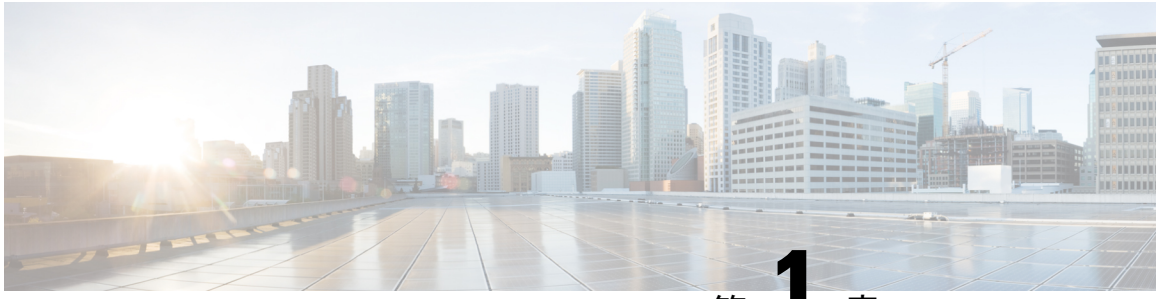
Cisco IOS XE Fuji 16.9.x QoS モジュラ QoS コマンドライン インターフェイス設定ガイド

シスコシステムズ合同会社
〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー
http://www.cisco.com/jp
お問い合わせ先：シスコ コンタクトセンター
0120-092-255 （フリーコール、携帯・PHS含む）
電話受付時間：平日 10:00～12:00、13:00～17:00
http://www.cisco.com/jp/go/contactcenter/

目 次



第 1 章

最初にお読みください

Cisco IOS XE 16 に関する重要な情報

現行の Cisco IOS XE Release 3.7.0E（Catalyst スイッチング用）および Cisco IOS XE Release 3.17S（アクセスおよびエッジルーティング用）の 2 つのリリースは、単一バージョンのコンバージドリリース Cisco IOS XE 16 に進化（マージ）しました。これにより、スイッチングおよびルーティング ポートフォリオにおける広範なアクセス製品およびエッジ製品を 1 つのリリースでカバーします。

機能情報

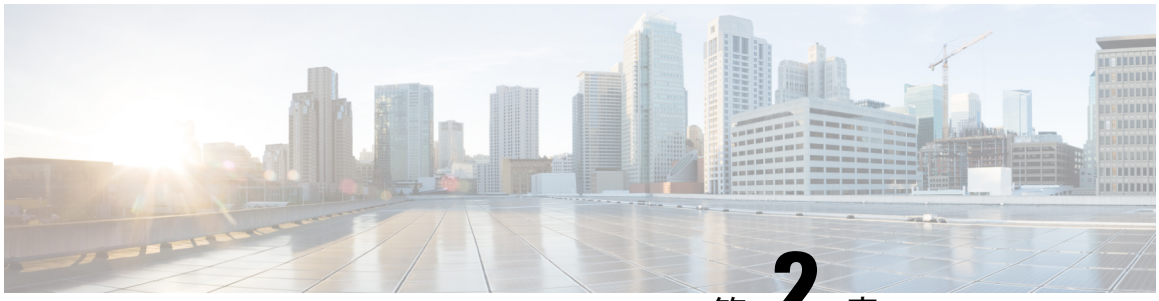
機能のサポート、プラットフォームのサポート、およびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco.com のアカウントは必要ありません。

参考資料

- 『Cisco IOS コマンド リファレンス』、すべてのリリース

マニュアルの入手方法およびテクニカル サポート

- シスコからタイムリーな関連情報を受け取るには、Cisco Profile Manager でサインアップしてください。
- 重要な技術によりビジネスに必要な影響を与えるには、シスコ サービスにアクセスしてください。
- サービス リクエストを送信するには、シスコ サポートにアクセスしてください。
- 安全で検証済みのエンタープライズクラスのアプリケーション、製品、ソリューション、およびサービスを探して参照するには、Cisco Marketplace にアクセスしてください。
- 一般的なネットワーキング、トレーニング、認定関連の出版物を入手するには、Cisco Press にアクセスしてください。
- 特定の製品または製品ファミリの保証情報を探すには、Cisco Warranty Finder にアクセスしてください。



第 2 章

MQC を使用した QoS 機能の適用

概要

この章では、モジュラ QoS CLI（MQC）の概要、Cisco ASR 1000 シリーズ アグリゲーション サービス ルータですべての QoS 機能を設定する方法を説明します。MQC は、シスコのルーティングおよびスイッチング プラットフォームで QoS を有効にするための標準化されたアプローチです。

この章では、あらゆる QoS 設定に必要な設定作業の概要について説明します。個々の機能については、適切なモジュールで説明します。

Cisco Modular QoS CLI

MQC では、QoS を有効にして検証するための 4 つの簡単なステップを実行します。各ステップで例が示されます。（機能の説明については各章を参照してください）

1. **クラスマップの作成**：トラフィック（アプリケーション）を使用するクラスに分類します。

```
class-map voice
  match dscp ef
class-map video
  match dscp AF41 AF42
```

2. **ポリシーマップの作成**：各クラスで必要とされる処理を定義します。

```
policy-map simple-example
  class voice
```

```
    priority
    police cir percent 10
  class video
    bandwidth remaining percent 30
```

3. **ポリシーマップの適用**：ポリシーを物理インターフェイスまたは論理インターフェイスにバインドし、ポリシーを動作させるトラフィックを特定します。そのインターフェイスを介してルータに入るトラフィック（入力）と、そのインターフェイスを介してルータから出るトラフィック（出力）のどちらにポリシーを適用するかを指定する必要があります。

```
interface gigabitethernet1/0/0
  service-policy out simple-example
```

4. **QoS ポリシーの動作の確認**：`show policy-map interface` コマンドを発行して、MQC で設定されたすべての QoS 機能の動作を確認します。

```
show policy-map interface gigabitethernet1/0/0
```

クラス マップの作成

クラスマップの作成時に、同様に処理される必要があるアプリケーションのグループを定義します。グループの名前を指定し、後に必要な処理を定義するときにその名前を使用します。

1 つまたは複数のフィルタ（分類ルール）を定義し、特定のパケット（アプリケーション）が指定したグループに属していることを確認します。クラスマップを作成するとき、そのグループの一部とみなされるには、パケットが 1 つのフィルタのみ (*match-any*) またはすべてのフィルタ (*match-all*) に一致させるかどうかを指定することができます。

次のようにクラスマップを作成します。

```
class-map [match-all|match-any] <traffic-class-name>
  match….    Filter1
  match…..   Filter2
```

次の例は、パケットが単一のフィルタにのみ一致する必要があるクラスを示しています。パケットが DSCP ef 値を持っている、**または** Cisco NBAR がそのパケットが skype アプリケーションを伝送していることを認識しているいずれかの場合に、パケットは音声クラスに属するものとみなされます。. ポリシーマップで音声という名前を使用して、このクラスに属すると分類されたパケットの処理方法を定義します。

```
class-map match-any voice
  match dscp ef
  match protocol skype
```

次の例では、match-all セマンティクスを使用しています。 パケットはクラスに属するためにすべてのフィルタと一致する必要があります。. トラフィックは MAPI として認識される必要があり（Cisco NBAR を使用）、アクセス リストで指定されたアドレスとの間で送受信される必要があります。

```
ip access-list extended mail-server-addr
  permit ip any host 10.10.10.1
```

```
  permit ip host 10.10.10.1 any
!
class-map match-all work-email
  match protocol mapi
  match access-group name mail-server-addr
```

前の例では、ASR 1000 シリーズプラットフォームのフィルター定義の柔軟性を示します。フィルタは、パケット ヘッダー（precedence、DSCP、Exp、またはCOS）内のマーク、アクセスリスト、 Cisco NBAR（match protocol*xxx*）または qos-group などの内部マーキングに基づいています。(利用可能な場合、サポートされているフィルタにおけるさらに完全な説明については、分類に関する章を参照してください)

便宜上、他の class-map を class-map にフィルタとして含めることもできます。:

```
class-map broadcast-video
  match dscp cs5
class-map multimedia-streaming
  match dscp af31 af32 af33
class-map multimedia-conferencing
  match dscp af41 af42 af43
class-map realtime-interactive
  match dscp cs4
!
class-map match-any all-video
  match class broadcast-video
  match class multimedia-streaming
  match class multimedia conferencing
  match class realtime-interactive
!
class-map match-any interactive-video
  match class multimedia conferencing
  match class realtime-interactive
```

この例では、*nested class-map* を クラス all-video と interactive-videoの定義に使用します。

定義上、特定のパケットは、クラスマップ内の複数のクラスの分類基準と一致する場合があります。その場合、ポリシーマップでクラスが定義されている順序によって、パケットが属するクラスが決まります。 パケットは、一致する最初のクラスに属します。

ポリシーマップの作成

ポリシーマップとは作成するトラフィックの各クラスに適用するアクションを指定する方法です。

上記の簡単な例を再度見てみましょう。

```
policy-map simple-example
  class voice
    priority
    police cir percent 10
  class video
    bandwidth remaining percent 30
```

ポリシーマップの名前は simple-example です。後でポリシーを 1 つ以上のインターフェイスに適用するときに使用されます。ポリシー自体は非常に読みやすくなっています。音声とビデオの2つのクラスのトラフィックを定義しました。音声トラフィックはスケジュールにおいて優先（低遅延）される必要がありますが、そのクラスのスループットはインターフェイス帯域幅の 10% に制限されています。ビデオ トラフィックの場合、専用のキューがあり、音声の処理後に残っているものの 30% を保証します。

上記のポリシーマップには、3 番目の暗黙的クラスがあります。class-default は、明示的に設定されているかどうかにかかわらず、ポリシーの最後のクラスです。これは、ユーザ定義クラスと一致しないすべてのトラフィックすべてを捕捉します。イーグレス ポリシーの class-default では、独自のキューがあり、暗黙の帯域幅余剰比率が 1 になります。帯域幅の値がパーセントで指定されている場合、class-default は未割り当てのパーセントを受け取ります (アスタリスクを参照)。これを踏まえ、上記のポリシーマップは実際には次のようになります。

```
class-map class-default
  match any
!
policy-map simple-example
  class voice
    priority
    police cir percent 10
  class video
    bandwidth remaining percent 30
  class class-default
    bandwidth remaining percent 70                ****
```

（注） class-default のクラスマップを作成する必要はありません。ここでは、ポリシーのしくみについての理解を深めるために、これを視覚化します。パケットが音声クラスまたはビデオクラスと一致しない場合は、常に class-default と一致させます。

上記のポリシーのアクションの例には **priority**、 **police**、および **bandwidth** コマンドが含まれています。アクションはコントロール ノブとして機能し、トラフィックのクラスがどのように扱われるかを区別します。

アクションを見るときの1つの非常に重要な差別化は、キューイングと非キューイングです。ここで、simple-example policy-map にもう 1 つのクラスを追加すると、次のようになります。

```
class-map youtube
  match protocol youtube
  !
policy-map simple-example
  class voice
    priority
    police cir percent 10
  class youtube
    police cir percent 5
  class video
    bandwidth remaining percent 30
```

YouTube には、リンク容量の 5% を超えることがないように YouTube トラフィックのレートを制限し、3 番目のユーザー定義クラスが追加されました。このクラスにはキューイング アク

ションが設定されていないため、キューは作成されません（キューを作成するアクションのリストについては下記を参照）。このクラスと一致するパケット（プロトコルが YouTube のクラス）は、ポリサーを通過した後、class-default キューにエンキューされます。

ポリシー定義でビデオクラスの前にYouTubeクラスを配置したことに気付きましたか。YouTube のトラフィックが、ビデオクラスではなく、常にこのクラスの一部であることを確認する必要があります。このクラスを policy-map を先に定義することで、ビデオ クラスの条件を確認する前に、このクラスとの一致をチェックします。

キューを作成する特定のアクションは、**priority**、**bandwidth**、**bandwidth remaining**、**shape**です。**fair-queue**、**queue-limit**、および**random-detect**などのその他のアクションは、キューを作成するアクションの 1 つが既に含まれているクラスでのみ使用できます。**policing** と **set** のアクションではキューは作成されませんが、キュー アドミッション コントロールの **police** コマンドを使用することができます。

キューイング アクションと非キューイング アクションを区別する主な理由の 1 つは、入力トラフィックに適用されるポリシーマップに ASR 1000 シリーズ ルータのキューイング アクションが含まれていない可能性があるからです。それでは、どのアクションがキューイングであり、どれが非キューイングであるかをまとめます。

キューイングおよび非キューイング アクション		アクション
キューイング アクション		
	スケジューリング	
		priority
		bandwidth
		bandwidth remaining
		shape
		fair-queue
	キュー管理/輻輳回避	
		queue-limit
		random-detect
非キューイング アクション		
	レート制限/アドミッション コントロール	
		police
	マーキング	
		set

階層型ポリシーマップは、別のポリシーマップのクラス内にポリシーマップを埋め込むことによって作成できます。

```
policy-map child
  class voice
    priority
    police cir percent 10
  class video
    bandwidth remaining percent 30
!
policy-map parent-vlan
  class class-default
    shape average 100m
    service-policy child
```

よく使用されるのは、「親ポリシーでのシェーピング/子ポリシー上のキューイング」ポリシーを作成することです。 このポリシーは、VLAN やトンネルなどの論理インターフェイスに接続できます。

分類の観点からすると、パケットは、特定の子クラスのメンバーと見なされるためには、親クラスと同様に子の分類基準にも準拠する必要があります。この例では、親クラスはclass-defaultであり、定義により、トラフィックはこのクラスと一致します。

階層型ポリシーを定義するときには、利便性を考慮して、ポリシーマップを再利用できます。

次の例では、parent-vlan100 と parentvlan200 の両方で「子」という名前のポリシーマップを使用します。インスタンス化（インターフェイスに接続）すると、 parent-vlan100 の音声クラスは 10 Mbps（100m の親シェーパーの 10％）に制限され、parent-vlan200 の音声クラスは 5 Mbps（50m の親シェーパーの 10％）に制限されます。

```
policy-map child
  class voice
    priority
    police cir percent 10
  class video
    bandwidth remaining percent 30
!
policy-map parent-vlan100
  class class-default
    shape average 100m
    service-policy child
!
policy-map parent-vlan200
  class class-default
    shape average 50m
    service-policy child
!
int gigabitethernet1/0/0.100
  service-policy out parent-vlan100
int gigabitethernet1/0/0.200
  service-policy out parent-vlan200
```

この例は、定義は共有される場合もありますが、異なるインターフェース上のポリシーのインスタンスは真に一意であることを示しています。

また、ユーザ定義クラスで使用されるポリシーマップを使用して階層型ポリシーを作成することもできます。

次の例は、ASR 1000 シリーズ ルータで現在サポートされている 3 レベルの階層型ポリシーを示しています。パケットがアプリケーション レベルでクラスと一致するには、子の音声またはビデオの分類子、vlan-sharing policy-map の vlan 分類子、および物理レベル policy-map の class-default（すべて）の要件 3 つを満たす必要があります。

```
class-map vlan100
  match vlan 100
class-map vlan200
  match vlan 200
!
policy-map child
  class voice
    priority
    police cir percent 10
  class video
    bandwidth remaining percent 30
!
policy-map vlan-sharing
  class vlan100
    shape average 100m
    service-policy child
  class vlan200
    shape average 50m
    service-policy child
!
policy-map physical-policy
  class class-default
    shape average 500m
    service-policy vlan-sharing
!
interface gigabitethernet1/0/0
  service-policy out physical-policy
```

ポリシーマップを作成すると、IOS はポリシーに対してエラーチェックを実行します。たとえば、制約のないプライオリティ キューイングを使用してポリシーを作成してから別のキューへの帯域幅を保証すると、IOS では切断が認識されます。 制約のない優先度キューがインターフェイスの帯域幅全体を占有する可能性がある場合、明らかにその帯域幅のいずれかを別のキューに保証することはできません。

```
policy-map create-error-example
  class unconstrained-priority
    priority
  class bandwidth-guarantee
    bandwidth percent 50
```

IOS で作成中にポリシーのエラーが検出された場合は、設定が拒否され、その時点でエラーが表示されます。

ポリシーマップの適用

Cisco MQC を使用する 3 番目の手順は、ポリシーマップをインスタンス化することです(つまり、ポリシーをインターフェイスに適用し、トラフィックの制御を開始します)。**service-policy**

コマンドを使用して、ポリシーを適用し、そのインターフェイスに入ってくるトラフィックとそのインターフェイスから出て行くトラフィックのどちらで動作するかを指定します。

```
interface gigabitethernet1/0/0
  service-policy out simple-example
```

キューイング ポリシーは出力トラフィック（**service-policy out** *policy-name*）でサポートされていますが、非キューイングアクションのみを含むポリシーは、入力トラフィック（**service-policy in***policy-name*）または出力トラフィックに適用される場合があります。

service-policy コマンドを適用するインターフェイスのことを接続点と呼びます。. この接続点は、（イーサネット インターフェイスや T1 インターフェイスなどの）物理インターフェイス、またはVLANサブインターフェイスやトンネルインターフェイスなどの論理インターフェイスである場合もあります。

ポリシーマップにキューイングアクションが含まれているが、階層ポリシーがない場合は、そのポリシーをフラット ポリシーと呼びます。フラット ポリシーは、物理インターフェイスにのみ適用できます。

キューイング ポリシーを論理インターフェイスに適用するには、子スタイルポリシー親ポリシーでの階層型シェーピング/子ポリシー上のキューイングを使用する必要があります。

先ほど学習したように、ポリシーを作成するとエラーチェックが行われます。2回目のエラーチェックは、ポリシーをインターフェイスに適用したときに発生します。たとえば、特定の種類のインターフェイスでは実現できない帯域幅の保証を含むポリシーを作成する可能性があります。ポリシーマップは、定義されている場合は有効である可能性がありますが、適用対象のインターフェイスに関する情報と組み合わせると、IOS では次の例にあるようなエラーが認識されます。

```
policy-map attach-error-example
  class bulk-data
    bandwidth 200000
```

