



# Get Started with Function Packs

Function packs extend the functionality of Crosswork Optimization Engine. Function packs can be developed to support additional use cases that are not available with the Crosswork Optimization Engine base software package. By leveraging SDKs and APIs, network operators can build additional function packs to support more optimization workflows.

This section describes functions packs that can be purchased with Crosswork Optimization Engine and how to configure and use them. To obtain a Crosswork Optimization Engine function pack license, contact your Cisco representative.

This section contains the following topics:

- [Install Function Packs, on page 1](#)
- [Update Network Configuration for Function Packs, on page 3](#)
- [Bandwidth on Demand, on page 3](#)
- [Bandwidth Optimization, on page 8](#)

## Install Function Packs

To install the Bandwidth on Demand (BWoD) or Bandwidth Optimization function pack, do the following:

### Before you begin

You must have the applicable [Bandwidth on Demand](#) or [Bandwidth Optimization](#) function pack license.

**Step 1** Connect to the Cisco Crosswork Optimization Engine VM via SSH.

**Step 2** Shift to sudo user access (**sudo su**).

**Step 3** (Optional) If you do not know the orchestrator pod name, enter the following command:

```
# kubectl get pods -n kube-system | grep robot-orch  
  
robot-orch-5d4d79d549-81zwr          1/1      Running    0          6d
```

In this example, `robot-orch-5d4d79d549-81zwr` is the orchestrator pod name.

**Step 4** Log into the orchestrator pod:

```
# kubectl exec -it <pod_name> bash -n kube-system
```

For example:

```
# kubectl exec -it robot-orch-5d4d79d549-8lzwr bash -n kube-system
```

**Step 5** Open the pod.conf file (`vi /root/pod.conf`) and add the following service names to the "Pods" and "Services" section and save changes.

- optima-bwopt
- optima-bwod

**Figure 1: Example: pod.conf File**

```
namespace : default
pods:
- robot-dlminmgr
- tyk
- helios
- robot-grafana
- neo4j-topo-svc
- robot-topo-svc
- topo-ui-service
- robot-ui
- cas
- optima-traffic
- optima-engine
- optima-pce-dispatcher
- optima-bwopt
- optima-bwod
service:
- robot-dlminmgr
- tyk
- helios
- robot-grafana
- neo4j-topo-svc
- robot-topo-svc
- topo-ui-service
- robot-ui
- topo-ui-service
- cas
- optima-traffic
- optima-engine
- optima-pce-dispatcher
- optima-bwopt
- optima-bwod
```

**Step 6** Update the orchestrator. The update may take up to 30 seconds.

```
# robotctl update
Are you sure? [y/n]: y
Operation [ update ] succeeded
```

**Step 7** Start the function packs:

```
# robotctl start optima-bwopt
Are you sure? [y/n]: y
Operation [ start,optima-bwopt ] is success

#robotctl start optima-bwod
Are you sure? [y/n]: y
Operation [ start,optima-bwod ] is success
```

**Step 8** Exit and confirm the function packs are running.

```
# exit
# kubectl get pods
```

You should see `optima-bwod-xxx` and `optima-bwopt-xxx` listed.

**Step 9** Update network configuration. See [Update Network Configuration for Function Packs, on page 3](#).

# Update Network Configuration for Function Packs

The following network configuration parameters must be updated prior to using the Bandwidth on Demand or Bandwidth Optimization function packs.

```
# kubectl exec -it optima-engine-55665ff5cb-vvqmqz bash
# vi /config/optima-engine.json

{
  "verbosity":60,
  "heartbeat":30,
  "traffic-notification-interval":60,
  "log-toposvc-messages":true,
  "measured-traffic-threshold":0.01,
  "initial-notification-delay":330,
  "use-srtm-policy-traffic":false
}
```

The following parameters can be useful and updated per your preference:

- **measured-traffic-threshold**—Sends traffic update notification if the traffic value has changed at least by the percentage you indicated. This can be any value between 0 (0%) and 1 (100%). In the example above, notification of traffic updates will be sent if the traffic value changes by at least 1% .
- **initial-notification-delay**—The delay in seconds before the first notification goes to function packs after startup.
- **use-srtm-policy-traffic**—Whether to update the policy measured traffic from SRTM. In the example above, it is set to **false**, so policy measured traffic only comes from SNMP.



