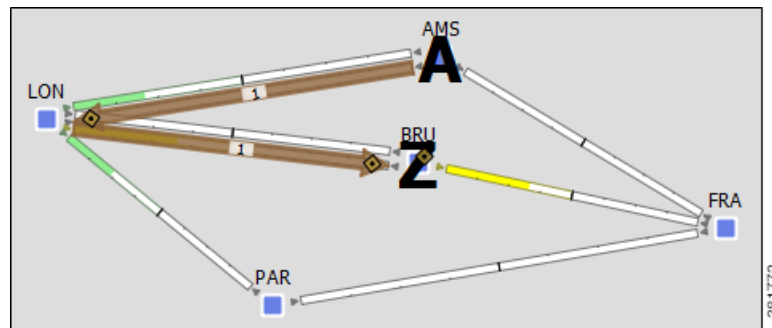
- 
- Step 1** Select the Export->Export Routes menu. The Export Routes dialog box appears.
  - Step 2** Select the object for which you want to export routes.
  - Step 3** Except for circuits and L1 circuits, select the hop type that you want to export. For circuits, L1 circuits, and L1 circuit paths, the only available option is Layer 1 links hops.
  - Step 4** If exporting only routes that changed due to a failure, select the associated option.
  - Step 5** Click OK, and save the .txt file to the location of your choice.
-

## Export Explicit LSP Path Settings

Tab-delimited .txt files listing all explicit LSP path settings are created by selecting the File->Export->Explicit LSP Settings menu in the WAE Design GUI, or using `export_routes` CLI tool. The Hops column lists the explicitly routed path of each LSP in object notation. For a complete list of these object notations, see the [Plan Files and Tables](#) chapter.

[Table 5-4](#) is an example output file. In addition to these columns, the file also contains SetupBW, SetupPri, and HoldPri columns. [Figure 5-1](#) shows the first LSP path option listed in [Table 5-4](#).

**Figure 5-2** Path Option 1 for the `cr1.ams_er1.BRU` LSP



**Table 5-4** Example File Containing Explicit LSP Path Settings

Name	LSP	Source	Destination	Preference	Hops
cr1.ams_er1.BRU_1	cr1.ams_er1.BRU	cr1.ams	er1.BRU	1	if{er1.BRU {to_cr1.BRU}},if{cr1.BRU {to_cr2.lon}},if{cr2.lon {to_cr1.ams}}
cr1.ams_er1.BRU_1	cr1.ams_er1.BRU	cr1.ams	er1.BRU	2	if{er1.BRU {to_cr2.BRU}},if{cr2.BRU {to_cr1.lon}},if{cr1.lon {to_cr2.ams}},if{cr2.ams {to_cr1.ams}}

## Export L1 Link Wavelength Utilization

Changes in the routing and Lambdas of L1 circuits paths affect the utilized wavelengths on L1 links. Any L1 failure or a site failure could cause the L1 circuit path to be rerouted. If Auto Lambda is enabled, then the L1 circuit path could be forced to select a Lambda on a different path option in order to avoid lambda blocking. For more information about Layer 1, see the *WAE Design User Guide*.

The Export Lambda Utilization tool enables you to create and export a file containing a <LambdaUtil> table that identifies all of the L1 circuit path Lambda Sim values that are routed on each L1 link. This is useful for determining how many L1 circuits paths are using an L1 link, to determine how fragmented the wavelength usage is across L1 links, and to perform more elaborate wavelength assignments.

The table shows whether there is a failure, the L1 link, and the Lambda values of the L1 circuit paths traversing it. Each Lambda column correlates to a different L1 circuit path. To see the names of the L1 circuit paths, use the “Specify L1 circuit paths” option.

**Example:** In this simple network there are no failures, and only four Lambdas are used. The ams-lon L1 link has an L1 circuit path using Lambda1 and none using the remaining Lambdas. The lon-par L1 link has four L1 circuit paths using it.

---

<LambdaUtil>

Failure	Name	L1NodeA	L1NodeB	Lambda1	Lambda2	Lambda3	Lambda4
none	ams-lon	cr1.ams	cr2.lon	T	F	F	F
none	lon-par	cr2.lon	cr1.par	T	T	T	T

To see the effects of L1 link, L1 node, and sites failures on the wavelength utilization of L1 links, select the relevant failure sets.

**Example:** Row 2 shows that the L1lnk{romllonlrom-lon} failure causes an L1 circuit path to switch from one Lambda 1 to another lambda on the ams-lon L1 link or causes it to switch path options that it is using. Row 3 shows that the L1lnk{parlromlpar-rom} failure prevents L1 circuit paths from using either Lambda1 or Lambda2 when traversing the ams-lon L1 link.

---

<LambdaUtil>

	Failure	Name	L1NodeA	L1NodeB	Lambda1	Lambda2	Lambda3	Lambda4
1	none	ams-lon	cr1.ams	cr2.lon	T	F	F	F
2	L1lnk{romllonlrom-lon}	ams-lon	cr1.ams	cr2.lon	F	T	T	T
3	L1lnk{parlromlpar-rom}	ams-lon	cr1.ams	cr2.lon	F	F	T	T

- 
- Step 1** If you want to export the lambda utilization for only selected L1 links, first select those L1 links.
  - Step 2** Select the File->Export->Lambda Utilization menu.
  - Step 3** Select which L1 links to use: All, those selected, or a selected tag.
  - Step 4** If you want to run Simulation Analysis on L1 links, L1 nodes, or sites, select one or more of these options.
  - Step 5** If you want to know the name of the L1 circuit path for each Lambda referenced in the table, select "Specify L1 circuit paths." Then click OK.
- 

## Export Current Table

To export the table that is currently showing in the WAE Design GUI, select the File->Export->Table menu. Only the columns that are showing appear in the exported table. To show and hide tables and columns, use the View->Tables menus.

You can also export a table or portions of a table using a simple cut-and-paste method. Select the rows and columns, use Ctrl-C (Cmd-C on Mac) to copy the selection, and then paste it to the desired document. If you paste this information into an Excel document, the data is correctly placed into the Excel rows and columns.

## Related Topics

- *WAE Design User Guide*