このポリシーでは、バルクデータ用に 200 Mbps を予約する必要があることが規定されています。このポリシーを GigabitEthernet インターフェイスに適用した場合、正常に動作します。しかし、このポリシーを POS OC3 インターフェイスに適用すると、適用時に拒否されます。OC3 インターフェイスの公称帯域幅は 155 Mbps です。特定のトラフィック クラスに対して 200 Mbps を予約することはできません。

QoS ポリシーにおける検証

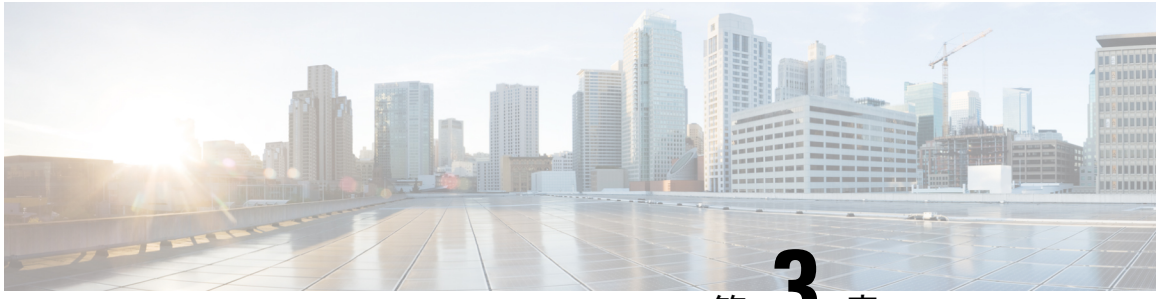
1 つのコマンドは常に、QoS ポリシーの動作を確認するために使用できます。

show policy-map interface *interface-name*

このコマンドの出力には、ポリシーマップの各クラスのセクションが表示されます。また、そのクラスに属していると分類されたパケットとバイト、およびそのクラスに設定されている各アクションの統計も表示されます。

（注） このコマンドで使用可能な統計情報は、CISCO-CLASSBASED-QOS-MIB で SNMP を介して使用することもできます。

QoS ポリシーが DMVPN などのマルチポイント インターフェイスに適用されている場合は、コマンドの **show policy-map multipoint tunnel** トンネル番号バリアントを使用します。同様に、ポリシーがブロードバンド セッションに適用されている場合は、コマンドの **show policy-map session uid** セッション番号バリアントを使用します。



第 3 章

3 レベルのユーザ定義キューイング ポリシーのサポート

3 レベルのユーザ定義キューイング ポリシーのサポート機能を使用して、最上位レイヤのユーザ定義クラスが含まれる 3 レベル ポリシーで、階層内のトラフィック クラスの柔軟性をサポートし、強化することができます。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「Bug Search Tool」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

3 レベルのユーザ定義キューイング ポリシーのサポートに関する制約事項

- 3 レベルの階層型キューイング ポリシーの最上位レイヤのユーザ定義クラスは、ポートチャネルのメイン インターフェイスではサポートされません。

  最上位レイヤのユーザ定義クラスは、論理ターゲットではサポートされません。論理ターゲットには、サービス グループ、トンネル、セッション、ディーラー インターフェイスなどがあります。

3 レベルのユーザ定義キューイング ポリシーのサポートに関する情報

階層型 QoS の 3 つのパラメータによるスケジューラ

従来の IOS は、最大値（shape）と最小（bandwidth）を使用して、トラフィックの輻輳が発生した場合の各スケジューラ ノードの動作を定義します。つまり、2つのパラメータによるスケジューラです。

ASR 1000 は、3 つのパラメータ、最大値（shape）、最小値（bandwidth）、および超過帯域幅の共有を調整する超過値（bandwidth remaining）による別のスケジューラです。2 つのパラメータによるスケジューラでは、超過帯域幅は比例的に（各クラスの帯域幅率と同じに）クラス間で共有されます。しかし、3 つのパラメータによるスケジューラでは、最小帯域幅の要件が満たされると、デフォルトによって超過帯域幅は等分に共有されますが、これは、「bandwidth remaining」コマンドを使用すると調整できます。ISR 4000 プラットフォームは、同じ設計を共有しています。

従来の IOS では、階層のリーフと中間ノードで帯域幅を設定することができます。IOS XE では、階層のリーフ ノードでのみ、bandwidth（bandwidth rate または bandwidth percent）が許可されます。つまり、bandwidth（bandwidth rate または bandwidth percent）クラスは、キューイング機能を含んでいる子ポリシーマップに対応付けることはできません。これはソフトウェアでの制約であるため、将来、解除される可能性があります。

従来の IOS QoS ポリシー マップを IOS XE プラットフォームに移行する現在の導入の場合、中間ノードの bandwidth コマンドを bandwidth remaining コマンドに変換することが最良の選択肢です。bandwidth remaining percent コマンドまたは bandwidth remaining ratio コマンドを使用すると、非常によく似た動作を実現できます。

階層型ポリシーのガイドライン

一般に、階層の 3 つのレベルが ASR 1000 でサポートされています。階層型ポリシーは QoS をサポートするほとんどの物理ターゲットと論理ターゲットに適用できます。

キューイング ポリシーと非キューイング ポリシーが階層に混在している場合、非キューイング ポリシーマップは、そのポリシーマップのリーフ レベル（親の親および親キューイング ポリシー下の子ポリシーなど）にする必要があります。

ポリシーマップを（トンネルまたはセッションなどの）仮想インターフェイスに適用すると、構成によっては階層を 2 つのキューイングの レベルに制限する追加制約がある場合があります。

- キューイング機能：shape、bandwidth、bandwidth remaining、random-detect、fair-queue、queue limit、および priority。
- 非キューイング機能：police、mark、および account。

階層機能の組み合わせ	入力ポリシー サポート	出力ポリシー サポート
1 レベルの非キューイング ポリシー	対応	対応
2 レベルの非キューイング ポリシー（カラー対応ポリシングを含む）	対応	対応
3 レベルの非キューイング ポリシー（階層型カラー対応ポリシングを含む）	対応	対応
1 レベルのキューイング ポリシー	-	対応
2 レベルのキューイング ポリシー	-	対応
3 レベルのキューイング ポリシー	-	対応
2 レベルの混合ポリシー、親レベルでのキューイング機能	-	対応
3 レベルの混在ポリシー、親の親レベルまたは親の親 + 親レベルでのキューイング機能	-	対応

HQoS の最上位レベル ポリシーのユーザ定義トラフィック クラス

「クラス マップ」によって明示的に設定されたどのトラフィック クラスも「ユーザ定義クラス」と呼ばれます。class-default クラスは設定する必要はなく、これらのクラスをポリシー内に使用して、ユーザ定義クラスに属していないすべてのトラフィックに一致させることができます。

Polaris 16.3 リリースまでは、3 レベルのキューイング ポリシーで最上位レベルに設定できるのは class-default クラスのみです。Polaris 16.3 以降は、3 レベルの階層型ポリシーの最上位レベルでユーザ定義クラスがサポートされています。

3 レベルのユーザ定義のキューイング ポリシー サポートの設定方法

3 レベルの階層型 QoS ポリシーの設定

```
enable
configure terminal
class-map vlan10
 match vlan10
class-map vlan20
 match vlan 20
class-map ef
 match dscp ef
policy-map child
 class ef
  priority
  police 1000000
 class class-default
 police 3000000
policy-map parent
 class vlan10
  shape average 4000000
  service-policy child
 class vlan20
  shape average 8000000
  service-policy child
policy-map grand-parent
 class class-default
  shape average 10000000
  service-policy parent
end
```

最上位レベルのポリシーでのユーザ定義のトラフィッククラスの設定

```
ip access-list extended PEER
permit ip host 200.0.0.2 any

class-map match-all ef
 match dscp ef
class-map match-all vlan100
```

```
 match vlan 100
class-map match-all vlan101
 match vlan 101
class-map match-all PEER
 match access-group name PEER

policy-map child
 class ef
  bandwidth remaining percent 15
 class class-default
  fair-queue
  queue-limit 512 packets
  bandwidth remaining percent 85

policy-map parent
 class PEER
  shape average 8000000
  bandwidth remaining percent 10
  service-policy child
class class-default
 shape average 8000000

policy-map grandparent
 class vlan100
  shape average 8000000
  bandwidth remaining ratio 1000
  service-policy parent
 class vlan101
  shape average 8000000
  bandwidth remaining ratio 1000
  service-policy parent
 class class-default
  bandwidth remaining ratio 1
  shape average 10000000
end
```

3 レベルのユーザ定義のキューイング ポリシー サポートに関するその他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『Cisco IOS Master Commands List, All Releases』

シスコのテクニカル サポート

説明	リンク
シスコのサポート Web サイトでは、シスコの製品やテクノロジーに関するトラブルシューティングにお役立ていただけるように、マニュアルやツールをはじめとする豊富なオンライン リソースを提供しています。 お使いの製品のセキュリティ情報や技術情報を入手するために、Cisco Notification Service（Field Notice からアクセス）、Cisco Technical Services Newsletter、Really Simple Syndication（RSS）フィードなどの各種サービスに加入できます。 シスコのサポート Web サイトのツールにアクセスする際は、Cisco.com のユーザ ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

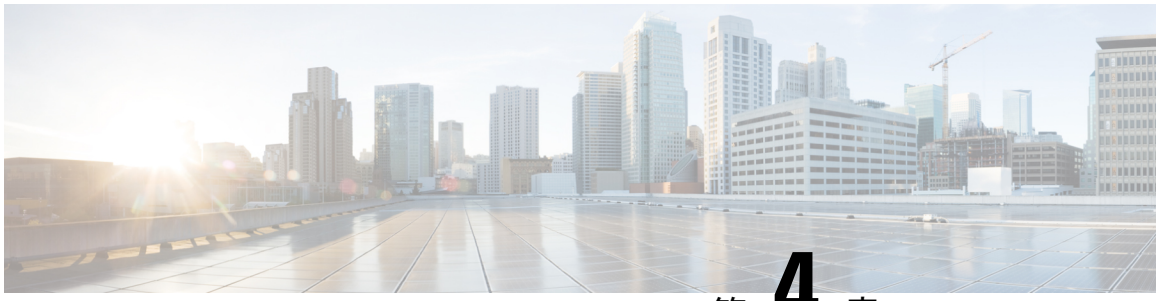
3 レベルのユーザ定義キューイング ポリシーのサポートの機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *1: 3* レベルのユーザ定義キューイング ポリシーのサポートの機能情報

機能名	リリース	機能情報
3 レベルのユーザ定義キューイング ポリシーのサポート	Cisco IOS XE Denali 16.3.1.	この機能は、Cisco ASR 1000、ISR4000、CSR1000v のプラットフォームに導入されました。ユーザ定義クラスは、3 レベル階層型ポリシーの最上位レイヤで設定できます。



第 4 章

IP/ATM 間サービス クラスの設定

ここでは、IP/ATM 間サービス クラス（CoS）の設定作業について説明します。IP-ATM CoS とは、IP と ATM の間に QoS 特性をマップする機能セットです。

プラットフォームのサポートおよびソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator を使用すると、Cisco IOS および Catalyst OS ソフトウェア イメージがサポートする特定のソフトウェア リリース、フィーチャ セット、またはプラットフォームを確認できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn からアクセスします。Cisco.com のアカウントは必要ありません。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「Bug Search Tool」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

単一の ATM VC での IP/ATM 間 CoS の設定タスク リスト

単一の ATM 仮想回線（VC）に IP/ATM 間 CoS を設定するには、次の項で説明する作業を実行します。最初の 2 つの項の作業は必須です。残りの項の作業は任意です。

IP/ATM 間 CoS 機能には、ATM 相手先固定接続（PVC）の管理が必要です。

WRED パラメータ グループの定義

コマンド	目的
`Router(config)#` **`random-detect-group`** *`group-name`*	WREDまたはVIP分散WRED（DWRED）グループパラメータを定義します。

WRED パラメータ グループの設定

手順の概要

1. Device(config)# **random-detect-group** *group-name*
2. Device(config)# **exponential-weighting-constant** *exponent*
3. Device(config)# **precedence** *precedence min-threshold max-threshold mark-probability-denominator*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	Device(config)# **random-detect-group** *group-name*	WRED または DWRED のパラメータ グループを指定します。
ステップ 2	Device(config)# **exponential-weighting-constant** *exponent*	WREDまたはDWREDの指定したパラメータ グループの平均キューサイズ計算のための指数加重係数を設定します。 または
ステップ 3	Device(config)# **precedence** *precedence min-threshold max-threshold mark-probability-denominator*	特定の IP precedence の指定された WRED または DWRED パラメータ グループを設定します。

WRED パラメータの表示

コマンド	目的
`Router#` **`show queueing random-detect`** [**`interface`** *`atm_subinterface`* [**`vc`** [[*`vpi`*/] *`vci`*]]]	指定された ATM サブインターフェイスで、WREDまたはDWREDがイネーブルであるすべてのVCのパラメータを表示します。

キューイング統計情報の表示

コマンド	目的
`Router# show queueing interface interface-number [vc [[vpi/] vci]]`	インターフェイスの特定の VC のキューイングの統計情報を表示します。

ATM バンドルでの IP/ATM 間 CoS の設定タスク リスト

ATM バンドルで IP/ATM 間 CoS を設定するには、次の項に示されているタスクを実行します。

IP/ATM 間 CoS 機能には、ATM PVC の管理が必要です。

VC バンドルの作成

コマンド	目的
`Router(config-subif)# bundle bundle-name`	指定されたバンドルを作成し、バンドル コンフィギュレーション モードを開始します。

バンドルレベルのパラメータの適用

バンドル レベルのパラメータの設定

コマンド	目的
`Device(config-atm-bundle)# protocol protocol {protocol-address | inarp} [[no] broadcast]`	スタティック マップを設定するか、バンドルの Inverse Address Resolution Protocol（Inverse ARP）または Inverse ARP ブロードキャストをイネーブルにします。 （注） バンドルレベルのパラメータは、VC クラスを割り当てるか、またはバンドルにそれらのクラスを直接適用することによって、適用できます。バンドルに割り当てられた VC クラスを使用して適用されたパラメータは、バンドル レベルで適用されたパラメータに置き換えられます。バンドルレベルのパラメータは、個々の VC に適用されるパラメータで置換されます。
`Device(config-atm-bundle)# encapsulation aal-encap`	ATM アダプテーション層（AAL）およびそのバンドルのカプセル化のタイプを設定します。

コマンド	目的
`Device(config-atm-bundle)# inarp minutes`	すべての VC バンドル メンバの Inverse ARP 期間を設定します。
`Device(config-atm-bundle)# broadcast`	すべての VC バンドル メンバのブロードキャスト転送をイネーブルにします。
`Device(config-atm-bundle)# oam retry up-count down-count retry frequency`	運用管理および保守（OAM）の管理に関連する VC バンドル パラメータを設定します。
`Device(config-atm-bundle)# oam-bundle [manage] [frequency]`	エンドツーエンドの F5 OAM ループバック セルの生成と、バンドルのすべての VC の OAM 管理をイネーブルにします。

バンドルに適用する VC クラスのパラメータの設定

コマンド	目的
`Router(config-vc-class)# oam-bundle [manage] [frequency]`	エンドツーエンドの F5 OAM ループバック セルの生成と、バンドルのすべての VC の OAM 管理をイネーブルにします。 （注） VC クラスを使用すると、クラス自体をバンドルに適用できるため、一度に複数の属性を適用するようにバンドルを設定できます。クラスを使用すると、すべての VC でパラメータを一般化することができます。一部のパラメータについては、その後で個々の VC のパラメータを変更できます（詳細については、「個々の VC へのパラメータの適用」の項を参照してください）。

バンドルへのクラスの適用

コマンド	目的
(config-atm-bundle)# **class-bundle** *vc-class-name*	指定された VC クラスに含まれるバンドル レベルのコマンドでバンドルを設定します。 （注） VC クラスに含まれるバンドル レベル コマンドを使用して設定されたパラメータは、バンドルとそのすべての VC メンバに適用されます。バンドル上に直接設定されたコマンドを通じて適用されたバンドルレベルのパラメータは、VC クラスを通じて適用されたパラメータよりも優先されます。VC クラスを通じて適用されたバンドルレベルのパラメータやバンドルに直接適用されたバンドルレベルのパラメータの一部は、バンドル VC コンフィギュレーション モードで個々の VC に直接適用されたコマンドによって置き換えることができます。

バンドルへの VC のコミット

コマンド	目的
Device(config-atm-bundle)# **pvc-bundle** *pvc-name* [*vpi*/] [*vci*]	指定された VC をバンドルに追加し、指定された VC バンドル メンバを設定するためにバンドル VC コンフィギュレーション モードを開始します。

個々の VC へのパラメータの適用

VC バンドル メンバーの直接設定

コマンド	目的
Device(config-if-atm-member)# **ubr** *output-pcr* [*input-pcr*]	未指定ビット レート（UBR）の QoS の VC を設定し、出力のピーク セル レート（PCR）を指定します。
Device(config-if-atm-member)# **ubr+** *output-pcr output-mcr* [*input-pcr*] [*input-mcr*]	UBR の QoS の VC を設定し、出力の PCR と出力の最低保証セル レートを指定します。
Device(config-if-atm-member)# **vbr-nrt** *output-pcr output-scr output-mbs* [*input-pcr*] [*input-scr*] [*input-mbs*]	可変ビット レート非リアルタイム（VBR-NRT）の VC を設定し、出力の PCR、出力の平均セル レート、および出力の最大バースト セル サイズを指定します。

コマンド	目的
`Device(config-if-atm-member)# precedence [other | range]`	VC の Precedence レベルを設定します。
`Device(config-if-atm-member)# bump {implicit | explicit precedence-level | traffic}`	VC のバンピング ルールを設定します。
`Device(config-if-atm-member)# protect {group | vc}`	VC がバンドルの保護されたグループに所属するか、個別に保護された VC バンドル メンバになるように設定します。