---

**Note** Function packs rely on telemetry data received from devices. Confirm that devices are configured (see [Prerequisites for Device Telemetry](#)).

---

## Bandwidth on Demand

The Bandwidth on Demand (BWoD) function pack provides a bandwidth-aware Path Computation Element (PCE) to derive SR policy paths with requested bandwidth when available. Computed paths are deployed to the network through SR-PCE. BWoD continuously monitors link utilization to ensure no congestion occurs along the path. If conditions change in the network which causes link utilization to exceed the congestion threshold set by the user, BWoD automatically reoptimizes the policy path.

BWoD utilizes a near real-time model of the network along with a demand matrix derived from telemetry-based Segment Routing Traffic Matrix (SRTM) reporting to ensure BWoD policies meet their bandwidth constraints. Users may fine tune the behavior BWoD, affecting the path it computes, through the selection of application options including network utilization threshold (definition of congestion) and path optimization objectives. The BWoD function pack works as a bandwidth-aware PCE for SR policies created through the Crosswork Optimization Engine UI, and for SR policies created through CLI configuration on a headend with delegation to SR-PCE. In the latter case, SR-PCE will subdelegate the SR policy with bandwidth constraint to BWoD for path computation and relay the computed path returned by BWoD to the headend for instantiation.

### Operation Modes

There are two modes of operation for BWoD based on the "Priority" option setting for the application. In non-Priority mode, BWoD takes into account all traffic in the network when computing a path for a SR policy with bandwidth constraint. In this case, BW SR policies compete with all other traffic for resources and may be provided a path that is longer to avoid congestion on links along the shortest path. Note that




---

**Note** In non-Priority mode, BWoD *should not* be enabled at the same time as the Bandwidth Optimization function pack to ensure they do not conflict.

---

The Priority mode allows BWoD to ignore all other traffic in the network that is not flowing through a BWoD SR policy and give its policies priority treatment when computing paths. This means that BWoD policies are only contending for resources with other BWoD policies and will likely take the shortest path unless there are links that include a significant amount of other BWoD traffic.




---

**Note** To mitigate any congestion that may occur by ignoring other traffic, the Bandwidth Optimization application *should* be used in conjunction with BWoD in the Priority mode to shift other traffic away from any hotspots caused by the BWoD traffic.

---

The other traffic may then be sent over alternate (possibly longer) paths to mitigate congestion in this case, while BWoD maintains its policies along the shortest paths.

## Important Notes and Limitation for BWoD

Consider the following notes and limitations when using BWoD:

- If BWoD cannot find a path for a policy that guarantees its requested bandwidth, BWoD will attempt to find a *best effort* path. The best effort path may not meet the requested bandwidth for the policy.
- BWoD temporarily pauses operation whenever the Optimization Engine model is unavailable due to an Optimization Engine restart or a rebuild of the topology from Topology Services. Any requests to BWoD during this time are rejected. When the model becomes available and BWoD receives 2 traffic updates from the Optimization Engine, BWoD will resume normal operation.
- BWoD uses simulated traffic based on measured SRTM data to determine link utilizations for computing BWoD paths. The simulated interface utilization BWoD utilizes should closely align with the SNMP-based interface utilization that is displayed in the Optimization Engine UI. However, due to various factors, including SNMP polling cadence and rate averaging techniques, they may differ at times. This can result in scenarios like a link along a BWoD path appearing to be congested in the UI and BWoD not re-optimizing it.

## Configure Bandwidth on Demand

Do the following to enable and configure Bandwidth on Demand.

### Before you begin

Bandwidth on Demand must be installed (see [Install Function Packs, on page 1](#)).



