



自動化アプリケーション

ここでは、次の内容について説明します。

- [自動化アプリケーション](#) (1 ページ)
- [オンデマンド帯域幅の構成ワークフロー](#) (1 ページ)
- [オンデマンド帯域幅のシャットダウン](#) (7 ページ)
- [Bandwidth Optimizationアプリケーションワークフロー](#) (7 ページ)
- [帯域幅最適化のシャットダウン](#) (9 ページ)

自動化アプリケーション

自動化アプリケーションは、リアルタイムのネットワークモデルを使用することに依存しています。アプリケーションは、最新のネットワークモデル（マスターモデル）のコピーを取得し、アプリケーションの目的または機能に基づいて最適化を実行するか操作します。

その後、WAE Design ([**ファイル (File)**] > [**開く場所 (Open From)**] > [**WAEモデリングデーモン (WAE Modeling Daemon)**]) を使用して、ネットワークモデルを表示できます。

オンデマンド帯域幅の構成ワークフロー

オンデマンド帯域幅アプリケーションは、新しいサービスの影響をモデル化し、予測します。このアプリケーションは、永続的な帯域幅と特定の IGP または TE メトリック需要を必要とする新しいサービスをプロビジョニングするときに使用されます。アプリケーションは、ネットワークで委任されている SR ポリシーのパスを見つけます。オンデマンド帯域幅アプリケーションの詳細については、[オンデマンド帯域幅アプリケーション](#)を参照してください。

このワークフローでは、オンデマンド帯域幅およびその他のコンポーネントを構成するための構成手順の概要について説明します。



- (注) オンデマンド帯域幅を有効にする前に、Bandwidth Optimizationアプリケーションが実行されていないことを確認してください。両方のアプリケーションを同時に実行することはできません。

手順	詳細
1. デバイス認証グループと SNMP グループの構成	エキスパートモードを使用したデバイスアクセスの構成。
2. ネットワーク アクセス プロファイルの構成	ネットワーク アクセスの設定
3. XTC エージェントの構成	エキスパートモードを使用した XTC エージェントの構成
4. アグリゲータの構成	NIMO 収集の統合
5. WAE モデリングデーモン (WMD) の構成	WAE モデリングデーモン (WMD) の構成
6. XTC Agent to Patch (XATP) モジュールの構成	XTC Agent to Patch モジュールの構成
7. トポロジと追加の NIMO の実行	ネットワーク インターフェイス モジュール (NIMO)
8. オンデマンド帯域幅アプリケーションと SR ポリシーの構成	オンデマンド帯域幅の設定 (3 ページ)
9. ネットワークモデルを開く	<p>WAE Design を使用して、視覚的なネットワーク モデルレイアウトを取得できます。WAE Design から、[ファイル (File)] > [開く場所 (Open From)] > [WAEモデリングデーモン (WAE Modeling Daemon)] に移動し、最終ネットワークモデルを選択します。</p> <p>(注) 初期構成後、いつでも NIMO を実行でき、オンデマンド帯域幅アプリケーションはネットワークモデルを更新します。</p>

オンデマンド帯域幅の設定

始める前に

この手順では、オンデマンド帯域幅アプリケーションの構成オプションについて説明します。完全な構成ワークフローについては、[オンデマンド帯域幅の構成ワークフロー \(1 ページ\)](#) を参照してください。ワークフロー全体の例については、[初期オンデマンド帯域幅の CLI 構成例 \(4 ページ\)](#) を参照してください。



- (注) Bandwidth Optimization アプリケーションは、オンデマンド帯域幅アプリケーションと同時に実行できません。他のアプリケーションを使用するには、一方のアプリケーションを無効にする (`enable = false`) 必要があります。Bandwidth Optimization アプリケーションを適切にシャットダウンする方法については、[帯域幅最適化のシャットダウン \(9 ページ\)](#) を参照してください。

ステップ 1 エキスパートモードから、`/wae:wae/components/bw-on-demand:bw-on-demand` に移動し、`[config]` タブをクリックします。

ステップ 2 次の値を入力します。

- `[xtc-host]` : XTC ホストの IP アドレスを入力します。このエントリは、REST API を使用して XTC に接続する方法を決定します。これは永続的な接続であり、有効にすると、委任された PCEP 要求を受け取れるようになります。
- `[xtc-port]` : XTC ポートを入力します。
- `[keepalive]` : キープアライブインターバルを入力します。オンデマンド帯域幅アプリケーションと XTC は、永続的な接続を維持するためにキープアライブメッセージを交換します。接続が失敗した場合、オンデマンド帯域幅アプリケーションはシャットダウンし、現在の状態をクリアして再接続を試み、SR ポリシーを再委任します。
- `[priority]` : オンデマンド帯域幅アプリケーション インスタンスが複数ある場合は、このインスタンスの優先度を入力します。XTC は、優先度に応じてインスタンスを委任します。
- `[util-threshold]` : 輻輳制約を入力します (パーセンテージ)。オンデマンド帯域幅アプリケーションは、委任されているポリシーのパスを検索するときに、輻輳使用率のしきい値を超える可能性のあるパスを回避します。
- `[enable]` : オンデマンド帯域幅アプリケーションを有効にするには、`[true]` を選択します。