VC バンドル メンバーに適用する VC クラスのパラメータの設定

コマンド	目的
`Device(config-vc-class)# bump {implicit | explicit precedence-level | traffic}`	クラスを適用する VC メンバのバンピング ルールを指定します。これらのルールは、キャリアのVCバンドルメンバがダウンした場合に、バンドルトラフィックのVCを送信するルールを決定します。 （注） また、**ubr** コマンド、**ubr+** コマンド、**vbr-nrt** コマンドを VC クラスに追加すると、VC バンドル メンバーの設定に使用できます。VC が VC バンドルのメンバーである場合、VC クラス モードでの VC の設定に **encapsulation** コマンド、**protocol** コマンド、**inarp** コマンド、および **broadcast** コマンドを使用することはできません。これらのコマンドは、バンドル メンバ レベルではなくバンドル レベルでのみ使用できます。個々の VC に対する設定は、VC クラスをバンドルに適用することによってすべての VC バンドル メンバに適用された共有の設定を上書きします。
`Device(config-vc-class)# precedence precedence min-threshold max-threshold mark-probability-denominator`	クラスを適用する VC メンバの Precedence レベルを定義します。
`Device(config-vc-class)# protect {group | vc}`	VC をバンドルの保護されたグループのメンバとして設定するか、個別に保護された VC として設定します。

個々の VC バンドル メンバーへの VC クラスの適用

コマンド	目的
`Device(config-if-atm-member)# class-vc vc-class -name`	VC クラスを VC バンドル メンバーに割り当てます。

バンピングされたトラフィックを拒否する場合の VC の設定

コマンド	目的
`Device(config-if-atm-member)# no bump traffic`	リダイレクトされる可能性のあるバンピングされたトラフィックを受け入れないように VC を設定します。

VC バンドルとそれらの VC メンバーのモニタリングと保守

コマンド	目的
`Device# show atm bundle bundle-name`	各バンドルの VC メンバに割り当てられたバンドルの属性と、VC メンバの現在の稼働ステータスを表示します。
`Device# show atm bundle bundle-name statistics [detail]`	指定されたバンドルの統計情報または詳細な統計情報を表示します。
`Device# show atm map`	ATM ネットワークおよび ATM バンドル マップのリモート ホストに対するすべての設定済みの ATM スタティック マップのリストを表示します。
`Device# debug atm bundle errors`	バンドルのエラーに関する情報を表示します。
`Device# debug atm bundle events`	バンドルのイベントの記録を表示します。

VC 単位の WFQ および CBWFQ の設定タスク リスト

VC 単位の WFQ および CBWFQ に IP/ATM 間 CoS を設定するには、次の項で説明するタスクを実行します。

IP/ATM 間 CoS 機能には、ATM PVC の管理が必要です。

クラスベースの重み付け均等化キューイングの設定

VC の CBWFQ を設定する前に、標準の CBWFQ コマンドを使用して次の作業を実行する必要があります。

- VC 間で送信されるトラフィックの分類に使用される 1 つまたは複数のクラスを作成する
- サービス ポリシーとして使用するクラスを含むポリシーマップを定義する

（注） クラス ポリシーはルータで定義可能な数だけ設定できます（最大 64 個）。ただし、VC に適用されるポリシーマップに含まれ、すべてのクラスに割り当てられている帯域幅の合計は、VC の使用可能な帯域幅の 75% を超えてはいけません。使用可能な帯域幅の残りの 25 % は、ATM セルのオーバーヘッド（ATM セル タックスとも呼ばれる）、ルーティングとベストエフォート型のトラフィック、およびオーバーヘッドが想定されるその他の機能などのカプセル化に使用されます。帯域幅割り当ての詳細については、「輻輳管理の概要」モジュールを参照してください。

CBWFQ では最小帯域幅のみを保証するため、使用可能ビット レート（ABR）および可変ビット レート（VBR）のサービス クラスを持つ VC のみに CBWQF を適用できます。UBR および未指定ビット レート プラス（UBR+）には VC 単位の WFQ および CBWFQ を適用できません。これは、両方のサービス クラスはどちらもベストエフォート型のクラスであり、最小帯域幅が保証されないためです。VC の CBWFQ がイネーブルの場合、サービス ポリシーの一部として設定されているすべてのクラスは、均等化キューイング システムに組み込まれます。

VC レベルで CBWFQ を設定するほかに、IP to ATM CoS 機能を使用すると、VC レベルでフローベースの WFQ を設定できます。フローベースの WFQ はベストエフォート型のサービス クラスを提供する（つまり、最小帯域幅が保証されない）ため、すべてのタイプの CoS VC（ABR、VBR、UBR、および UBR+）に VC 単位の WFQ を設定できます。

VC 単位の WQF では class-default クラスを使用します。そのため、VC 単位の WFQ を設定するには、最初にポリシーマップを作成してから class-default クラスを設定する必要があります（事前に設定されているため、class-default クラスを作成する必要はありませんが、設定する必要はあります）。VC 単位の WFQ の場合、class-default クラスは **fair-queue** ポリシーマップ クラス コンフィギュレーション コマンドを使用して設定する必要があります。

fair-queue ポリシーマップ クラス コンフィギュレーション コマンドの設定に加えて、**queue-limit** コマンドまたは **random-detect** コマンドの両方ではなく、どちらか一方を使用して、デフォルト クラスを設定できます。さらに、デフォルト クラスでフローベースの WFQ を使用する場合は、**bandwidth** ポリシー マップ クラス コンフィギュレーション コマンドを使用してデフォルト クラスを設定することはできません。このコマンドを使用すると、デフォルト クラスがフローベースの WFQ として見なされないため、そのクラスを含むサービス ポリシーの適用が ABR および VBR の VC に制限されます。

サービス ポリシーの適用と VC に対する CBWFQ のイネーブル化

スタンドアロン VC へのポリシーマップの適用と CBWFQ のイネーブル化

コマンド	目的
`Router(config-if-atm-vc)#` **service-policy output** *policy-map*	CBWFQ をイネーブル化し、指定されたサービス ポリシーマップを作成または変更されている VC に適用します。

個々の VC へのポリシーマップの適用と CBWFQ のイネーブル化

コマンド	目的
`Router(config-if-atm-member)#` **service-policy output** *policy-map*	CBWFQ をイネーブル化し、指定されたサービス ポリシーマップを作成または変更されている VC に適用します。

（注） **service-policy output** コマンドと **random-detect-group** コマンドは同時に使用できません。サービス ポリシーを適用することによって CBWFQ をイネーブルにした VC に WRED グループを適用することはできません。さらに、1 つのコマンドが設定されている場合に、他のコマンドを設定できるようにするには、他のコマンドをディセーブルにする必要があります。

フローベースの WFQ を使用する場合の VC の設定

手順の概要

1. Device(config)# **policy-map** *policy-map*
2. Device(config-pmap)# **class class-default** *default-class-name*
3. Device(config-pmap-c)# **fair-queue** *number-of-dynamic-queues*
4. 次のいずれかを実行します。
   - Device(config-pmap-c)# **queue-limit** *number-of-packets*
   - Device(config-pmap-c)# **random-detect**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	Device(config)# **policy-map** *policy-map*	作成または変更するポリシーマップの名前を指定します。

	コマンドまたはアクション	目的
ステップ 2	Device(config-pmap)# **class class-default** *default-class-name*	ポリシーを設定または変更できるようデフォルト クラスを指定します。 （注） フローベースの WFQ クラスを含むポリシーマップと同じポリシー マップに他のクラスを含めることができます。他の方法では一致しなかったパケットが、デフォルトの class-default クラスの一致基準によって選択されます。
ステップ 3	Device(config-pmap-c)# **fair-queue** *number-of-dynamic-queues*	デフォルト クラスで実行するフローベースの WFQ が使用するために予約する、ダイナミック キューの数を指定します。 （注） デフォルトでは（つまり、**fair-queue** ポリシーマップ クラス コンフィギュレーション コマンドを使用して class-default クラスを設定しておらず、**bandwidth** ポリシーマップ クラス コンフィギュレーション コマンドを使用して class-default クラスを設定していない場合でも）、デフォルト クラスはフローベースの WFQ として定義されます。
ステップ 4	次のいずれかを実行します。 • Device(config-pmap-c)# **queue-limit** *number-of-packets* • Device(config-pmap-c)# **random-detect**	そのクラスのキューに投入できるパケットの最大数を指定します。 WRED をイネーブルにします。クラス ポリシーは、テール ドロップではなく WRED を使用してパケットをドロップします。

スタンドアロン VC へのポリシーマップの適用と WFQ のイネーブル化

コマンド	目的
Device(config-if-atm-vc)# **service-policy output** *policy-map*	作成または変更される VC に class-default クラスを含む指定されたポリシーマップを適用することで、VC の WFQ をイネーブルにします。

個々の VC へのポリシーマップの適用と WFQ のイネーブル化

コマンド	目的
`Device(config-if-atm-member)# service-policy output policy-map`	class-default クラスを含む指定されたポリシーマップを VC バンドル メンバにアタッチすることで、VC バンドル メンバの WFQ をイネーブルにします。

VC 単位の WFQ および CBWFQ のモニタリング

コマンド	目的
`Device# show policy-map interface interface-number [vc [vpi/] vci]`	特定のインターフェイスまたは VC のキュー内部のパケットのコンテンツを表示します。

コンソールへのエラー メッセージのロギングのイネーブル化

コマンド	目的
`Router(config)# logging console level`	コンソールに記録されるメッセージを重大度に基づいて制限します。

IP/TM 間 CoS の設定例

例：WRED グループおよび IP プレシデンスを使用した単一の ATM VC

次に、ATM インターフェイスに PVC を作成し、その PVC に sanjose という WRED パラメータ グループを適用します。次に、WRED パラメータ グループ sanjose の IP Precedence の値を設定します。

```
interface ATM1/1/0.46 multipoint
 ip address 200.126.186.2 255.255.255.0
 no ip mroute-cache
 shutdown
pvc 46
 encapsulation aal5nlpid
 random-detect attach sanjose
!
random-detect-group sanjose
 precedence 0  200   1000 10
 precedence 1  300   1000 10
 precedence 2  400   1000 10
 precedence 3  500   1000 10
 precedence 4  600   1000 10
 precedence 5  700   1000 10
```

```
 precedence 6  800   1000 10
 precedence 7  900   1000 10
```

例：VC クラスを使用した VC バンドルの設定

この例では、Intermediate System-to-Intermediate System（IS-IS）を IP ルーティング プロトコルとして使用するルータに VC バンドル管理を設定します。

Bundle-Class クラス

最初に、この設定では、VCパラメータのコンフィギュレーションコマンドを含むbundle-classという名前のVCクラスを定義します。bundle-classクラスがバンドル レベルで適用されると、そのバンドルに属しているすべての VC にこれらのパラメータが適用されます。バンドル VC モードのバンドルの個々のVCに直接適用されたコマンドは、バンドル レベルでグローバルに適用されたコマンドよりも優先されます。階層の優先ルールを考慮すると、クラスbundle-classが適用されるバンドルに属する VC は、aal5snap カプセル化、ブロードキャスト オン、Inverse ARP を使用したIPアドレスの解決、OAMのイネーブル化などのパラメータを特性として持ちます。

```
router isis
 net 49.0000.0000.0000.1111.00
vc-class atm bundle-class
 encapsulation aal5snap
 broadcast
 protocol ip inarp
 oam-bundle manage 3
 oam retry 4 3 10
```

次の設定のセクションで、そのVCにクラスを割り当てることによって、バンドル内の個々の VC に適用可能なパラメータを指定するコマンドを含んだ VC クラスを定義します。

Control-Class クラス

control-class という名前のクラスを VC に適用すると、VC は IP プレシデンス レベルが 7 のトラフィックを伝送します。このクラスを割り当てたVCがダウンした場合はバンドルも一緒にダウンします。これは、このクラスが VC を保護された VC にするためです。このクラスを使用する VC の QoS タイプは vbr-nrt です。

```
vc-class atm control-class
 precedence 7
 protect vc
 vbr-nrt 10000 5000 32
```

Premium-Class クラス

premium-class という名前のクラスを VC に適用すると、その VC は IP プレシデンス レベルが 6 と 5 のトラフィックを伝送します。この VC では他のトラフィックのバンピングを許可しません。このクラスが適用された VC がダウンした場合、バンピングされたトラフィックは IP プレシデンス レベルが 7 の VC にリダイレクトされます。このクラスは、VC をバンドルの保

護されたグループのメンバにします。保護されたグループのすべてのメンバがダウンした場合は、バンドルもダウンします。このクラスを使用する VC の QoS タイプは vbr-nrt です。

```
vc-class atm premium-class
 precedence 6-5
 no bump traffic
 protect group
 bump explicitly 7
 vbr-nrt 20000 10000 32
```

Priority-Class クラス

priority-class という名前のクラスを VC に適用すると、VC は IP Precedence の範囲が 4 ～2 のトラフィックを伝送するように設定されます。この VC は明示的なバンピング ルールを使用し、トラフィックをバンプできるようにします。この VC はバンドルの保護されたグループに属しています。このクラスを使用する VC の QoS タイプは ubr+ です。

```
vc-class atm priority-class
 precedence 4-2
 protect group
 ubr+ 10000 3000
```

Basic-Class クラス

basic-class という名前のクラスを VC に適用すると、その VC は **precedence other** コマンドを使用して設定され、プロファイルに指定されていない IP プレシデンス レベルを持つトラフィックを伝送します。このクラスを使用する VC は、バンドルの保護されたグループに所属します。このクラスを使用する VC の QoS タイプは ubr です。

```
vc-class atm basic-class
 precedence other
 protect group
 ubr 10000
```

次のコマンドセットは、ルータのサブインターフェイスが 3 つのネイバーへの接続に使用する 3 つのバンドルを設定します。これらのバンドルは new-york、san-francisco、および los-angeles と呼ばれます。バンドル new-york には 4 つの VC メンバ、バンドル san-francisco には 4 つの VC メンバ、バンドル los-angeles には 3 つの VC メンバがあります。

new-york バンドル

この例の最初の部分では、サブインターフェイスの IP アドレスとルータのプロトコル（ルータは IP ルーティング プロトコルとして IS-IS を使用）を指定します。また、1 つめのバンドル new-york を作成し、バンドル コンフィギュレーション モードを開始します。

```
interface atm 1/0.1 multipoint
 ip address 10.0.0.1 255.255.255.0
 ip router isis
 bundle new-york
```

バンドル コンフィギュレーション モード内から、コンフィギュレーションの次の部分では 2 つのプロトコル コマンドを使用して IP と開放型システム間相互接続（OSI）のトラフィック フローをバンドル内で有効にします。OSI ルーティング パケットは、バンドルで一番優先度の高い VC を使用します。OSI データ パケットがある場合は、バンドル内の最下位の先行 VC を使用します。設定されている場合、IPX や AppleTalk などの他のプロトコルは常にバンドル内の最下位の先行 VC を使用します

先行するコマンドと後続のコマンドのインデント レベルが示すように、new-york バンドルの下位には、プロトコルを設定するコマンドと、bundle-class というクラスをそれに適用するコマンドがあります。

```
protocol ip 1.1.1.2 broadcast
protocol clns 49.0000.0000.2222.00 broadcast
class-bundle bundle-class
```

bundle-class という名前のクラスは、バンドル new-york に適用され、**protocol ip inarp** コマンドが含まれています。継承ルールに従って、バンドル レベルで設定される **protocol ip** は、bundle-class クラスに指定された **protocol ip inarp** より優先されます。

pvc-bundle ny-control 207 で始まる次の一連のコマンドは、（ny-control、ny-premium、ny-priority、および ny-basic という）4 つの VC をバンドル new-york に追加します。特定のクラス（この設定例で事前に定義したクラスの 1 つ）が各 VC に適用され、そのクラスに含まれているコマンドによって指定されたパラメータで設定されます。

この設定の場合と同様、バンドルに属する個々の VC を設定するには、ルータがマザー バンドルに対してバンドル モードである必要があります。バンドルに所属する各 VC では、下位のモードは特定の VC の pvc モードです。

次のコマンドでは、バンドル new-york の個々の VC を設定します。

```
pvc-bundle ny-control 207
 class-vc control-class
pvc-bundle ny-premium 206
 class-vc premium-class
pvc-bundle ny-priority 204
 class-vc priority-class
pvc-bundle ny-basic 201
 class-vc basic-class
```

san-francisco バンドル

次のコマンド セットは、san-francisco という名前のバンドルを作成し、設定します。バンドル コンフィギュレーション レベルでは、bundle-class クラスに含まれるコンフィギュレーション コマンドは、バンドル san-francisco およびバンドルに所属する個々の VC に属します。その後、**pvc-bundle** コマンドを個々の VC ごとに実行してバンドルに追加します。VC が追加され、バンドル VC コンフィギュレーション モードが開始されると、特定の事前設定済みのクラスが VC に割り当てられます。そのクラスを構成するコンフィギュレーション コマンドは、VC の設定に使用されます。この時点で階層のルールが適用されます。適用されたクラスに含まれているコマンド パラメータは、バンドル コンフィギュレーション レベルで適用された同じパラメータによって置き換えられ、それらのパラメータは VC に直接適用された同じパラメータによって置き換えられます。

```
bundle san-francisco
 protocol clns 49.0000.0000.0000.333.00 broadcast
 inarp 1
 class-bundle bundle-class
 pvc-bundle sf-control 307
  class-vc control-class
 pvc-bundle sf-premium 306
  class-vc premium-class
 pvc-bundle sf-priority 304
  class-vc priority-class
 pvc-bundle sf-basic 301
  class-vc basic-class
```

los-angeles バンドル

次のコマンドセットは、los-angeles という名前のバンドルを作成して設定します。バンドル コンフィギュレーション レベルでは、bundle-class クラスに含まれるコンフィギュレーション コマンドは、バンドル los-angeles およびバンドルに所属する個々の VC に属します。その後、**pvc-bundle** コマンドを個々の VC ごとに実行してバンドルに追加します。VC が追加され、バンドル VC コンフィギュレーション モードが開始されると、VC に優先度が設定され、その VC は保護されたグループ（protect グループ）のメンバとして設定されるか、個別に保護された VC として設定されます。特定のクラスが各 VC に割り当てられて、さらに特徴付けられます。階層のルールが適用されます。バンドル コンフィギュレーション レベルでバンドル全体に適用したパラメータよりもバンドル VC コンフィギュレーション レベルで VC にクラス内で適用した同じパラメータのほうが優先され、さらに、それらのパラメータよりも VC に直接かつ個別に適用したコマンドのパラメータが優先されます。

```
bundle los-angeles
 protocol ip 1.1.1.4 broadcast
 protocol clns 49.0000.0000.4444.00 broadcast
 inarp 1
 class-bundle bundle-class
 pvc-bundle la-high 407
  precedence 7-5
  protect vc
  class-vc premium-class
 pvc-bundle la-mid 404
  precedence 4-2
  protect group
  class-vc priority-class
 pvc-bundle la-low 401
  precedence other
  protect group
  class-vc basic-class
```