**Note** If a policy is configured on the router with affinity constraints, they are not considered in the **Bandwidth on Demand** optimization calculation.

- Step 1** From the main menu, choose **Optimization Engine > Function Packs > Bandwidth on Demand**.
- Step 2** From the **Enable** tile, toggle the slider to **True**.  
Notice that each time a tile is updated it turns blue.
- Step 3** Select one of the following **Primary Objectives**:
- **Maximize Available Bandwidth**—Computes an SR policy path maximizing the overall available bandwidth in the network. This setting generally attempts to maximize usable network capacity at the expense of potentially longer paths.
  - **Metric Minimization**—Computes an SR policy path minimizing the metric selected. This setting generally results in the shortest available paths for a metric type.
- Step 4** In the **Link Utilization** tile, enter the congestion constraint (in percentage). When the Bandwidth on Demand application searches a path for the policies being delegated, it will avoid any paths that may exceed the congestion utilization threshold.
- Step 5** In the **Reoptimization Interval** tile, enter the duration (in seconds) after which the paths will be reoptimized if conditions in the network change. This is a count down timer where the BWoD policy will wait to reoptimize until this duration has expired.
- Step 6** In the **Metric Reoptimization Interval** tile, enter the duration (in seconds) after which the paths can be reoptimized for metric optimization. If the bandwidth constraint is still being met, but a shorter IGP or TE path is available, BWoD will not run reoptimization until the timer has expired. This value is meant to dampen frequent path change and reoptimizations in the network.
- Step 7** From the **Priority Mode** tile, toggle the slider to **True** if you have also enabled Bandwidth Optimization. See [Bandwidth on Demand, on page 3](#) for more information on Priority Mode.
- Step 8** Click the **Advanced** tab for more advanced configuration (see the following table for field descriptions).
- Step 9** Click **Commit Changes** to save the configuration.

**Table 1: Advanced Bandwidth on Demand Fields**

Field	Description
Private New SR Policies	If <b>True</b> , all policies that are created using Bandwidth on Demand are private.

Field	Description
<b>SR Policy Traffic</b>	<p>Determines the type of bandwidth optimization is performed with each policy.</p> <ul style="list-style-type: none"> <li>• <b>Simulated</b>—Uses the current simulated traffic on the BWoD provisioned SR policies for optimization calculations.</li> <li>• <b>Measured</b>—Uses the current measured traffic of BWoD provisioned SR policies for optimization calculations.</li> <li>• <b>Max Simulated Requested</b>—Uses the maximum value between the current simulated traffic on BWoD provisioned SR policies or the amount of bandwidth requested for optimization calculations.</li> <li>• <b>Max Measured Requested</b>—Uses the maximum value between the current measured traffic on BWoD provisioned SR policies or the amount of bandwidth requested for optimization calculations.</li> </ul>
<b>Deployment Timeout</b>	The time (in seconds) to wait for a PCE dispatcher response.
<b>Update Throttle</b>	This option is a knob for throttling updates from the Crosswork Optimization Engine. By setting the knob, it tells BWoD to only accept 1 update per x seconds. Set it to 0 to disable the throttle.
<b>Debug Optimizer</b>	
<b>Debug Opt Max Plan Files</b>	The maximum number of debug plan files you would like to save.
<b>Debug Opt</b>	If <b>True</b> , debug log files will be saved (see <a href="#">Collect and Share Cisco Crosswork Network Automation Logs and Metrics</a> ).

**What to do next**

Create a Bandwidth on Demand SR policy (see [Create Bandwidth on Demand SR Policies, on page 6](#)).

## Create Bandwidth on Demand SR Policies

This task creates a dynamic path steering policy that finds the optimal path that requires a persistent bandwidth and IGP, TE, or delay metrics.

**Before you begin**

You must have the Bandwidth on Demand function pack installed and enabled (see [Install Function Packs, on page 1](#) and [Configure Bandwidth on Demand, on page 4](#)).