(注) `[advanced]` オプションについては、Cisco WAE の担当者にお問い合わせください。

ステップ 3 `[コミット (Commit)]` をクリックして、構成を保存します。

ステップ 4 デバイスの帯域幅と IGP または TE メトリックタイプを使用して、新しい SR ポリシーを構成します。特定のデバイス構成については、該当する Cisco IOS XR のドキュメントを参照してください (たとえば、[『Configure SR-TE Policies』](#))。デバイス構成の例：

```
segment-routing
  traffic-eng
    policy BWOD_2TO3_IGP
```

```

bandwidth 10000
color 100 end-point ipv4 198.51.100.3
candidate-paths
  preference 10
  dynamic mpls
    pce
      address ipv4 198.51.100.1
      exit
    metric
      type igp

```

ステップ 5 WAE Design ([WAE Design] > [ファイル (File)] > [開く場所 (Open From)] > [WAEモデリングデーモン (WAE Modeling Daemon)]) を使用して、得られたネットワークモデルを開きます。

初期オンデマンド帯域幅の CLI 構成例

以下は、Cisco Virtual Internet Routing Lab (VIRL) テスト環境内の初期オンデマンド帯域幅の CLI 構成の例です。初期構成後、いつでも NIMO を実行でき、オンデマンド帯域幅アプリケーションはネットワークモデルを更新します。

デバイスとネットワークの検出を構成します。

```

# config
# devices authgroups group vir1_test default-map
# devices authgroups group vir1_test default-map remote-name cisco
# devices authgroups group vir1_test default-map remote-password cisco
# devices authgroups group vir1_test default-map remote-secondary-password cisco
# devices authgroups snmp-group vir1_test default-map
# devices authgroups snmp-group vir1_test default-map community-name cisco
# wae nimos network-access network-access vir1_test default-auth-group vir1_test
# wae nimos network-access network-access vir1_test default-snmp-group vir1_test
# wae nimos network-access network-access vir1_test node-access 198.51.100.1 auth-group
  vir1_test
# wae nimos network-access network-access vir1_test node-access 198.51.100.1 snmp-group
  vir1_test
# wae nimos network-access network-access vir1_test node-access 198.51.100.1 ip-manage
  192.0.2.131
# wae nimos network-access network-access vir1_test node-access 198.51.100.2 auth-group
  vir1_test
# wae nimos network-access network-access vir1_test node-access 198.51.100.2 snmp-group
  vir1_test
# wae nimos network-access network-access vir1_test node-access 198.51.100.2 ip-manage
  192.0.2.132
# wae nimos network-access network-access vir1_test node-access 198.51.100.3 auth-group
  vir1_test
# wae nimos network-access network-access vir1_test node-access 198.51.100.3 snmp-group
  vir1_test
# wae nimos network-access network-access vir1_test node-access 198.51.100.3 ip-manage
  192.0.2.133
# wae nimos network-access network-access vir1_test node-access 198.51.100.4 auth-group
  vir1_test
# wae nimos network-access network-access vir1_test node-access 198.51.100.4 snmp-group
  vir1_test
# wae nimos network-access network-access vir1_test node-access 198.51.100.4 ip-manage
  192.0.2.134
# wae nimos network-access network-access vir1_test node-access 198.51.100.5 auth-group
  vir1_test
# wae nimos network-access network-access vir1_test node-access 198.51.100.5 snmp-group

```

```
virl_test
# wae nimos network-access network-access virl_test node-access 198.51.100.5 ip-manage
192.0.2.135
# wae nimos network-access network-access virl_test node-access 198.51.100.6 auth-group
virl_test
# wae nimos network-access network-access virl_test node-access 198.51.100.6 snmp-group
virl_test
# wae nimos network-access network-access virl_test node-access 198.51.100.6 ip-manage
192.0.2.136
# wae nimos network-access network-access virl_test node-access 198.51.100.7 auth-group
virl_test
# wae nimos network-access network-access virl_test node-access 198.51.100.7 snmp-group
virl_test
# wae nimos network-access network-access virl_test node-access 198.51.100.7 ip-manage
192.0.2.137
```

XTC エージェントを構成します。

```
# wae agents xtc xtc virl enabled xtc-host-ip 192.0.2.131
```