例：スタンドアロン VC での VC 単位の WFQ と CBWFQ

次に、2 つのクラス マップを作成し、それらの一致基準を定義する例を示します。最初のマップ クラス class1 では、番号付きのアクセス コントロール リスト（ACL）101 を一致条件として使用します。2 つめのマップ クラス class2 では、番号付きの ACL 102 を一致基準として使用します。

次に、この例では、policy1 という名前のポリシーマップにこれらのクラスが含まれています。class1 のポリシーでは、このクラスのために予約されたキューに対して、最小帯域幅の割り当て要求に 500 kbps、最大パケット数の制限が 30 と指定されています。class2 のポリシーでは、最小帯域幅の割り当て要求に 1000 kbps のみが指定されているため、デフォルトのキュー制限は 64 パケットであると想定されます。policy1 を構成する 2 つのクラスの帯域幅要求の合計が、ポリシーマップが適用された cisco という名前の PVC の総帯域幅（2000 kbps）の 75% になることに注意してください。

この例では、policy1 というポリシーマップを PVC に適用します。ポリシーマップ policy1 が PVC に適用されると、そのクラスによってその PVC の CBWFQ サービス ポリシーが構成されます。この PVC で送信されるパケットの一致基準を ACL 101 および 102 と照合して確認し、それに応じて分類します。

class-default コマンドはこのポリシーマップに対して明示的に設定されていないため、サービス ポリシーを構成している 2 つのクラスの一致基準を満たさないすべてのトラフィックは、事前定義された class-default クラスによって処理されます。これによって、ベストエフォートのフローベースの WFQ が実現します。

```
class-map class1
 match access-group 101
class-map class2
 match access-group 102
policy-map policy1
 class class1
  bandwidth 500
  queue-limit 30
 class class2
  bandwidth 1000
interface ATM1/1/0.46 multipoint
 ip address 200.126.186.2 255.255.255.0
 pvc 46
  vbr-nrt 2000 2000
  encap aal5snap
  service policy output policy1
```

例：バンドルメンバー VC での VC 単位の WFQ と CBWFQ

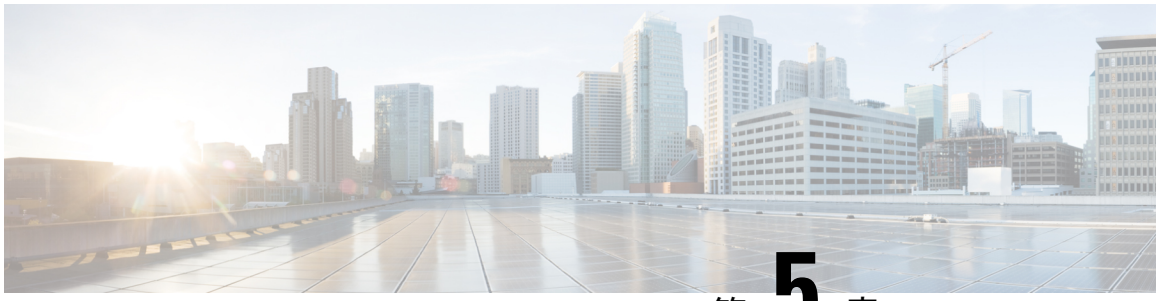
次の例では、VC 単位の WFQ および CBWFQ がイネーブルであり、サービス ポリシーが設定されているメンバを持つ san-francisco という名前の PVC バンドルを示します。この例では、ポリシーマップ policy1、policy2、および policy4 に含まれているクラスが定義されており、それらのポリシーマップが作成されていることを前提としています。各 PVC では、IP/ATM 間 CoS **pvc-bundle** コマンドを使用して、指定されたポリシーマップを適用する PVC を指定します。

PVC 0/34 および 0/31 には、同じポリシーマップ policy2 が適用されていることに注意してください。同じポリシーマップを複数の VC に割り当てることはできますが、各 VC に割り当てることができるのは、出力 PVC で適用されるポリシーマップ 1 つのみです。

```
bundle san-francisco
 protocol ip 1.0.2.20 broadcast
 encapsulation aal5snap
 pvc-bundle 0/35
```

```
 service policy output policy1
 vbr-nrt 5000 3000 500
 precedence 4-7
pvc-bundle 0/34
 service policy output policy2
 vbr-nrt 5000 3000 500
 precedence 2-3
pvc-bundle 0/33
 vbr-nrt 4000 3000 500
 precedence 2-3
 service policy output policy4
pvc-bundle 0/31
 service policy output policy2
```

例：バンドルメンバー VC での VC 単位の WFQ と CBWFQ



第 5 章

QoS Scheduling

この章では、インターフェイスを終了するために次のパケットを選択するプロセスと、それがいつ起こるかを概説します（以降、スケジューリングと呼ばれます）。スケジューリングのトピックでは、 **priority**、 **bandwidth**、**bandwidth remaining**、 **shape** および **fair-queue** コマンドを使用します。これらのコマンドを使用して、輻輳が発生した場合に帯域幅を分配し、アプリケーションがネットワーク上で動作するために必要な処理を確実に受信できるようにします。

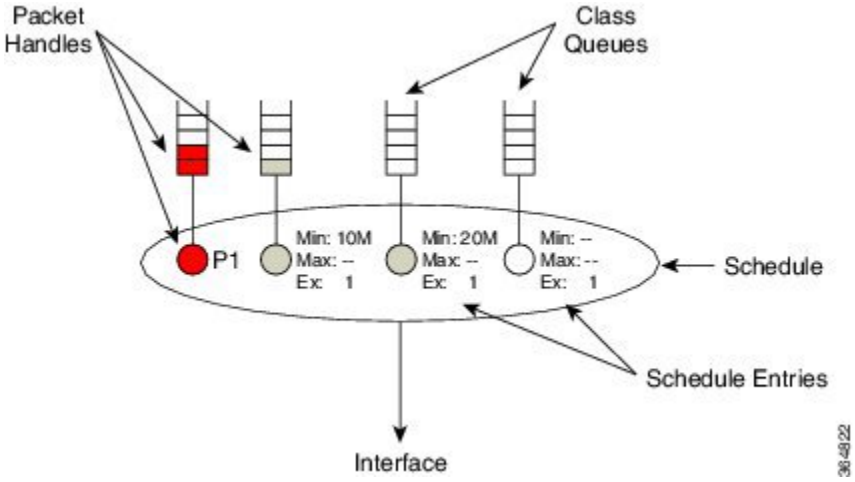
この章では、物理インターフェイスに適用されたフラットポリシーについて重点的に説明します。ここに記載されている内容は、以降の章で説明される階層型スケジューリングの概念についての理解を深めるためのものです。

QoS Scheduling について

定義

このセクションでは、コアの「スケジューリング」用語を定義します。

図 *1:* スケジューリングの定義



パケット処理（Packet Handle）

ルータはパケットを転送する準備が整うと、ルータはそのパケットを表すパケット処理を出力キューの1つに配置します。この処理は、パケットの長さやメモリ内のパケットの位置などの情報を保持します。

クラス キュー（Class Queues）

出力QoSが設定されると、*class queue*は、キューイング アクションを設定している各クラスに対して作成されます。同様に、明示的に作成されたキューイング クラスのいずれかに一致しないトラフィックに対して、*implicit class-default queue* を作成します。非キューイング アクションのみを使用してクラスを設定した場合（たとえば、マーキングのみを設定したクラス）、「一致する」パケットは class-default キューに入れられます。

スケジュール（Schedule）

意思決定者としてスケジュール（スケジューラ）を表示する必要があります。パケット処理を選択することによって、スケジュールでは、次に終了するパケットとその送信時期が選択されます。上の図の「楕円」 は、クラス キューの 1 つからパケットを選択する単一のスケジュールを表しています。

（注） 個々のスケジュールは、各インターフェイスに対して作成されます。

スケジュール エントリ（Schedule Entry）

スケジュールをキューから選択するには、各キューの予想される処理を知る必要があります。このタイプの情報は、スケジュールエントリに保存されます。たとえば、キューイング コマンド（例：**bandwidth 10 Mbps**）を設定することによって、スケジュール エントリを設定しています。

また、スケジュール エントリには、そのキューからのパケットが最後に送信された時刻や、そのキューからの現在のパケット処理（ある場合）などの内部状態も格納されます。

スケジュール エントリにはプライオリティ キュー （52 ページ）および帯域幅キュー （60 ページ）の 2 種類があります。

スケジュール エントリのプログラム方法

このセクションでは、スケジュールエントリ内で設定されるパラメータについて簡単に紹介します。実際のコマンドについては、この章で後ほど詳しく説明します。

まず、スケジュール エントリは、プライオリティ エントリまたは帯域幅エントリ（プライオリティ キューまたは帯域幅キュー） のいずれかとして設定されます。

以下の説明では、プライオリティ エントリが *P1* エントリまたは *P2* エントリにさらに分割することができることが分かります。P1 エントリ（既定値）は、**priority** または **priority level 1** コマンドのいずれかを使用して設定します。同様に、**priority level 2** コマンドを使用して P2 エントリを設定します。

帯域幅エントリには 最小レート（Min）、最大レート（Max）、および超過ウェイト（図では「Ex」と表示）の 3 つの異なるパラメータがあります。

（注） ASR 1000 シリーズ アグリゲーション サービス ルータのスケジュールは、3 つのパラメータによるスケジュールとよく呼ばれます。

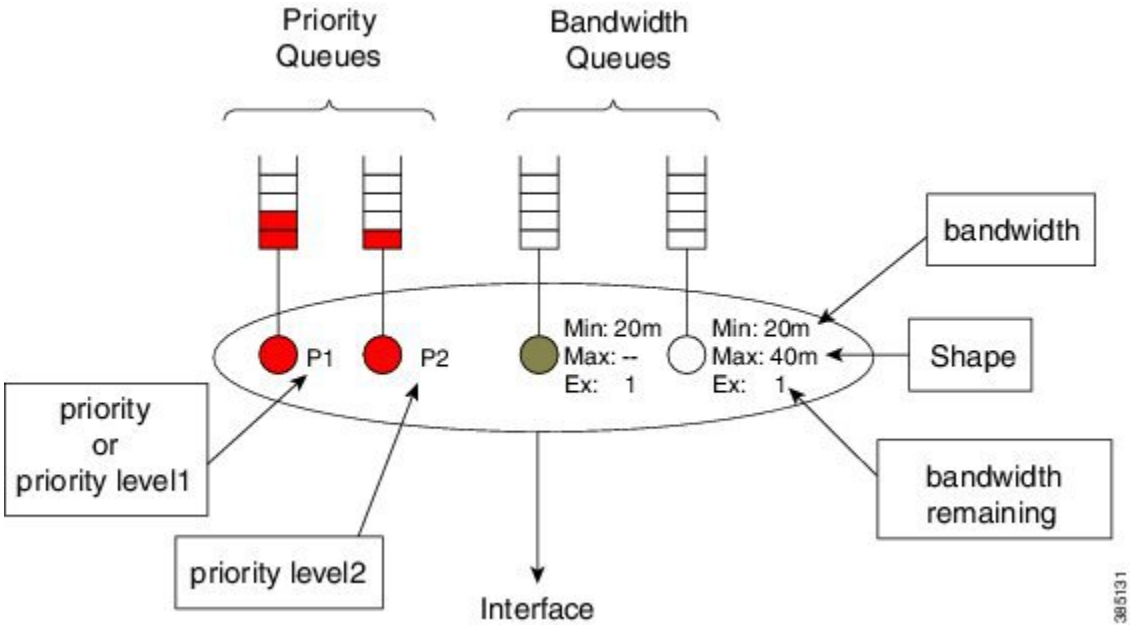
最小（最小レート）エントリは、 最小帯域幅における保証スループット量をキューに割り当てます。最小エントリは **bandwidth** コマンドで設定され、明示的に設定しない限り設定されません。IOS の設定チェックでは、スケジュールに設定されている最小レートを満たすのに十分な帯域幅が常にあることを確認しようとします。スループットの監視と事前設定された目標レートに基づいてキューを処理することは、リアルタイム スケジュール（「スケジューラによる時間表記 （71 ページ）」を参照）と呼ばれることがあります

最大（最大レート）エントリは、 キューが受け取ることができるスループットの量の上限を設定します。最大エントリは **shape** コマンドを使用して設定され、明示的に設定しない限り設定されません。Max はキューのスループットに上限を設定しますが、それ自体はそのキューへのスループットを保証するものではありません。

Ex（超過ウェイト） エントリでは、Priority および Min の保証されたスループットの量が満たされた後に使用可能な帯域幅（超過帯域幅、または優先順位および帯域幅保証に対して保証されていない、または使用されていない使用可能帯域幅）に対するキューの競合方法を指定します。超過重量を **bandwidth remaining** コマンドで設定し、明示的に設定されていない限り、デフォルトは 1 になります。超過帯域幅の共有 は、キューの超過ウェイトに比例します （レートが設定されておらず、相対的な動作が重要であるため、仮想時間のスケジュールとも呼ばれます）。帯域幅共有についての考察については、「スケジュール エントリのプログラム方法 （39 ページ）」を参照してください。

次の図は、上に示した内容をまとめたものです (各スケジュール エントリを設定するコマンド)。

図 *2:* スケジュール エントリを設定する *IOS* コマンド



スケジュール操作

スケジュールがどのようにパケット シーケンスを決定するかを、以下にまとめました。

（注） 各パケットが転送された後、ステップ 1 に戻ります。

1. P1 キューが空でない場合は、P1 パケットを送信します。
2. PI キューが空で、P2 キューが空でない場合は、P2 パケットを送信します。
3. すべてのプライオリティ キューが空の場合、スケジュールは最低帯域幅保証（Min）があるすべてのキューを処理し、帯域幅保証が満たされるまでそのようなキューを処理し続けます。均等性を確保するために、スケジューラは、適格キュー、つまり帯域幅保証を超えず最も長く待機していたキューを選択することによって、最低帯域幅保証が設定されているキューから選択します。
4. プライオリティ キューが空で、すべての帯域幅保証が満たされている場合はどうなりますか。超過帯域幅は、すべての帯域幅が消費されるか、特定のキューが設定された最大帯域幅に達するまで、まだサービスを必要とするキュー間で分散されます。そのキューのスケジュール エントリで設定された Ex では、 各キューがこの超過帯域幅を受け取る割合を決定します。

シェーパーなしでのスケジュール操作

次の例は、スケジュールがどのように機能し、指定された負荷に対して各キューが受け取る帯域幅を決定する方法を示しています。

例を説明する前に、プライオリティ キュー アドミッション コントロールの概念について説明します。スケジュール操作のこれまでの説明では、スケジュールがプライオリティキューをどのように処理するかにおいて、レートが設定されていないに気付くでしょう。パケットが含まれているときはいつでも、スケジュールは単にプライオリティ キューを選択します。

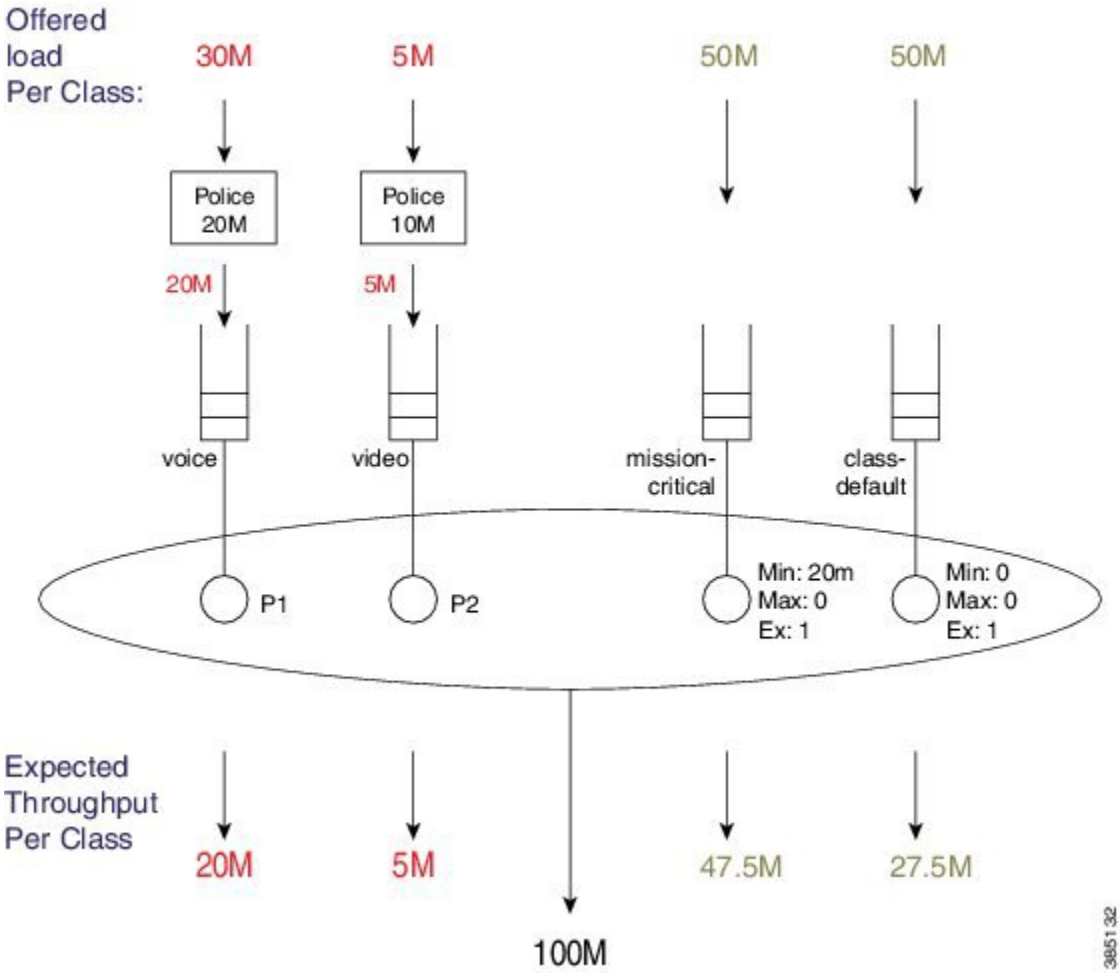
プライオリティ キュー（クラス）により他のサービス キューが利用できる帯域を消費されないようにするには、ポリサーを使用して、利用できる帯域幅を制限します。このようなポリサーは、そのキューにパケットを入れることができるレートを制限します。