**Step 1** From the **SR Policies** table, click **Create**.

**Step 2** Enter the following SR policy values:

a) Required fields:

- **Headend**—Where the SR policy is instantiated. Note: You can either select a node (from the map or drop-down list) or enter part of the node name to filter the headend and endpoint node entries.
- **Endpoint**—The destination of the SR policy.
- **IP Address**—After the endpoint is selected, the SID list is populated and you can select the loopback IP address.

- **Color**—A numerical value that distinguishes between two or more policies to the same node pairs (Headend – Endpoint). Every SR policy between a given headend and endpoint must have a unique color.
- **Path Name**—Enter a name for this SR policy path. SR policy paths from the same headend must be unique. Policy path names are not case sensitive.

b) Optional values:

- **Description**—Enter details or a description of this policy.
- **Explicit Binding SID**—The binding segment is a local segment identifying an SR policy. Each SR policy is associated with a binding segment ID (BSID). The BSID is a local label that is automatically allocated for each SR policy when the policy is instantiated. If you wish to use a specific segment ID, rather than the default one that is automatically assigned, then enter it here.
- **Profile ID**—Identification used to associate an SR policy with a set of features applied to the policy by the headend. It should correspond with a profile configured on the headend.

**Step 3** Under Policy Path, click **Bandwidth On Demand**.

**Note** This option is only available when the Bandwidth on Demand function pack is installed and enabled.

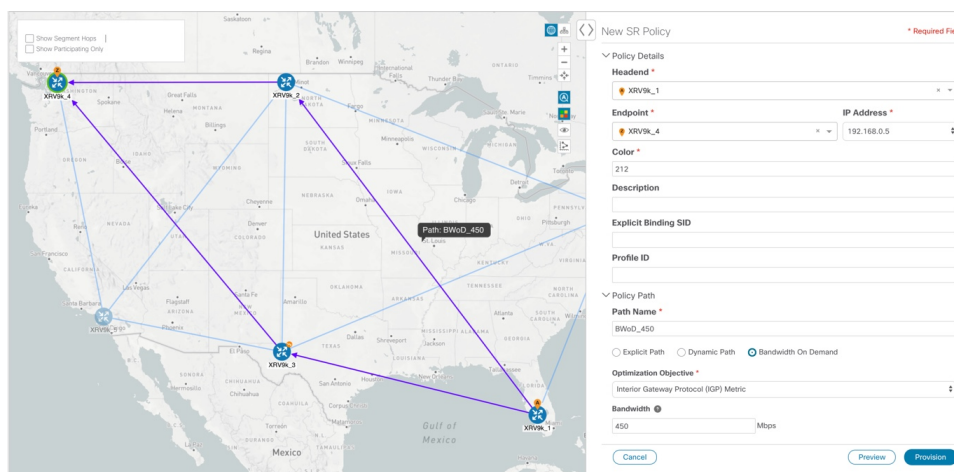
**Step 4** **Optimization Objective**—Depending on the bandwidth constraint chosen (see [Configure Bandwidth on Demand, on page 4](#)), select the specific metric to optimize for Bandwidth on Demand

- **Interior Gateway Protocol (IGP) Metric**—Minimizes total path IGP metric.
- **Traffic Engineering (TE) Metric**—Minimize total path TE metric.
- **Latency**—Minimize total path latency.

**Step 5** **Bandwidth**—Enter the requested bandwidth amount.

**Step 6** Click **Preview**. The path is highlighted on the map.

**Figure 2: Bandwidth on Demand SR Policy Example**



**Step 7** If you are satisfied with the policy path, click **Provision**.

**Step 8** When the policy is provisioned successfully, a window appears with the following options:

- **View SR Policy List**—Displays the **SR Policies** table that lists all SR policies including the one that was just created.
- **Create New**—Allows you to create another SR policy.

## Troubleshoot BWoD

The following are some of the most common error conditions for BWoD and some possible corrective actions that may fix the issue.