BGP ネットワーク (topo-bgpls-xtc-nimo) を構成します。

```
# networks network virl_bgpls nimo topo-bgpls-xtc-nimo xtc-host virl igp-protocol isis
extended-topology-discovery true backup-xtc-host virl network-access virl_test advanced
nodes remove-node-suffix virl.info
```

LSP PCEP ネットワーク (lsp-pcep-xtc-nimo) を構成します。

```
# networks network virl_pcep_lsp nimo lsp-pcep-xtc-nimo xtc-hosts virl xtc-host virl
# networks network virl_pcep_lsp nimo lsp-pcep-xtc-nimo source-network virl_bgpls advanced
sr-use-signaled-name true
```

アグリゲータが書き込むネットワークを設定します。

```
# networks network virl_final_model
```

継続的なポーリング (traffic-poll-nimo) を構成します。

```
# networks network virl_cp nimo traffic-poll-nimo network-access virl_max source-network
virl_dare iface-traffic-poller enabled
# networks network virl_cp nimo traffic-poll-nimo lsp-traffic-poller enabled
# networks network virl_cp nimo traffic-poll-nimo advanced snmp-traffic-population
scheduler-interval 0
```

送信元ネットワークにサブスクリブするようにアグリゲータを構成します。

```
# wae components aggregators aggregator virl_final_model sources source virl_bgpls
# wae components aggregators aggregator virl_final_model sources source virl_pcep_lsp
```

WMD を構成します。この例では、WMD は、WMD を使用するすべてのアプリケーションに対してデマンドメッシュとデマンド推論を実行するように設定されます。したがって、継続的なポーラーが WMD を更新すると、WMD はデマンド推論をトリガーします。

```
# wae components wmd config network-name virl_final_model dare dare-destination
virl_final_model
# wae components wmd config network-name virl_final_model demands add-demands true
demand-mesh-config dest-equals-source true
```

XATP モジュールを構成します。この例では、`connection-attempts` パラメータは 0 に設定されています。つまり、WMD 接続が失敗した場合、アプリケーションは接続が成功するまで WMD への接続を試行し続けます。

```
# wae components xatp config xtc-agent virl dare aggregator-network virl_dare
pcep-lsp-xtc-nimo-network virl_pcep_lsp topo-bgppls-xtc-nimo-network virl_bgppls
wae components xatp config wmd connection-attempts 0
```

WMD と XATP を有効にします。

```
# wae components wmd config enable true
# wae components xatp config enable true
# commit
# exit
```

NIMO（ネットワーク収集）を実行します。

```
networks network virl_bgppls nimo topo-bgppls-xtc-nimo run-xtc-collection
networks network virl_pcep_lsp nimo lsp-pcep-xtc-nimo run-collection
```

オンデマンド帯域幅を構成します。

```
# configure
# wae components bw-on-demand config xtc-host 192.0.2.131 xtc-port 8080 util-threshold
90.0
# wae components bw-on-demand config advanced lsp-traffic max-simulated-requested
primary-objective min-metric private-new-lsps true
# commit
# exit
```

WAE Design ([WAE Design] > [ファイル (File)] > [開く場所 (Open From)] > [WAEモデリングデーモン (WAE Modeling Daemon)]) を使用して基本的なネットワークモデルを開き、結果のネットワークモデルを比較します (SR ポリシーが構成され、オンデマンド帯域幅アプリケーションが実行された後)。

デバイスで SR ポリシーを構成します。

```
# configure
# segment-routing
# traffic-eng
# policy BWOD_2TO3_IGP
# bandwidth 1000
# color 100 end-point ipv4 192.0.2.132
# candidate-paths
# preference 10
# dynamic mpls
# pce
# address ipv4 192.0.2.130
# exit
# metric
# type igp
# commit
# end
```

SR ポリシー構成がコミットされると、WMD が更新され、オンデマンド帯域幅アプリケーションは、輻輳抑制と IGP メトリックを考慮してベストパスを計算します。WAE Design ([WAE Design] > [ファイル (File)] > [開く場所 (Open From)] > [WAEモデリングデーモン (WAE

Modeling Daemon) J) を使用して結果として得られるネットワークモデルを開き、ベースライン ネットワーク モデルを新しいネットワークモデルと比較します。

オンデマンド帯域幅のシャットダウン

オンデマンド帯域幅アプリケーションを適切にシャットダウンするには、次の手順を順番に実行する必要があります。

ステップ 1 オンデマンド帯域幅を停止します。

```
# wae components bw-on-demand config enable false
# commit
```

ステップ 2 XTC Agent to Patch モジュールを停止します。

```
# wae components xatp config enable false
# commit
```

ステップ 3 WAE モデリングデーモンを停止します。

```
# wae components wmd config enable false
# commit
```

ステップ 4 XTC エージェントを停止します。

```
# wae agents xtc xtc <network_name> disable xtc-host-ip <xtc_ip_address>
# commit
```