次の例では、100 Mbps インターフェイスにポリシーを適用します。

```
policy-map scheduling-example
  class voice
    priority level 1
    police 20m
  class video
    priority level 2
    police 10m
  class mission-critical
    bandwidth 20000
  class class-default
```

（注） 帯域幅は Kbps 単位で設定されます。**police** および **shape** コマンドでは、ユニットを指定するために接尾辞をサポートしますが、**bandwidth** コマンドではサポートしません。

図 *3:* スケジュール操作



各クラスに提供された負荷は図の上部に示されているように、30 M、５ M、50 M、50 M です。プライオリティ キューにポリサー（20M と 10M）を適用しました。

30 Mbps が音声クラスに提供され、最初に 20 Mbps ポリサーを通過し、20 Mbps を P1 キューに入れます。常にこのキューを最初に処理するため、キューに入れた 20 Mbps がすべて転送されます。

5 Mbps がビデオ クラス（すべてが 10 mbps ポリサーを通過）に提供され、5 mbps がビデオ キューにキューイングされます。80 Mbps（100 Mbps ～ 20 Mbps）の帯域幅がまだ使用できるため、5 Mbps すべてが転送されます。

プライオリティ キューの処理後、明示的な帯域幅保証のあるキューに進みます。mission-critical クラスには最大帯域幅 20 Mbps があるため、そのクラスは少なくてもその量のスループットを得ることになります。

使用可能な超過帯域幅は、55Mbps（100 Mbps - 20 Mbps - 5 Mbps - 20 Mbps）です。class-default クラスと mission-critical クラスの両方にデフォルトの超過ウェイトとして 1 があるため、27.5 Mbps（55Mbps/2 =）の使用可能な超過帯域幅をそれぞれ同等に得ることができます。

mission-critical クラスは、47.5 Mbps（20 Mbps + 27.5 Mbps）の総スループットを監視します。

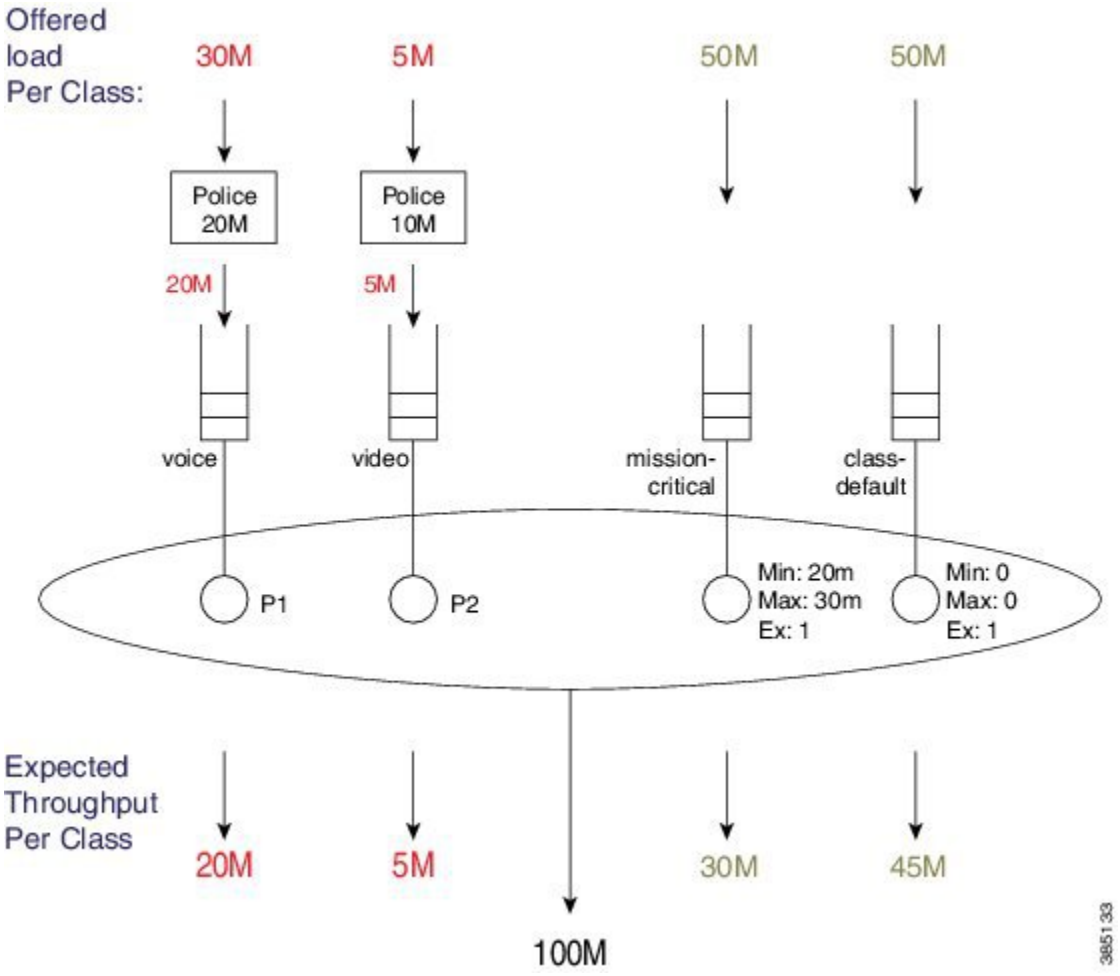
シェーパーを使用したスケジュール操作

設定を少し変更してみましょう。mission-critical クラスに最大値（シェーパーを設定）を追加します。

```
policy-map scheduling-example
  class voice
    priority level 1
    police 20m
  class video
    priority level 2
    police 10m
  class mission-critical
    bandwidth 20000
    shape average 30m
```

（注） ポリシー定義から class-default を除外しました。明示的に定義しているかどうかにかかわらず、常に class-default は存在します。

図 *4:* スケジュール操作



各クラスに提供される負荷は、以前とまったく同じで、30M、5M、50M、および 50M です。

30 Mbps が音声クラスに提供され、最初に 20 Mbps ポリサーを通過し、20 Mbps を P1 キューに入れます。常にこのキューを最初に処理するため、キューに入れた 20 Mbps がすべて転送されます。

5 Mbps がビデオ クラス（すべてが 10 Mbps ポリサーを通過）に提供され、5 Mbps が P2 キューにキューイングされます。 80 Mbps（100 Mbps ～ 20 Mbps）の帯域幅がまだ使用できるため、5 Mbps すべてが転送されます。

プライオリティ キューの処理後、明示的な最低帯域幅保証のあるキューに進みます。mission-critical クラスには 20 Mbps の帯域幅保証があるため、少なくともそのスループットの量を得ることになります。

使用可能な超過帯域幅は、55 Mbps（100-20-5-20 Mbps）です。class-default クラスと mission-critical クラスの両方にデフォルトの超過ウェイトとして 1 があるため、使用可能な超過帯域幅をそれぞれ同等に得ることができます。帯域幅がキューの Ex に比例する超過帯域幅の共有「ルール」

をベースに、各クラスは 27.5 Mbps を受け取ります。（この「ルール」の詳細については、「スケジュール エントリのプログラム方法 （39 ページ）」を参照してください。）

帯域幅保証と帯域幅の共有に基づいて、mission-critical キューは 47.5 Mbps（20 + 27.5 Mbps）を受け取ります。ただし、Max が設定したシェーピング レートが 30 Mbps に設定されているため（この例では Max は 0 に設定されているます）、キューはこれほど多くの帯域幅を使用できません。その結果、キューは（帯域幅の共有から受信した 47.5 Mbps のうち）30 Mbps を使用し、追加の 17.5 Mbps の帯域幅は超過プールに戻ります。

class-default はまだ帯域幅を要求している唯一のキューであるため、競合することはなく、この余分な 17.5 Mbps を消費する可能性があり、その合計スループットは 45 Mbps に増加します。

（注） この例は、帯域幅が浪費されない方法を示しています。スケジューリングは、次のいずれかに該当するまで、適格なキューおよび配分帯域幅をソートし続けます。

- 各キューは空です。
- すべての最大値に達しました。
- すべての帯域幅が消費されました。

レートおよびバースト パラメータの設定

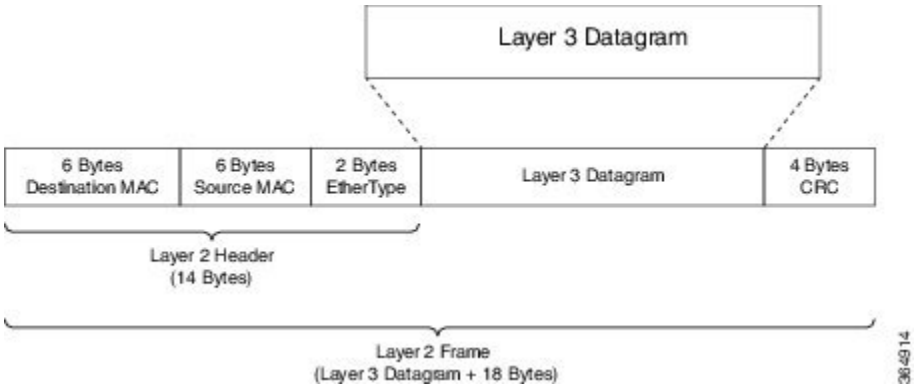
スケジューリング レート計算（オーバーヘッド アカウンティング）に含まれるもの

スケジュール操作 （40 ページ）の説明から、Min と Max がビット/秒で設定されていることがわかります。さて、これらのレートには何が含まれますか。簡単に説明すると、スケジュールにはレイヤ 3 のデータグラムとレイヤ 2 のヘッダーの長さが含まれていますが、CRC とパケット間のオーバーヘッドはどちらも含まれません。

レイヤ 3 データグラム

さらに理解するために、GigabitEthernet リンクを介して IP データグラムを転送することを想像してみましょう。

（注） 今後は、スケジューリング長として「パケット長に対するスケジュールの認識」を参照します。

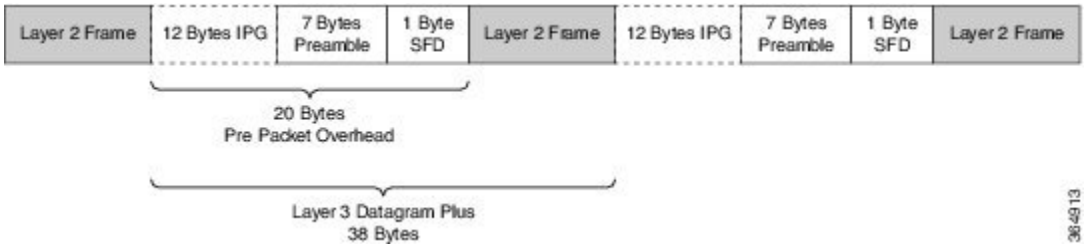


イーサネット オーバーヘッド

データグラムをGigabitEthernetリンク上で転送するには、まずイーサネットフレームに正しくカプセル化する必要があります。このプロセスでは、14バイトのレイヤ2ヘッダーと追加の4バイトのCRCが追加されます（カプセル化の合計は18バイトになります）。

このレイヤ2のフレームが物理メディアを介して送信されるとどうなるかを考えてみましょう。イーサネットは、合計20バイトのプリパケットオーバーヘッドに対して、12バイトのデータの送信時間に等しい最小インターパケットギャップ（IPG）と、プリアンブルの7バイト、およびフレーム開始区切り文字（SFD）を必要とします。

図 *5:* イーサネット オーバーヘッド



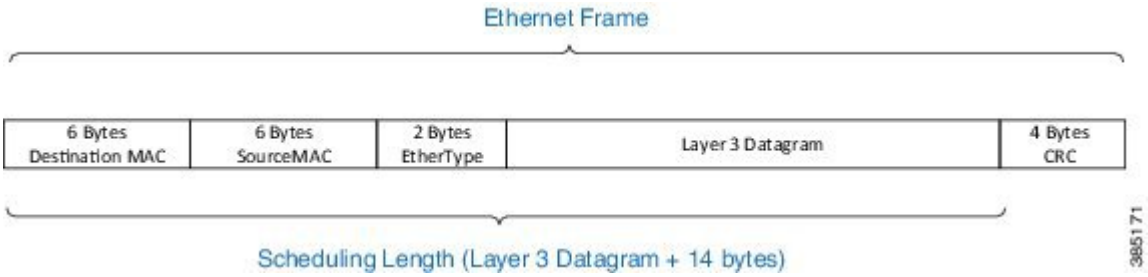
したがって、複数のイーサネットフレームを順次に送信する場合、各レイヤ3データグラムのパケットごとのオーバーヘッドの合計は、追加の38バイト（カプセル化（18バイト）＋イーサネットパケット間オーバーヘッド(20バイト)）です。たとえば、GigabitEthernetリンクのライン レートで100バイトのIPデータグラムを送信する場合、1秒あたりのパケットの予想スループットは次のようになります。

```
Linerate / Bits Per Byte / ( Layer 3 length + Per Packet Overhead) = Packets Per Second
1 Gbps / 8 / ( 100 + 38 ) = 905, 797 pps
```

スケジューリング長

スケジューラの観点からは、パケットの長さはレイヤ3データグラム＋レイヤ2ヘッダ（GigabitEthernetインターフェイスでは14バイト）です。

図 *6 :* スケジューリング長



ここで、GigabitEthernet インターフェイスに設定された 500 Mbps シェーパーについて考えてみましょう。（シェーピングはダウンストリームデバイスで輻輳が発生しないようにトラフィック レートを調整しながら、トラフィックの最大レートを強制するプロセスのことです。）前の例と同様に、100 バイトの IP データグラムすべてをスケジュールに送信し、その結果、「スケジューリング長」は 114 バイト（100 バイト（データグラム）+ 14 バイト（イーサネット レイヤ 2 ヘッダー））になります。次の式に従うと、予想されるスループットは次のようになります。

```
  Shaper Rate / Bits per Byte / (Layer 3 length + Layer 2 header length)
= Packets Per Second

  500 Mbps / 8 / (100 + 14) = 548,246 pps
```

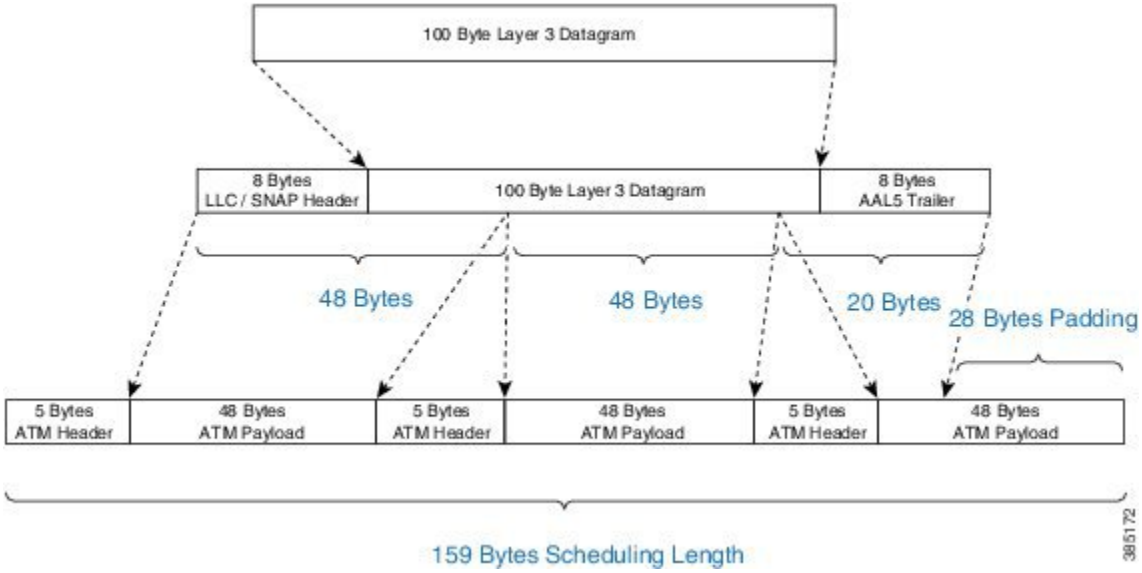
100% のラインレート（100 バイトのデータグラムすべて）は 1 秒間に 905,797 パケットであったのに対して、500 Mbps（100 バイトのデータグラムすべて）にシェーピングすると、毎秒 548,246 パケットのスループットが得られました。これは明らかに物理容量の 50% をはるかに超えています。帯域幅を割り当てるためにレートを指定する場合、レートには、そのパケットを転送するために必要なすべてのオーバーヘッドが含まれないことに注意してください。

ATM インターフェイス上のスケジューラ

キューイング ポリシー（スケジューリング ポリシー）が ATM VC に接続されている場合、そのポリシーのスケジューリング レートにはすべてのセル タックスが含まれます。これは AAL5 ヘッダーのみが含まれているようなポリシーとは設定が異なります。