**Table 2: Errors**

Error Event Message	Possible Causes and Recommended Corrective Action
OptimaModelError	<p>The network model used by BWoD from the Optimization Engine is corrupt or is missing key data that is needed to properly support BWoD. Possible causes include network discovery issues or synchronization problems between the Optimization Engine and Topology Services. Try restarting the Optimization Engine pod to rebuild the model.</p> <p>This error can also occur if the time required to discover a policy and add it to the model after it has been deployed exceeds the <b>Deployment Timeout</b> option set for BWoD. The default is 30 seconds which should suffice for small to medium sized networks. However, larger networks may require additional time.</p>
NATSTimedOutError	<p>The deployment of a bandwidth policy through SR-PCE exceeds the <b>Deployment Timeout</b> option set for BWoD. Increase the <b>Deployment Timeout</b> option to allow for additional time for deployments in larger networks.</p>
Traceback or other errors found in the log file	Please contact your Cisco service representative.

## Bandwidth Optimization

The Bandwidth Optimization (BWOpt) function pack provides automated SR policy based tactical traffic engineering capability to detect and mitigate congestion in your network. It achieves this through a real-time view of the network topology overlaid with a demand matrix built through telemetry-based Segment Routing Traffic Matrix (SRTM). BWOpt uses the threshold interface utilization requested by the user and compares it to the actual utilization in the network. When interface congestion is detected by BWOpt, it attempts to shift traffic away from hot spots through the use of tactical traffic engineered SR policies which are deployed to the network via SR-PCE. As network conditions (topology and/or traffic) change over time, BWOpt will continue to monitor interface utilization and manage any tactical SR policies deployed, including changing their paths and/or removing them from the network when deemed no longer necessary.



## Important Notes and Limitations for BWOpt

Consider the following notes and limitations when using BWOpt:

- Only traffic that is not in an SR policy or existing BWOpt SR policy can be rerouted to mitigate congested links. BWOpt will not shift traffic in existing SR policies that it did not create. This may prevent it from being able to mitigate congestion if most of the traffic on the congested link is in non-BWOpt SR policies.
- BWOpt relies on the PCC's autoroute feature to steer traffic into the tactical SR policies it creates. Autoroute is applied to these policies through the proper **Profile ID** option set in BWOpt (to align with configuration on the PCC associating that Profile ID with autoroute feature). This is critical to tactical SR policies shifting traffic away from congested links.
- BWOpt does not support multi-area or multi-level IGP (see "IGP and Inter-AS Support" in the *Cisco Crosswork Optimization Engine Installation Guide*). Autoroute will not properly steer traffic onto inter-area or inter-level tactical SR policies. So, although they can be provisioned, traffic will not use them. Therefore, BWOpt will be ineffective if enabled in this environment.
- BWOpt uses simulated traffic based on measured SRTM data to determine link utilizations and when to mitigate congestion. The simulated interface utilization that BWOpt monitors should closely align with the SNMP-based interface utilization that is displayed in the Optimization Engine UI. However, due to various factors, including SNMP polling cadence and rate averaging techniques, they may differ at times. This can result in scenarios like a link appearing to be congested in the UI and BWOpt not reacting.
- BWOpt only creates tactical SR policies on PCCs that are sources of SRTM telemetry data. Only these nodes (typically provider edge routers) provide the telemetry-based data needed to create simulated traffic demands in the internal model representing the traffic from that node to other PE nodes in the network.
- Only solutions that produce interface utilization below the threshold (set across all interfaces) will be deployed. If BWOpt is unable to mitigate congestion across the entire network, it will not deploy any tactical SR policies and a "Network Congested. BWOpt unable to mitigate." alarm is set. This alarm is unset when congestion either subsides on its own or can be addressed successfully through BWOpt tactical SR policy deployments.
- BWOpt temporarily pauses operation whenever the Optimization Engine model is unavailable due to an Optimization Engine restart or a rebuild of the topology from Topology Services. When this occurs, an alarm indicating this condition is set by BWOpt. During this time, BWOpt will not evaluate congestion in the network. All currently deployed tactical SR policies are maintained, but will not be modified or deleted. As soon as the model becomes available, the alarm is cleared and BWOpt will resume normal operation.