Bandwidth Optimization アプリケーション ワークフロー

Bandwidth Optimization アプリケーションは、ネットワークの状態の変化に対応してトラフィックを管理するように設計されています。ネットワーク状態の変化が輻輳を引き起こすかどうかを判断します。その場合、Bandwidth Optimization アプリケーションは LSP を計算し、展開のために XTC に送信します。

このワークフローでは、Bandwidth Optimization アプリケーションおよびその他のコンポーネントを構成するために必要な構成手順の概要について説明します。

手順	詳細
1. デバイス認証グループと SNMP グループの構成	エキスパートモードを使用したデバイスアクセスの構成。
2. ネットワーク アクセス プロファイルの構成	ネットワーク アクセスの設定
3. XTC エージェントの構成	エキスパートモードを使用した XTC エージェントの構成

手順	詳細
4. アグリゲータの構成	NIMO 収集の統合
5. WAE モデリングデーモン (WMD) の構成	WAE モデリングデーモン (WMD) の構成
6. XTC Agent to Patch (XATP) モジュールの構成	XTC Agent to Patch モジュールの構成
7. トポロジと追加の NIMO の実行	ネットワーク インターフェイス モジュール (NIMO)
8. Bandwidth Optimization アプリケーションの構成	Bandwidth Optimization の設定 (8 ページ)
9. ネットワークモデルを開く	<p>WAE Design を使用して、視覚的なネットワーク モデルレイアウトを取得できます。WAE Design から、[ファイル (File)] > [開く場所 (Open From)] > [WAEモデリングデーモン (WAE Modeling Daemon)] に移動し、最終ネットワークモデルを選択します。</p> <p>(注) 初期構成後、いつでも NIMO を実行でき、オンデマンド帯域幅アプリケーションはネットワークモデルを更新します。</p>

Bandwidth Optimization の設定

この手順では、Bandwidth Optimization アプリケーションの構成オプションについて説明します。完全な構成ワークフローについては、[Bandwidth Optimization アプリケーション ワークフロー \(7 ページ\)](#) を参照してください。

始める前に

Bandwidth Optimization アプリケーションは、オンデマンド帯域幅アプリケーションと同時に実行できません。他のアプリケーションを使用するには、一方のアプリケーションを無効にする (enable = false) 必要があります。オンデマンド帯域幅アプリケーションを適切にシャットダウンする方法については、[オンデマンド帯域幅のシャットダウン \(7 ページ\)](#) を参照してください。

ステップ 1 エキスパートモードから、`/wae:wae/components/bw-opt` に移動し、`[config]` タブをクリックします。

ステップ 2 次の値を入力します。

- `[enable]` : Bandwidth Optimization アプリケーションを有効にするには、`[true]` を選択します。

- [util-threshold] : 最適化を行う場合に超える必要があるパーセンテージを入力します。デフォルトは 100% です。
- [xtc-host] : XTC ホストのホスト名または IP アドレスを入力します。
- [xtc-port] : XTC ホストポートを入力します。
- [color] : XTC の SR ポリシーを表す色。詳細については、Cisco WAE の担当者にお問い合わせください。

ステップ3 [コミット (Commit)]をクリックして、構成を保存します。

ステップ4 ツールの実行後、[created-lsps] をクリックして、最適化されたルーティング用に作成された SR LSP を表示できます。

ステップ5 WAE Design ([WAE Design]>[ファイル (File)]>[開く場所 (Open From)]>[WAEモデリングデーモン (WAE Modeling Daemon)]) を使用して、得られたネットワークモデルを開きます。

例

CLI (構成モード) の例 :

```
# wae components bw-opt config color 2000 enable false threshold 90 xtc-host 192.0.2.131
xtc-port 8080
```

WAE SR ポリシーの制限事項

SR ポリシーに関連付けられた最新の IOS-XR SR 機能を使用する場合、次の WAE 制限事項が存在します。

- candidate-paths オプションで2つのパスが指定されている場合、最初のパスのみが考慮されます。
- SR LSP が WAE を介して作成される場合、デフォルトの色が SR LSP に設定されます。
- 複数の LSP が同じ色、送信元、および接続先を使用することはできません。

帯域幅最適化のシャットダウン

帯域幅最適化アプリケーションを適切にシャットダウンするには、次の手順を順番に実行する必要があります。

ステップ1 帯域幅最適化を停止します。

```
# wae components bw-opt config enable false
# commit
```

ステップ2 XTC Agent to Patch モジュールを停止します。

```
# wae components xatp config enable false
# commit
```

ステップ3 WAE モデリングデーモンを停止します。

```
# wae components wmd config enable false
# commit
```

ステップ4 XTC エージェントを停止します。

```
# wae agents xtc xtc <network_name> disable xtc-host-ip <xtc_ip_address>
# commit
```

翻訳について

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。