たとえば、AAL5 SNAP カプセル化で設定されている ATM VC 上で送信される 100 バイトのデータグラムについて考えてみます。 ルータは 8 バイトの LLC/SNAP ヘッダーをデータグラムに追加し、ポリシング長は 108 バイトになります（100 バイトの IP データグラムのスケジューリング長は 114 バイトに相当）。（スケジューリング レート計算（オーバーヘッドアカウンティング）に含まれるもの （45 ページ） を参照）。（ポリシング長さの詳細については、「プライオリティ ポリシング長 （58 ページ）」を参照してください。）

図 *7: ATM* インターフェイス上のスケジューラ



パケットを伝送するには、ルータで8バイトのAAL5トレーラを（ポリシング長に）追加してから、パケットをATMセルに流し込む必要があります。このパケットを伝送するには3つのATMセルが必要で、それぞれのパケットは48バイトです。3番目のセルのペイロードが48バイトになるようにパディングします。

これら3つのセルの長さはそれぞれ53バイト（48バイトのパケット＋5バイトのATMヘッダー）です。つまり、100バイトのデータグラムのスケジュールの長さは159バイト（3セルx各セル53バイト）になります。

論理インターフェイス上のスケジューラ

ポリシングの章では、ポリシーが物理または論理インターフェイスに適用されているかどうかによって、ポリサーのオーバーヘッドアカウンティングがどのように異なるかを説明します（「論理インターフェイス上のポリサー（312ページ）」を参照）。この状況は、スケジュールには適用されません。ポリシーは論理インターフェイス（トンネルインターフェイス）に適用されていますが、物理インターフェイスを出力するためにパケットをキューに入れる前に、すべての処理を完了し、必要なヘッダーを追加する必要があります。エンキュー時のパケットの最終的な長さがわかっているので、その時に応じてスケジューリング長を設定できます。

スケジュール オーバーヘッド アカウンティングの調整

前のセクションでは、スケジューラのレートの計算に既定で含まれるものについて説明しました。ただし、場合によっては、既定とは異なる動作をユーザが望むことがあります。

たとえば、ユーザーがリンクで消費される物理的な帯域幅をレートで表示する必要があるという話を聞きます。イーサネットインターフェイスの場合、各パケットに必要な4バイトCRCと20バイトのパケット間オーバーヘッドを含める必要があります。

また、サービス プロバイダは、レイヤ 3 レートでトラフィック スループットを顧客に請求することを希望しています。データグラムの長さは、パケットがさまざまなインターフェイス タイプまたはカプセル化プロトコルを通過するときに一定のままであるため、ユーザにとって分かりやすくなります。この例では、シェイプ レートの計算にレイヤ2のヘッダーの長さは含まれません。

（注） オーバーヘッドアカウンティングの変更は、ネットワークに影響を与える可能性があります。たとえば、ネットワークアドミッション制御にポリサーを使用する場合、通常、顧客宅内機器にシェーパーを設定してそのネットワークに接続します。シェーパーおよびポリサーには、CIR に含まれるものと同じビューが必要です。

スケジュール アカウント オプション

スケジュール アカウント オプション（**account** キーワード） を使用すると、目的の動作を実現するために、パケットごとにデフォルトの「スケジューリング長」に追加または削除する必要があるバイト数を指定できます。パケットごとに最大63バイトを加算または減算できます。このオプションは、 **shape** および **bandwidth** コマンドでサポートされています。

次の例では、イーサネットインターフェイスにシェーパーを適用し、シェーパーが実際の物理帯域幅の50%でスループットを制限するように、すべてのオーバーヘッドを含めます。パケットごとに24バイトを追加することで、4バイトのCRCと20バイトのパケット間オーバーヘッドを「カバー」します。

```
policy-map ethernet-physical-example
  class class-default
    shape average percent 50 account user-defined 24
```

（注） オーバーヘッドアカウンティングでは、階層型ポリシーを考慮する必要があります。アカウントオプションで親シェーパーが設定されている場合、すべての子シェーパーまたは帯域幅保証も、親ポリシーで指定されているのと同じ調整を継承します。

階層型スケジューリングの章では、シェーパーを使用してリモートリンクのトラフィックを調整し、そのシェーピング レート内で帯域幅を割り当てるために子ポリシーを使用する方法について説明します。その使用例では、リモートリンク上でのカプセル化は送信装置上でのカプセル化とは異なる場合があります（たとえば、イーサネット インターフェースを介してネットワークに接続されたエンタープライズ ハブ ルータはT1インターフェースに接続されたブランチにトラフィックを送信します）。T1リンクがHDLCカプセル化を使用している場合、各データグラムにはそのリンクの4バイトのレイヤ2ヘッダーが含まれるようになります。ただし、イーサネットでは、各パケットには 14 バイトのレイヤ 2 ヘッダーが含まれます。アカウントオプションを使用すると、そのリモートリンクに表示されるパケットをシェーピングし、スケジュールすることができます。つまり、イーサネット ヘッダーが存在しなくなるため、スケジューリング長から 14 バイトを削除し、HDLC レイヤ 2 のオーバーヘッドを表示するためにスケジューリング長に 4 バイトを追加します。

オーバーヘッド アカウンティングの調整（事前定義オプション）

加算または減算するバイト数を指定することに加えて（次の表を参照）、CLIにはリモート カプセル化を指定できるいくつかの事前定義済みオプションもあります。現在の定義済みオプションは、イーサネットインターフェイス上のトラフィックをネットワーク内の他の場所にある DSLAM に送受信することを前提としており、ブロードバンド ユース ケースに基づいています、Dot1Q またはQ-in-Q を含むイーサネット フレームにカプセル化していますが、DSLAM は何らかの形式の ATM カプセル化を受信します。DSLAM の後にトラフィックがどのように表示されるかが反映されるように、シェーパーでトラフィックを調整するようにします。いずれの場合も、スケジューリング長にセル タックスを追加します。

表 *2:* オーバーヘッド アカウンティング調整に事前に定義されたオプションの表

CLI	値（dot1q/qinq）	ATM	詳細（dot1q/qinq）
account dot1q|qinq aal5 mux-1483routed	-15/-19	○	dot1q: 3 byte 1483 routed - 18 byte dot1q qinq: 3 byte 1483 routed - 22 byte qinq
account dot1q|qinq aal5 mux-dot1q-rbe	0/-4	○	dot1q: 0 byte mux_rbe + 18 byte dot1q - 18 byte dot1q qinq: 0 byte mux_rbe + 18 byte dot1q - 22 byte qinq
account dot1q|qinq aal5 mux-pppoa	-22/-26	○	dot1q: 2 byte mux_pppoa - 6 byte pppoe - 18 byte dot1q qinq: 2 byte mux_pppoa - 6 byte pppoe - 22 byte dot1q
account dot1q|qinq aal5 mux-rbe	-4/-8	○	dot1q: 0 byte mux_rbe + 14 byte 802.3 - 18 byte dot1q qinq: 0 byte mux_rbe + 14 byte 802.3 - 22 byte qinq
account dot1q|qinq aal5 snap-1483routed	-12/-16	○	dot1q: 6 byte snap 1483 routed - 18 byte dot1q qinq: 6 byte snap 1483 routed - 22 byte qinq
account dot1q|qinq aal5 snap-dot1q-rbe	10/6	○	dot1q: 10 byte snap_rbe + 18 byte dot1q - 18 byte dot1q qinq: 10 byte snap_rbe + 18 byte dot1q - 22 byte qinq
account dot1q|qinq aal5 snap-pppoa	-20/-24	○	dot1q: 4 byte snap_pppoa - 6 byte pppoe - 18 byte dot1q qinq: 4 byte snap_pppoa - 6 byte pppoe - 22 byte qinq

CLI	値（dot1q/qinq）	ATM	詳細（dot1q/qinq）
account dot1q|qinq aal5 snap-rbe	6/2	○	dot1q: 10 byte snap_rbe + 14 byte 802.3 - 18 byte dot1q qinq: 10 byte snap_rbe + 14 byte 802.3 - 22 byte qinq
account user-defined <value>	<value>	x	
account user-defined <value> atm	<value>	○	

イーサネット インターフェイスで Dot1Q カプセル化されたパケットを転送することを想像してください。その後 DSLAM が次のようになることをさらに想像してください。

- パケットを受信する
- イーサネットおよび Dot1q ヘッダーを削除する
- AAL5-Mux 1483 ルーティング カプセル化を行う

前の表を参照すると、DSLAM は1 8 バイトのイーサネット/Dot1q を削除し、3 バイトの LLC ヘッダーを追加して、スケジューリング長において3バイトの変更を生成します。（展開図については、「スケジューリング レート計算（オーバーヘッド アカウンティング）に含まれるもの （45 ページ）」を参照してください。）

DSLAM が ATM ネットワークを経由して送信するとき、8 バイトの AAL トレーラーを追加し、その結果の PDU を 53 バイトのセルに分割します。ATM 値の "yes"（表を参照）は、ルータでこのセルタックスが計算され、その余分なオーバーヘッドがスケジューリング長に追加されることを示します。

```
policy-map atm-example
  class class-default
    shape average 50m account dot1q aal5 mux-1483routed
```

例：事前定義済みオーバーヘッド アカウンティング

イーサネット インターフェイスで Dot1Q カプセル化されたパケットを転送することを想像してください。その後 DSLAM が次のようになることをさらに想像してください。

- パケットを受信する
- イーサネットおよび Dot1q ヘッダーを削除する
- AAL5-Mux 1483 ルーティング カプセル化を行う

前の表を参照すると、DSLAM は1 8 バイトのイーサネット/Dot1q を削除し、3 バイトの LLC ヘッダーを追加して、スケジューリング長において3バイトの変更を生成します。（展開図については、「スケジューリング レート計算（オーバーヘッド アカウンティング）に含まれるもの （45 ページ）」を参照してください。）

DSLAM が ATM ネットワークを経由して送信するとき、8 バイトの AAL トレーラーを追加し、その結果の PDU を 53 バイトのセルに分割します。ATM 値の "yes"（表を参照）は、ルータでこのセルタックスが計算され、その余分なオーバーヘッドがスケジューリング長に追加されることを示します。

```
policy-map atm-example
  class class-default
    shape average 50m account dot1q aal5 mux-1483routed
```

プライオリティ キュー

プライオリティ キューは、パケット転送における不要な遅延を回避できるようにするスケジュール エントリの一種です。プライオリティ セマンティックにより、遅延やジッタに敏感なアプリケーションに対して、低遅延処理を保証 することができます。. 例として、Voice over IP（VOIP）を考えてみます。通常の VOIP 電話には 30 ms のデジッタ バッファがあり、ネットワーク全体で、エンドツーエンドで最大 30 ms のジッタを許容できます。

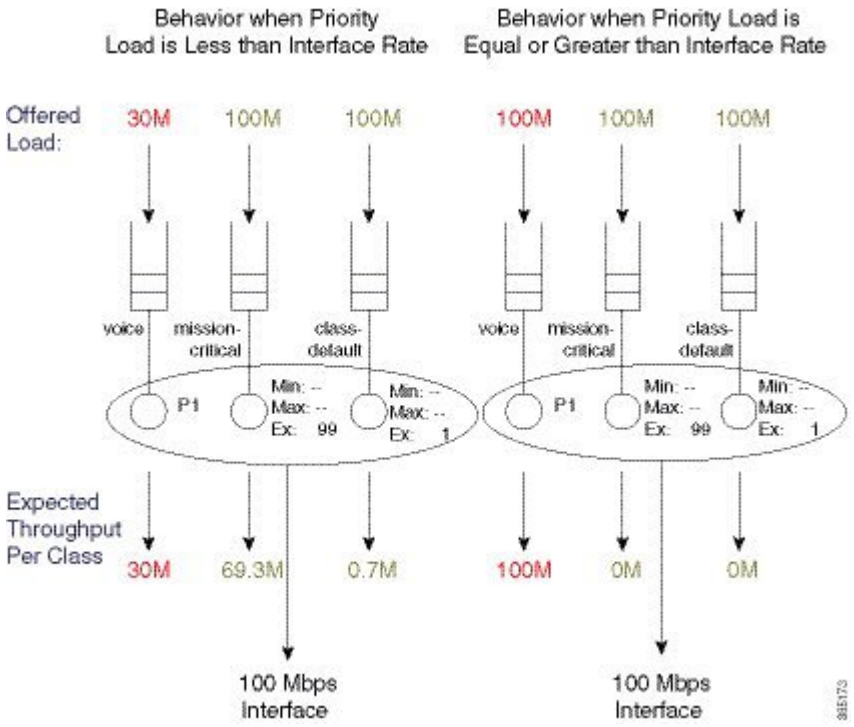
プライオリティ キューを設定するときに、そのトラフィック クラスで消費される可能性がある帯域幅を制御する 3 つの方法（非制約プライオリティ キュー、条件付きポリサー、または（無条件）常時オン ポリサー）があり、どれか 1 つを選択できます。

非制約プライオリティ キュー

帯域幅の消費を制御する 1 つの方法として、 非制約プライオリティ キューがあり（絶対プライオリティ キュー）、 これは、プライオリティ クラスによって消費される可能性のある帯域幅の量を制限せずに設定されたプライオリティ キューです。説明する上で、設定例と次の図の設定済みスケジュール エントリを見てください。プライオリティ コントロールへのアドミッション コントロールにポリサーがない ことに注意してください。

```
policy-map absolute_pq_example
  class voice
    priority
  class mission-critical
    bandwidth remaining percent 99
  class class-default
    bandwidth remaining percent 1
```

図 *8 :* 非制約プライオリティ キュー



左の例では、実際のインターフェイス帯域幅容量（100 Mbps）が、音声クラスのプライオリティ キューへの負荷（30 Mbps）を超えています。これにより、超過ウェイト（Ex）比（99：1、 **bandwidth remaining** コマンドで設定）に基づいて 70 Mbps の超過帯域幅が割り当てられます。

右側の例では、プライオリティの負荷が 100 Mbps に引き上げられています。これによって、余分な帯域幅が残らないため、他のキューのサービスがなくなってしまいます（予期される 0M スループット）。

ここで覚えておく必要がある点は、アドミッション コントロールなしのプライオリティ クラスです。このクラスによりインターフェイス全体の帯域幅が消費され、他のすべてのサービス キューの帯域幅がなくなる可能性があります。これにより、ミッションクリティカルなアプリケーションに悪影響が及ぶ可能性があります。さらに、制御メッセージが悪影響が及んでいるサービス キューにある場合、ネットワークが不安定になる可能性があります。

したがって、非制約プライオリティ キューは慎重に使用してください。プライオリティ キューによって他のサービスの帯域幅が消費されないようにするには、コール アドミッション制御（CAC）などの代替帯域幅制御システムの使用を検討することをお勧めします。

（注） 最低帯域幅保証（**bandwidth** コマンドによって設定される）を非制約プライオリティ キューと組み合わせて使用することはできません。プライオリティ キューが使用可能な帯域幅をすべて消費する可能性がある場合は、その帯域幅を他のクラスに保証することはできません。IOS ではこのような設定は拒否されます。

条件付きプライオリティ キュー

帯域幅の消費を制御する別の方法は、**priority** コマンドで値を入力することです。（priorityに関しては、コマンドページを参照してください。）これは、条件付きポリサーを使用してキューアドミッション コントロールを処理する方法です。

条件付きプライオリティ レートは、輻輳が親（ポリシーマップまたは物理インターフェイス）レベルで存在する場合にのみ、ポリサーによるトラフィックを制限します。この状態は、クラス（やインターフェイス）内を移動しようとするトラフィックにおける最大レートを超えた場合に発生します。

重要な要素は、条件付きポリサーがスケジュールが輻輳している場合にのみ、パケットをドロップすることです。つまり、提供された負荷が、使用可能な帯域幅（物理インターフェイスに適用されたフラットポリシーマップのコンテキストにおけるインターフェイス帯域幅）を超えた場合にのみパケットをドロップします。