## Configure Bandwidth Optimization

After Bandwidth Optimization is enabled, Cisco Crosswork Optimization Engine monitors all interfaces in the network for congestion based on the configured utilization threshold. When the utilization threshold is exceeded, it automatically deploys tactical policies and moves traffic away from the congested links. When congestion is alleviated, Bandwidth Optimization automatically removes the tactical SR policy.

Do the following to enable and configure Bandwidth Optimization.

### Before you begin

Bandwidth Optimization must be installed (see [Install Function Packs, on page 1](#)).




---

**Note** Bandwidth Optimization should only be enabled (Enable option set to True) together with Bandwidth on Demand if the Bandwidth on Demand "Priority Mode" option is set to True. Otherwise, their actions may conflict resulting in unpredictable behavior.

---


- Step 1** From the main menu, choose **Optimization Engine > Function Packs > Bandwidth Optimization**.
- Step 2** From the **Enable** tile, toggle the slider to **True**.
- Notice that each time a tile is updated it turns blue.
- Step 3** Select one of the following **Optimization Objectives**:
- **Maximize Available Bandwidth**—Leads to preferred paths that result in higher available bandwidth values on interfaces.
  - **Minimize the IGP/TE/Delay**—Leads to preferred paths that result in lower total IGP/TE or Delay metrics..
- Step 4** In the **Color** tile, enter a color value to be assigned to Bandwidth Optimization SR policies.
- Step 5** In the **Utilization Threshold** tile, enter a percentage that represents the interface utilization threshold for congestion. Traffic utilization on any interface exceeding this threshold will trigger Bandwidth Optimization to attempt to mitigate.
- Step 6** In the **Utilization Hold Margin** tile, enter a percentage that represents the utilization below the threshold required of all interfaces to consider removing existing tactical SR policies. For example, if the Utilization Threshold is 90% and the Utilization Hold Margin is 5%, then tactical SR policies deployed by Bandwidth Optimization will only be removed from the network if all interface utilization would be under 85% (90 - 5) without the tactical policy in the network. This serves as a dampening mechanism to prevent small oscillations in interface utilization from resulting in repeated deployment and deletion of tactical SR policies. Utilization Hold Margin must be between 0 and the Utilization Threshold.
- Step 7** In the **Maximum Global Reoptimization Interval** tile, enter the maximum time interval (in minutes) to reoptimize the existing tactical SR policies globally. During a global reoptimization, existing tactical policies may be rerouted or removed to produce a globally more optimal solution. Set to 0 to disable.
- Step 8** From the **Delete Tactical SR Policies when Disabled** tile, toggle the slider to **True** if you want all deployed tactical SR policies deleted when Bandwidth Optimization is disabled.
- Step 9** In the **Profile ID** tile, enter the profile ID that will be assigned to tactical SR policies that are created. Enter 0 if you do not wish to assign a profile ID.
- Step 10** Click the **Advanced** tab for more advanced configuration (see the following table for field descriptions).
- Step 11** Click **Commit Changes** to save the configuration. Cisco Crosswork Optimization Engine begins to monitor network congestion based on the threshold that was configured.
- Note**
- You can easily turn Bandwidth Optimization on or off by toggling the **Enable** slider to **True** or **False**.
  - Click  to view events relating to instantiation and removal of tactical SR policies created by Bandwidth Optimization.
-

