



## 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 1](#)
- [Reports, on page 3](#)

## Running the L1 Circuit Path Optimization Tool

*Table 1: Feature History*

Feature	Release Information	Description
Enhancements to L1 Circuit Path Optimization Tool	Cisco WAE Release 7.6.0	The new options <code>Path Search Space</code> and <code>Export Candidate Paths</code> are added to 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). You can specify the following:
  - **Path Search Space**—If specified the optimizer expands the solution search space to include paths up to the specified limit. Only two L1 circuits may be set per disjoint group and each L1 circuit may only have one L1 circuit path. Path Search Space takes input as an integer and the default value is 1. If you use the default value, the existing behavior prior to 7.6 would be retained.
  - **Export Candidate Paths**—Is the file to which the list of candidate paths will be exported to.
- **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