条件付きプライオリティクラスは、設定されたレートを超えるレートを使用できますが、同じポリシー内の他のクラスとの競合が存在しない場合に限ります。

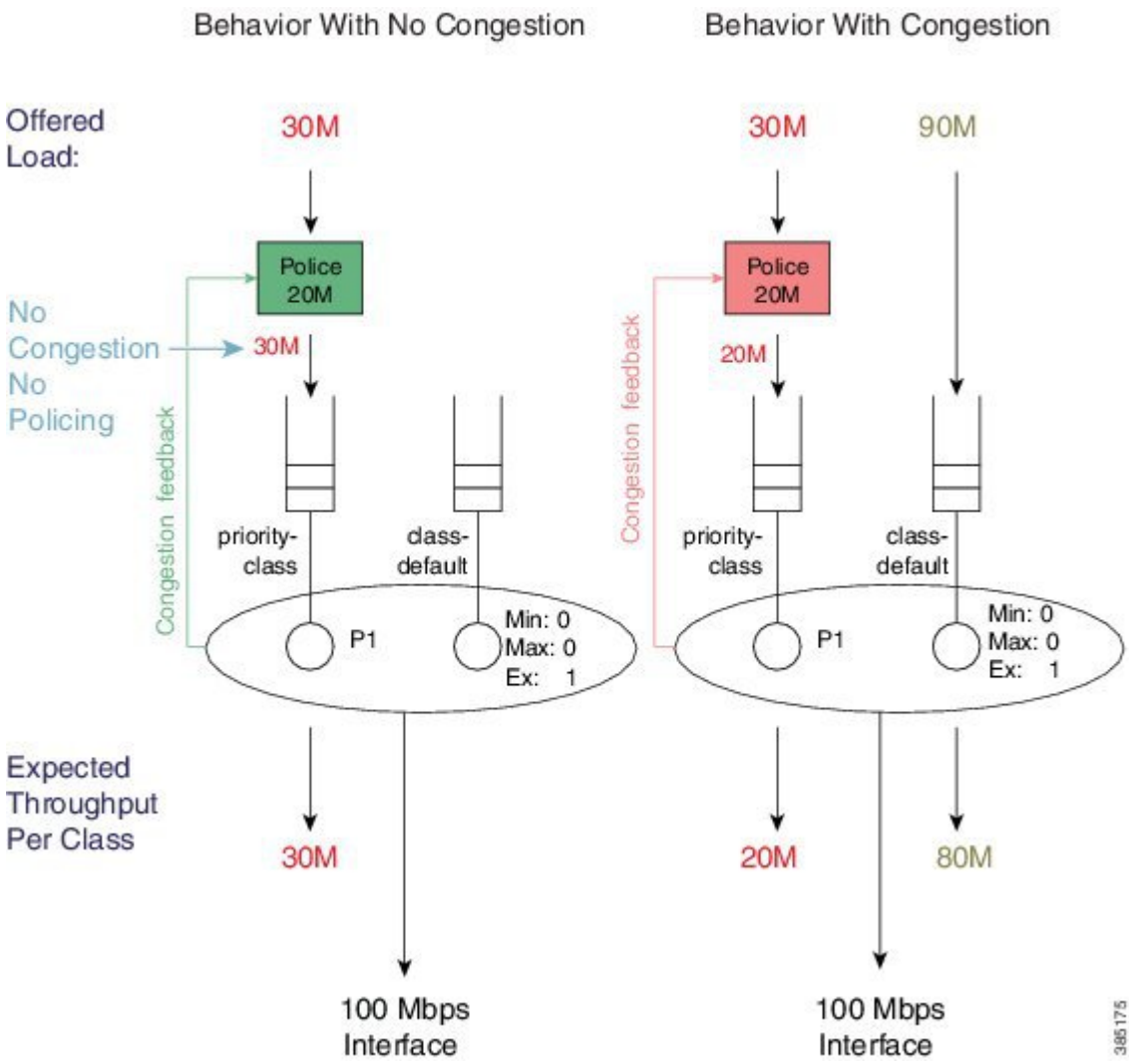
スケジュールは輻輳フィードバックをポリサーに提供します。ポリサーは観測した速度ですべてのパケットをカウントしますが、輻輳が発生しない限りドロップアクションを抑制します。そのため、輻輳が発生していない場合、プライオリティ キューは使用可能な帯域幅をすべて消費しますが、輻輳が発生するとキューは設定されたレートにポリシングされます。

```
policy-map conditional_policer_example
  class priority-class
    priority 20000
```

プライオリティ値は kps で設定され、**priority** コマンドで設定されていますが、スケジュールエントリを変更しません。（スケジュール エントリは、キューの予期される処理を格納する場所であることを思い出してください。）代わりに、パケットがエンキューされる前に実行されるポリサーを設定します。

次の図は、条件付きポリサーが輻輳の有無に関係なくどのように機能するかを示しています。この例では、以前の設定を 100 Mbps インターフェイスに接続しています。

図 *9 :* 条件付きプライオリティ キュー



左側に描かれているスケジュールでは、輻輳は発生していません。輻輳フィードバックでは輻輳は報告されないので、ポリサーはそのクラスに提供される 30 Mbps 全体をエンキューします。

輻輳が発生している右側に示されているスケジュールでは、ポリサーは **priority** コマンドで設定された 20 Mbps レートを適用します。

条件付きポリサーにはメリットとデメリットがあることに注意してください。

メリット	プライオリティ クラスは、現在他のクラスに使用されいないすべての帯域幅を使用できます。

デメリット	特定のインターフェイスの転送容量がわからない場合は、ネットワーク全体の優先容量を慎重に計画することはできません。真のプライオリティ サービスは、低遅延（キューの増大なし）であり、パケットのドロップがなく、エンドツーエンドである必要があります。 インターフェイスが輻輳しているかどうかによって、一貫性のない動作がみられるようになります。ポリシング レートをアンダープロビジョニングした場合、そのクラスを使用しているアプリケーションで断続的な問題が発生し、問題を診断することが非常に困難になる可能性があります。

（注） 条件付きポリサーとポリサーのオーバーヘッドアカウンティングの調整を同時に使用することはできません。

常時オン（無条件）ポリサーを使用したプライオリティ キュー

帯域幅の消費を制御する 3 つ目の方法は、キュー アドミッション コントロールに対して明示的常時オン（無条件）ポリサーを使用することです。

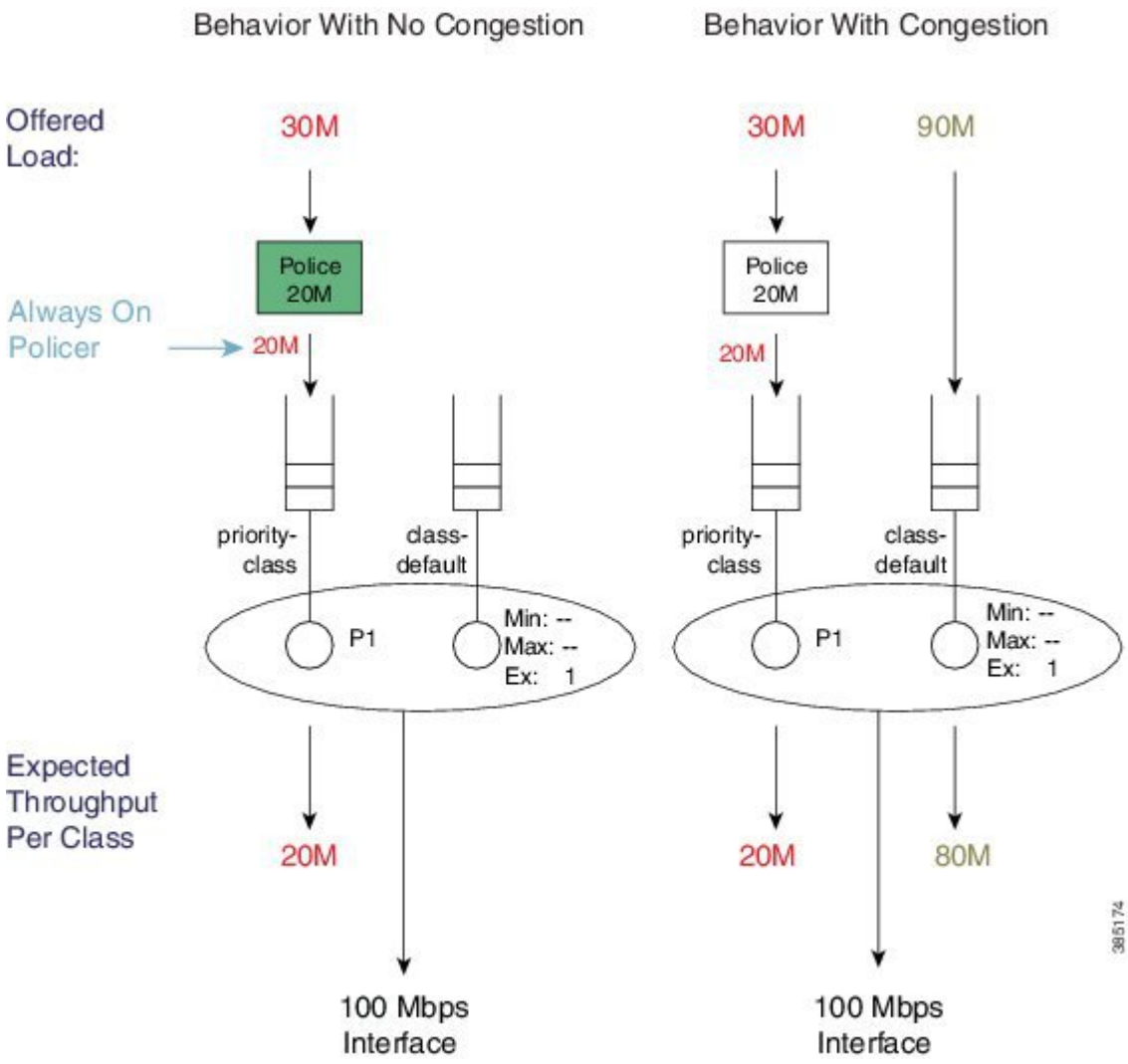
明示的なポリシング レートでプライオリティ クラスを設定する場合、輻輳条件に関係なく、トラフィックはポリシング レートに限定されます。 つまり、帯域幅が使用可能であっても、プライオリティ トラフィックは、明示的なレートを超えることはできません。

このような設定の例を次に示します。

```
policy-map always_on_policer
  class priority-class
    priority
    police cir 20m
```

次の図は、このようなポリサーの動作を示しています。

図 *10 :* 常時オン ポリサー



明示的なポリシング レートを使用してプライオリティ クラスを設定すると、このレートは常に適用されます。つまり、十分な帯域幅があっても、プライオリティ トラフィックが明示的なレートを超えることはできません。これは、決定論的動作が プライオリティ サービスにあることを意味します。ユーザーからアプリケーションのパフォーマンスが低下したというクレームがある場合は、ネットワーク内のポリサーによるドロップを見つけ、プライオリティ サービスに十分な帯域幅が割り当てられているかどうかを判断します。アプリケーションは、輻輳が発生しているかどうかにかかわらず、同じエクスペリエンスを持つ必要があります。

プライオリティ キューのバーストのおける考慮事項

前の章では、条件付きポリサーまたは常時オン ポリサーを使用してプライオリティ キューのキュー アドミッション コントロールを実行する方法について説明しました。 具体的には、シングルレート *2* カラー ポリサーを採用しています。ポリシングの章から、ポリサーはバースト

（シングルレート2カラーポリサー（306ページ））を許可するトークンバケット方式を使用して実装されていることがわかります。この方法でポリサーを使用する場合、この（バースト）許容値を制御することは非常に重要です。

ポリサーによって許可されているバーストは、プライオリティキューが蓄積されるという結果に終る可能性があります。それはそのキューの終わりに追加されるパケットの遅延が生成されるためです。パケットは送信前にキュー内の全ての先行パケットが送信されるのを待つ必要があります。遅延の量は、シリアル化遅延、つまり物理メディア上でこれらのパケットを送信するのにかかる時間によって異なります。

特定のフローから複数のパケットが到着すると、それらはプライオリティキューでは異なる状態になる可能性があります。あるパケットは到着し、すでに待機している多数のパケットの背後にキューイングされている可能性があるため、ある程度の遅延が発生します。 同じフロー内の次のパケットは、空のプライオリティキューに到着し、すぐにスケジュールされる可能性があります。 つまり、プライオリティ キューの輻輳による潜在的な遅延は、潜在的なジッタ（ジッタ、 受信パケットの遅延の潜在的な変動）でもあります。

IOS のポリサーのバーストの既定許容値 は、常時オン ポリサーで 250 ミリ秒に、 条件付きポリサーで200 ミリ秒に設定されています。ポリサーによってバーストのエンキューが許可される場合、これらの数値は潜在的なジッタにほぼ直接変換される可能性があります。 プライオリティ セマンティック（「プライオリティ キュー （52 ページ）」を参照）の概要では、音声アプリケーションは通常、ネットワーク全体で約 30 ms のジッタと 150 ms の遅延を許容できることが分かりました。 前者を考えると、通常、この予算の一部をネットワーク内の各ノードに割り当てます。 単純なガイドラインとしては、どの単一ノードであっても 5 ～ 10 ms のバースト許容値（および潜在的なジッタ）を許可することです。

たとえば、プライオリティ キューを 2 Mbps のレートでキュー アドミッション ポリサーを使用して設定し、バースト許容値を 5 ミリ秒とします。 5 ms で送信できるバイト数を計算します。

バースト ターゲット
= ポリース レート / 1 バイトあたり 8 ビット * 5 ミリ秒
= 2 Mbps / 8 * . 005 = 1250 バイト

常時オン ポリサーの場合、設定例は次のようになります。

```
policy-map always_on_policer_burst_example
  class voice
  priority
  police cir 2000000 1250
```

条件付きポリシングの場合、設定例は次のようになります。

```
policy-map conditional_policer_burst_example
  class voice
  priority 20000 1250
```

プライオリティ ポリシング長

「スケジューリング レート計算（オーバーヘッド アカウンティング）に含まれるもの （45 ページ）」の章では、スケジューリング長の概念を紹介しました。スケジューリング長とは、

スケジューラがレートへの適合性を評価するときにパケット長を「表示」する方法です。「ポリシング」の章では、ポリシング長（ポリサー レート計算（オーバーヘッド アカウンティング）に含まれるもの（311 ページ））と同様の概念も紹介しました。プライオリティ キューに設定されたレートはポリシング レートであるため、そのレートへの適合性を決定する際にポリシング長を使用します。この章で説明するように、ポリシーが物理インターフェイスに適用されている場合、ポリシングとスケジューリング長は同じになります。ポリシング長を変更するには、ポリサー オーバーヘッド アカウンティング機能を使用できます。

（注） **account** キーワードは、常時オン ポリサーではなく、条件付きポリサーでサポートされます。

マルチレベル プライオリティ キューイング

マルチレベル プライオリティ キューイング（MPQ）機能 では、単一のサービス ポリシーマップで、トラフィック クラスごとに異なるプライオリティ レベルを指定することによって、複数のトラフィック クラスに対して複数のプライオリティ キューを設定できます。

「スケジュール操作 （40 ページ）」では、プライオリティ キューが P1 または P2 になることを紹介しました。この機能の本来の目的は、トラフィック特性とジッタ耐性が異なるため、音声とビデオを別々のプライオリティ キューでサポートすることでした。特に、音声はパケット サイズがより小さく（通常、音声の場合約 80 バイトに対してビデオの場合は 1400 バイト）、より厳しいジッタ要件が求められます （非インタラクティブ ビデオは数百ミリ秒です）。そのため、音声用に P1 キュー、ビデオ トラフィック用に P2 キューを使用します。

今日、多くのビデオ アプリケーションは高度なアダプティブ コーデックを使用し、音声とビデオのコンテンツを別々のストリームに分離しています。一部の人は、ビデオ トラフィックは現在その動作が TCP に似ており、帯域幅キューではより効果的であると主張しています。低速リンクでのインタラクティブ ビデオでも P2 キューが必要になる場合があります。

マルチレベル プライオリティ キューイングを設定するには **priority** コマンドで **level** キーワードを使用する必要があります。 この機能は、条件付きポリサー、常時オン ポリサー、および絶対プライオリティ キューでサポートされています。

次に、条件付きポリサーを使用したマルチ レベル プライオリティ キューの例を示します

```
policy-map multilevel-example2
  class voice
    priority level 1 5000 3125
  class video
    priority level 2 10000 12500
```

次は、常時オン ポリサーを使用したマルチレベル プライオリティ キューの設定例です。

```
policy-map multilevel-example1
  class voice
    priority level 1
    police cir 5000000 3125
  class video
    priority level 2
```

```
police cir 10000000 12500
```

（注） レベルを明示的に設定しない場合、プライオリティ キューは P1 キューとして動作します。ただし、マルチレベル プライオリティ キューイングを構成する場合は、レベルを明示的に構成する必要があります。

たとえば、次の設定は拒否されます。音声クラスのプライオリティ レベルを明示的に設定する必要があります。

```
policy-map multilevel-rejection-example
  class voice
    priority
    police cir 5000000 3125
  class video
    priority level 2
    police cir 10000000 12500
```

帯域幅キュー

帯域幅キューにより、厳密な遅延要件がないアプリケーションに対して、インターフェイス帯域幅を割り当てることができます。スケジューリングの目的は、すべてのアプリケーションが必要な帯域幅を受信できるようにし、他のアプリケーションで使用可能にすることによって未使用の帯域幅を利用できるようにすることです。

次のように帯域幅の共有を検討できます。

アプリケーションが効果的に動作するように、帯域幅を保証します。たとえば、電子メール アプリケーションがビジネス クリティカルであり、ネットワークの輻輳時でも動作し続ける必要があると判断したとします。その場合は、ビジネス上の重要なアプリケーションに対して使用可能な帯域幅の量を常に保証する必要があります。

輻輳時に犠牲にするアプリケーションを決定します。たとえば、ソーシャル メディア アプリケーションはビジネス クリティカルではないと判断する場合があります。従業員はこのようなアプリケーション用にネットワークを使用できますが、ビジネス クリティカルなアクティビティを犠牲にすることはありません。これらのアプリケーションを、輻輳時に意図的にサービスが消費されるキューに配置することができます。

「スケジュール エントリのプログラム方法 （39 ページ）」で説明したように、域幅キュースケジュール エントリには、Min、Max、および Ex の異なる 3 つのパラメータがあり、それぞれ **bandwidth**、**shape**、および **bandwidth remaining** コマンドによって設定されます。それでは、これらのコマンドを詳しく見ていきましょう。

Bandwidth コマンド

bandwidth コマンドは、は、キューが処理されるスケジュール エントリに最低帯域幅（Min）保証を設定します。 アプリケーションの帯域幅の正確な要件を考慮すると、このコマンドは、アプリケーションが輻輳時に必要なものを確実に受信できるように便利な方法を提供します。

既定では、すべてのエントリに超過ウェイトが設定されているため、キューに対する追加保証サービスが提供される可能性があります。

帯域幅保証は Kbit/sec で設定され、1 Kbps 単位で設定できます。また、保証を物理回線レートとして設定することもできます。**show interface** コマンドを使用すると、帯域幅として公称インターフェイスレートのパーセンテージが表示されます。

たとえば、GigabitEthernet インターフェイスの **show interface gigabit x/y/z** コマンドは、1,000,000 kbit/sec の BW を表示し、パーセント値はこの公称レートのパーセントになります。したがって、Gigabitethernet インターフェイス上で **bandwidth percent 50** を設定した場合は、最小値 500 Mbps が設定されます。

コマンド **bandwidth** は **account** を受け入れます。これにより、レート適合性の計算に含まれるオーバーヘッドを調整できるようになります。ただし、設定された **account** 値はポリシーマップ全体で一貫している必要があります（ポリシーマップ内のすべての **bandwidth** コマンドとすべての **shape** コマンドは、同じアカウント値で設定されている必要があります）。