Table 3: Advanced Bandwidth on Demand Fields

Field	Description
<b>Fix Tactical SR Policy Duration</b>	The minimal time (in seconds) between the creation of a new tactical SR policy to when it can be removed or modified. This serves as a dampening factor to control the rate of change to deployed tactical SR policies.
<b>Removal Suspension Interval</b>	The time (in seconds) between any tactical SR policy change to when any tactical SR policy can be removed or modified. This allows SRTM to converge after a tactical SR policy creation, allowing traffic on the policy to be reported accurately.
<b>Deployment Timeout</b>	The maximum time (in seconds) to wait until deployment of tactical SR policies are confirmed. The value assigned should be larger for larger networks to account for the increased processing time needed by SR-PCE to deploy an SR policy. Tactical SR policies not confirmed before this timeout are declared failed and Bandwidth Optimization will disable itself for troubleshooting.
<b>Debug Optimizer</b>	
<b>Debug Opt Max Plan Files</b>	The maximum number of optimizer debug files written to disk.
<b>Debug Opt</b>	If <b>True</b> , optimizer debug files will be saved to disk in the <b>/tmp</b> dir of the Bandwidth Optimization container.

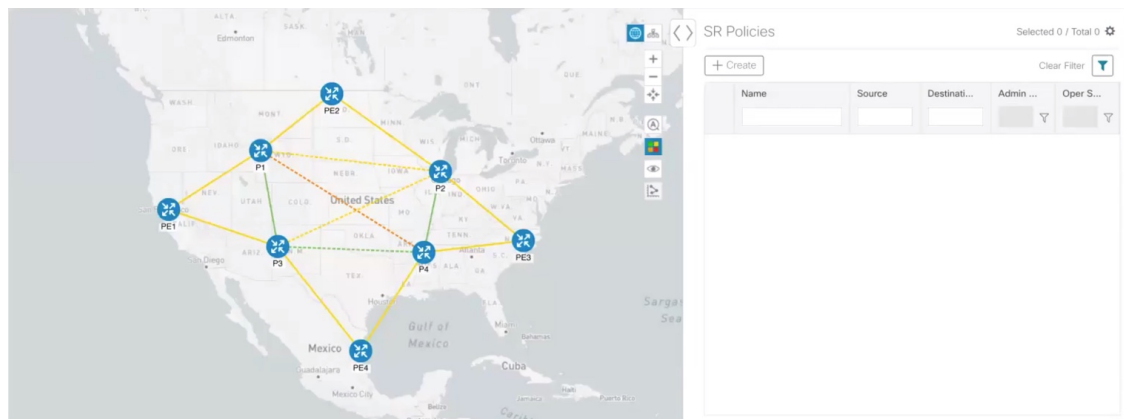
## Bandwidth Optimization Example

In this example, we have configured and enabled Bandwidth Optimization with the following options.

Figure 3: Bandwidth Optimization Configuration

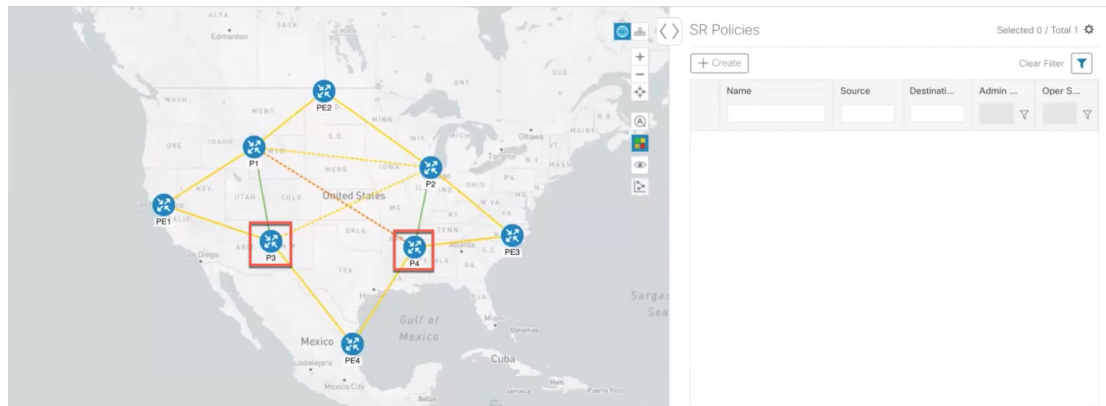
Below is a network with various devices and links that span the United States. Note that there are no SR policies listed in the **SR Policies** window.

Figure 4: Example: Current Network



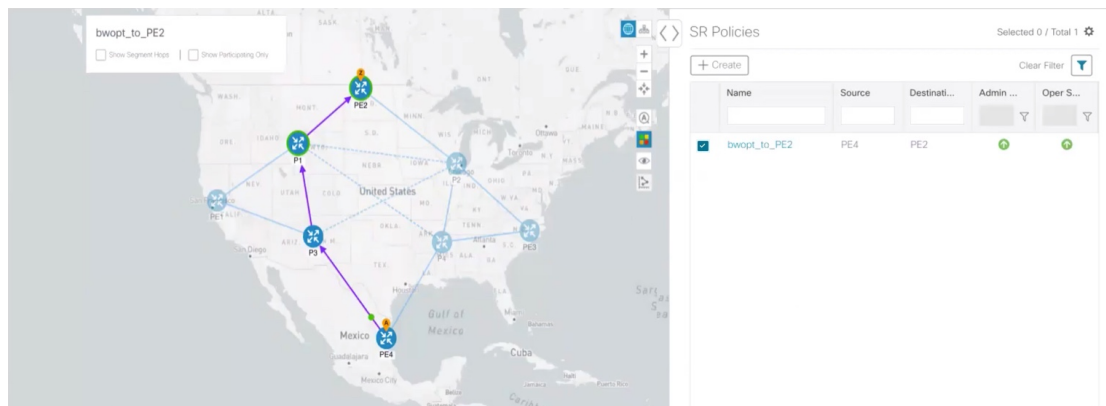
Suppose links between P3 and P4 devices go down. Traffic moves towards other links causing congestion and exceeds the utilization threshold set in Bandwidth Optimization.

**Figure 5: Example: Links Go Down Between P3 and P4**



Bandwidth Optimization recognizes the congestion and immediately calculates and deploys a tactical SR policy. This new tactical SR policy is listed in the **SR Policies** window.

**Figure 6: Example: Tactical SR Policy Deployed**



Cisco Crosswork Optimization Engine continually monitors the network. When the links between P3 and P4 are back up, Cisco Crosswork Optimization Engine will detect that the congestion (based on the criteria set in Bandwidth Optimization) has been mitigated. When the congestion falls under the set utilization threshold minus the utilization hold margin, the tactical SR policy is automatically removed from the network by Bandwidth Optimization.

## Troubleshoot BWOpt

BWOpt disables itself and issues an alarm when specific error conditions occur that hinder its ability to manage congestion properly and may lead to instability. The following table defines some of these conditions and possible causes to investigate. Additional details can be obtained for each error condition by referring to the BWOpt logs.

Table 4: Errors

Error Event Message	Possible Causes and Recommended Corrective Action
Optima Engine model error	<p>The network model used by BWOpt from the Optimization Engine is corrupt or is missing key data that is needed to properly support BWOpt. Possible causes include network discovery issues or synchronization problems between the Optimization Engine and Topology Services. Try restarting the Optimization Engine pod to rebuild the model.</p> <p>This error can also occur if the time required to deploy a tactical policy through SR-PCE, discover it, and add it to the model exceeds the <b>Deployment Timeout</b> option set for BWOpt. The default is 30 seconds which should suffice for small to medium sized networks. However, larger networks may require additional time.</p>
PCE Dispatch unreachable	<p>The deployment of a tactical policy to the network is not confirmed successful before the <b>Deployment Timeout</b> is exceeded. Increase the <b>Deployment Timeout</b> option to allow for additional time for deployments in larger networks.</p>
Unable to deploy a tactical SR policy	<p>A tactical SR policy deployment to SR-PCE was unsuccessful. There could be a variety of reasons for this. BWOpt and/or PCE Dispatch logs can provide some guidance as to the details of the failure. Confirm basic SR policy provisioning capability to the PCC via one of the SR-PCE providers is working.</p>