注意 高速インターフェイスで非常に低い帯域幅保証を設定しないでください。

「スケジュール エントリのプログラム方法 （39 ページ）」で学んだように、超過キューを処理する前に最低帯域幅が設定されているキューを処理します。 さらに、スケジューリング レート計算（オーバーヘッド アカウンティング）に含まれるもの （45 ページ） で説明されているように、デフォルトでは、パケットのスケジューリング長には、そのパケットを転送する上で消費される物理帯域幅が含まれていないことに注意してください（CRC またはパケット間オーバーヘッドは含まれません）これらの 2 つの事項を考慮に入れ、実際に利用可能な帯域幅を超える帯域幅を保証しないように注意する必要があります。それ以外の場合は、超過ウェイトを持つサービスによりキューが消費されてしまいます。

たとえば、最小値 98 Mbps をキューに設定し、FastEthernet インターフェイス（100 Mbps）にポリシーを適用するとします。100 バイトのフレームをすべて送信すると、各フレームのスケジューリング長は 96 バイトになりますが、それぞれ（必要なパケット間オーバーヘッドを含む）によって消費される実際の帯域幅は 120 バイトになります。

スケジューリング長の観点から見ると、物理帯域幅使用量として、98 Mbs は 120 バイト/96 バイト * 98 Mbps = 122.5 Mbps に変換されることになります。

実際の帯域幅/スケジューリング長 * Min = 物理帯域幅使用量
（100 バイト フレーム + パケット オーバーヘッドあたり 20 バイト）/（100 バイト フレーム - 4 バイト CRC）* 98 Mbps = 120 バイト/96 バイト * 98 Mbps = 122.5 Mbps

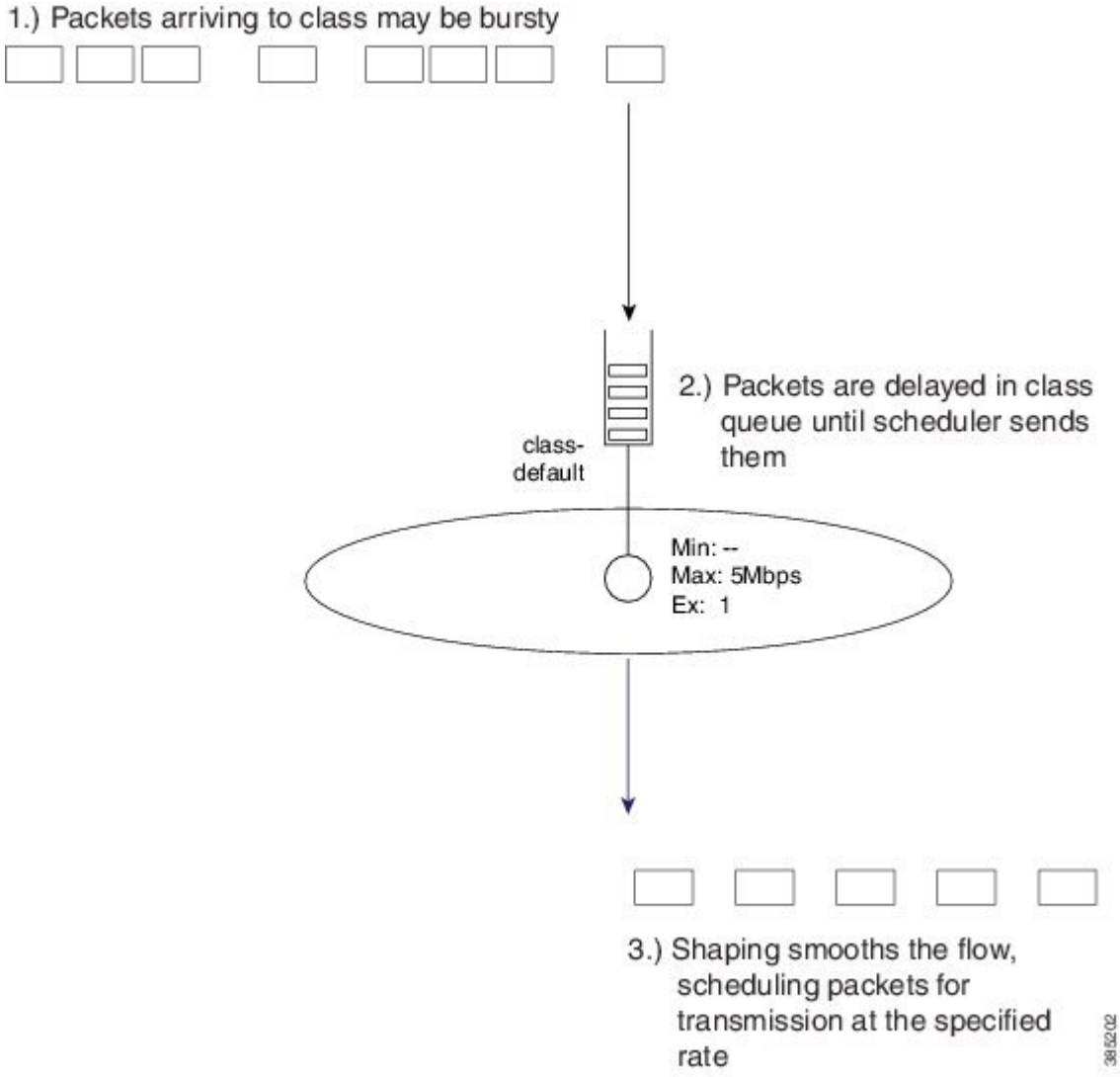
このように、帯域をお約束することはできません。 一般に、プライオリティ保証と最小帯域保証の合計が物理帯域の 75% 以上になる場合は、他のサービス キューの帯域が不足しているかどうかを確認してください。特に、class-default スケジュール エントリのデフォルト設定は超過ウェイトのみを設定するため、class-default トラフィックに何が発生する可能性があるかを考慮します。

Shape コマンド

shape コマンドは、 キューが処理されるスケジュール エントリの最大レートを設定します。 最大レートを設定しても、そのキューに対するスループットが保証されるわけではありません。 それは単に上限値を設定するのみです。単に **shape** shape コマンドを含むクラスを作成すると、そのクラスに割り当てられる帯域幅を決定するデフォルトの超過ウェイト設定（'1'）も追加されます。

シェーピングは階層型ポリシーで最も一般的に使用されています。ただし、場合によっては、フラットポリシーでシェーピングを使用することをお勧めします。つまり、帯域幅クラスに適応型ビデオがあり、制約のない場合は、帯域幅の使用量が物理的に利用可能な範囲を超えて拡大する可能性があります。 したがって、そのフローの拡大を制限するためにシェーピングを使用することをお勧めします。

図 *11 : Single-shaped* キュー



上の図は、single-shaped キューの例を示しています。この単純な例では、設定は次のようになります。

```
policy-map shape_example
  class class-default
    shape average 5m
```

パケットが到着すると、それらはそのクラスのキューの最後に追加されます。 スケジューラは、指定されたレートでキューの先頭からパケットをプルしています。 到着レート（パケットがキューに到着するレート）がサービス レート（パケットがキューからプルされるレート)を超える場合、パケット遅延となり、そのパケットはすべての先行パケットが送信されるまで

キューで待つ必要があります。この単純な例では、他のキューが帯域幅を奪い合うことはないため、サービス レートはシェープ レート（5 Mbps）と等しくなります。

この簡単な例から、シェーパーによりストリーミングが「円滑」に機能することがわかります。通常、スケジューラによって解放された単一のパケットではなく、少数の小さなパケットになります。最終結果は図のようになり、シェーパーはパケットが転送されるレートを測定します。

Shape Average

shape average コマンドは、クラスの最大レートを設定するための主要な手段です。

レートは1秒あたりのビット数、またはインターフェイス（または親シェーパー）のレートに対する割合で設定できます。他のスケジューリング コマンドと同様に、**account** キーワードを使用してスケジュール計算に含まれるオーバーヘッドを調整できます。

例として、次のようにイーサネット インターフェイスに CRC とパケット間オーバーヘッドを含めるように以前の設定スニペットを変更できます（詳細については、「スケジューリング レート計算（オーバーヘッド アカウンティング）に含まれるもの （45 ページ）」を参照してください）。

```
policy-map shape_example
  class class-default
    shape average 5m account user-defined 24
```

（注） **shape average** コマンドライン インターフェイスには、Bc（間隔あたりの持続または認定ビット数）およびBe（間隔あたりの超過ビット数）のオプションもあります。（これらのオプションは、IOS classic シェーピングのソフトウェア実装からの名残であり、ASR 1000 シリーズ アグリゲーション サービス ルータには影響を及ぼしません。）

ソフトウェアの実装では、処理のオーバーヘッドは、所定の間隔（通常はミリ秒）でのスケジューリングに関連する計算を実行することだけが可能であったことを意味します。

Bc の調整により、転送されたトラフィックのバースト性を犠牲にして、スケジューリングの頻度（そしてそれによる処理のオーバーヘッド）をさらに減らします。Cisco ASR 1000 シリーズ アグリゲーション サービス ルータでは、スケジューリング 決定は専用のハードウェアで実行されるため、（**頻繁に**）スケジューリング決定を行ってもパフォーマンスが低下することはありません。ハードウェアを最適化して、転送するストリームからバースト性を排除し、Bc または Be へのユーザー入力を排除しました。

Shape Peak

shape peak コマンドは ASR 1000 シリーズ アグリゲーション サービス ルータでサポートされますが、**shape average** コマンドを超える機能は提供していません。既存の IOS classic デバイスから ASR 1000 シリーズ アグリゲーション サービス ルータへ簡単に設定を移行できるようにサポートします。**shape peak** コマンドを使用すると、ルータは設定されたレート Bc を調べ、目標のシェーピング レートを計算します。このレートは、**show policy-map interface** コマンド出力に表示され、ASR 1000 シリーズ アグリゲーション サービス ルータは、ハードウェア ス

ケジュールエントリにプログラム設定されます。新しい設定を作成する場合は、**shape average** コマンドを使用する必要があります。

Bandwidth Remaining コマンド

The **bandwidth remaining** コマンドで、スケジュールエントリの超過ウェイトを設定するため、超過帯域幅のキューの割合が決定します。超過帯域幅は、**priority** または **bandwidth** コマンドによって別のキューに明示的に保証されず、保証されているキューによっても使用されない帯域幅として定義されていることを思い出してください。（超過ウェイトの詳細については、「スケジュール エントリのプログラム方法（39 ページ）」を参照してください。）超過帯域幅の共有を確定的な方法（初期状態によって完全に決まる動作）で分散すること により 、帯域幅の浪費を回避します。（帯域幅の共有の詳細については、「帯域幅キュー（60 ページ）」を参照してください。）

bandwidth remaining コマンドも、キューに帯域幅を保証する効果的な方法でもあります。超過帯域幅の共有のみを使用してすべての帯域幅を割り当てることは、完全に合理的であり、非常に一般的です。

bandwidth remaining コマンドには、**bandwidth remaining ratio** と **bandwidth remaining percent** の2つのバリアントがあります。いずれの場合も、スケジュールエントリに同じ超過帯域幅パラメータを設定します。2 つの形式における理論的根拠は、階層型ポリシーについて説明するときに理解ただし、物理インターフェイスに適用されたフラット ポリシーのコンテキストでは、どちらの形式を選択してもプロビジョニングが簡単になります。

（注） 両方のバリアントで（他のスケジューリング コマンドと同様に）**account** キーワードをサポートしています。

Bandwidth Remaining Ratio

bandwidth remaining ratio コマンドに関して最初に理解する必要がある点は、値を明示的に設定しない限り、すべての帯域幅キューのスケジュールエントリには1（'1'）の既定の超過ウェイト（Ex）が備わっているということです。（たとえば、作成時に、class-default キューのスケジュール エントリの Ex は 1 になります。）理解しやすい確定的既定値を持つことで、QoS スキームを設計する際のあいまいさが解消されます。

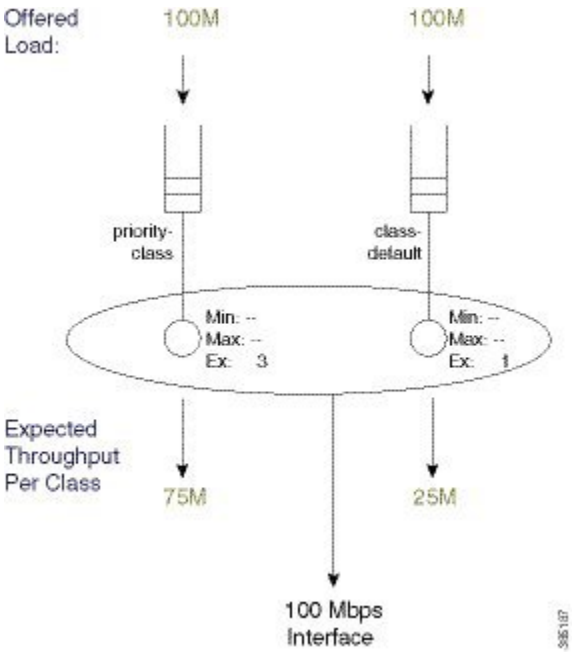
次のポリシーマップの例を考えてみましょう。

```
policy-map BRR-Example1
  class mission-critical
    bandwidth remaining ratio 3
```

このポリシーには、スケジューリング コマンドで明示的に作成する mission-critical クラス用と暗黙の class-default 用の 2 つのキューがあります。

では、このポリシーを 100 Mbps インターフェイスに適用し、各キューに 100 Mbps を提供してみましょう。スケジュール階層とクラスごとの予想スループットは、次のようになります。

図 *12 :* 比率によって明示的に割り当てられる帯域幅の分割



次に、 **shape** コマンドを使用して明示的にクラスを追加して、ポリシーを変更してみましょう。shape (class-map) コマンドのページで学んだように、**shape peak** コマンドでそのキューのスケジュール エントリの最大値を設定します。Ex は明示的に設定されていないため、デフォルトの 1 になります。

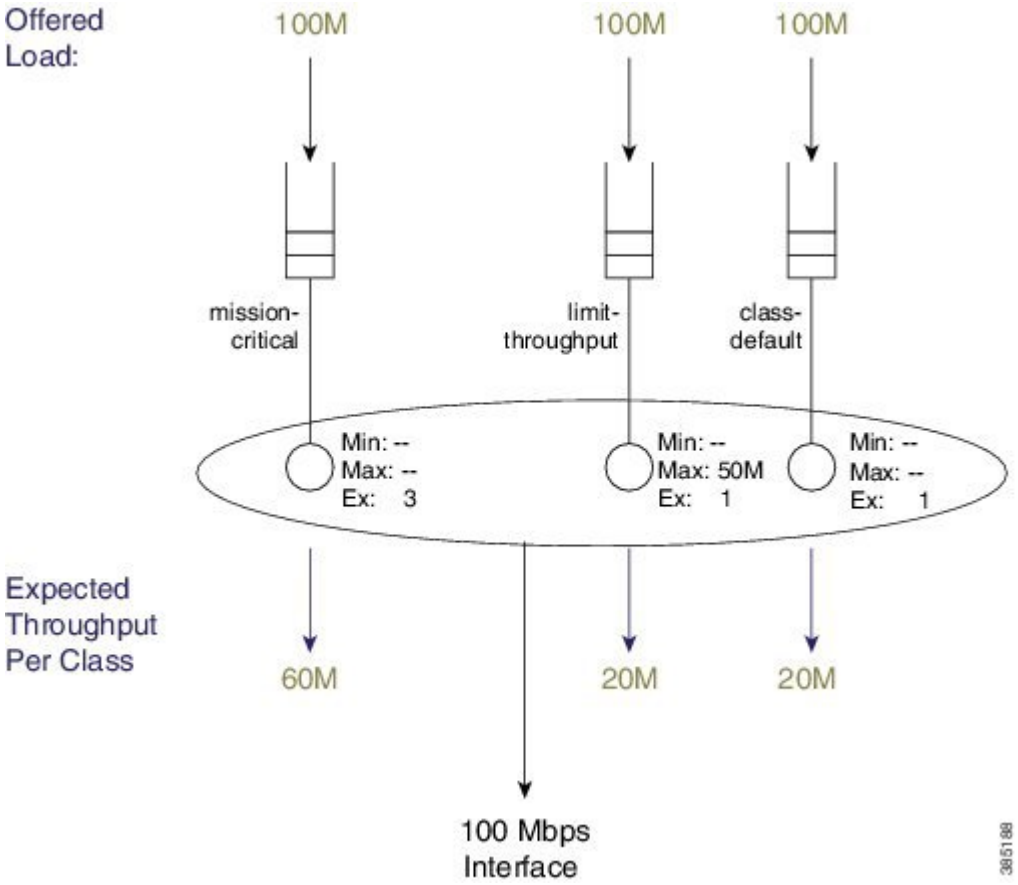
（注） 最大エントリは、キューに対する帯域幅の共有を保証するものではなく、単にそのキューの可能なスループットの上限値を設定します。

ポリシーは次のようになります。

```
policy-map BRR-Example1
  class mission-critical
    bandwidth remaining ratio 3
  class limit-throughput
    shape average 50m
```

このポリシーを 100 Mbps インターフェイスに適用し、各クラスに 100 Mbps を提供する場合、スケジュール階層と予想されるスループットは次のようになります。

図 *13 : bandwidth residual ratio* コマンドを使用した明示的クラスの超過ウェイトの変更



予想されるスループット（60M、20M、および20M）は、Ex値の比率（3、1、および1）を反映します。重要点は、**bandwidth remaining ratio** コマンドを使用して超過ウェイトを変更すると、明示的に変更するクラスのエントリのみが変更されることです。

帯域幅余剰比率は 1 から 1000 まで範囲があるため、異なるキューのサービス レート間で大きなバリアンスを獲得できます。

Bandwidth Remaining Percent

bandwidth remaining percent コマンドでも、帯域幅キューのスケジュールエントリの超過ウェイト（Ex）を変更することができます。明らかに、パーセントベースのスキームでは、すべての帯域幅キューにおける超過ウェイトの合計は100である必要があります。 これは、class-defaultと明示的に設定されていない他のすべてのキューの（明示的に割り当てられていない）割合を（同等に） 分配することによって実現します。

このコマンドの詳細については、bandwidth [remaining percent] コマンド ページを参照してください。

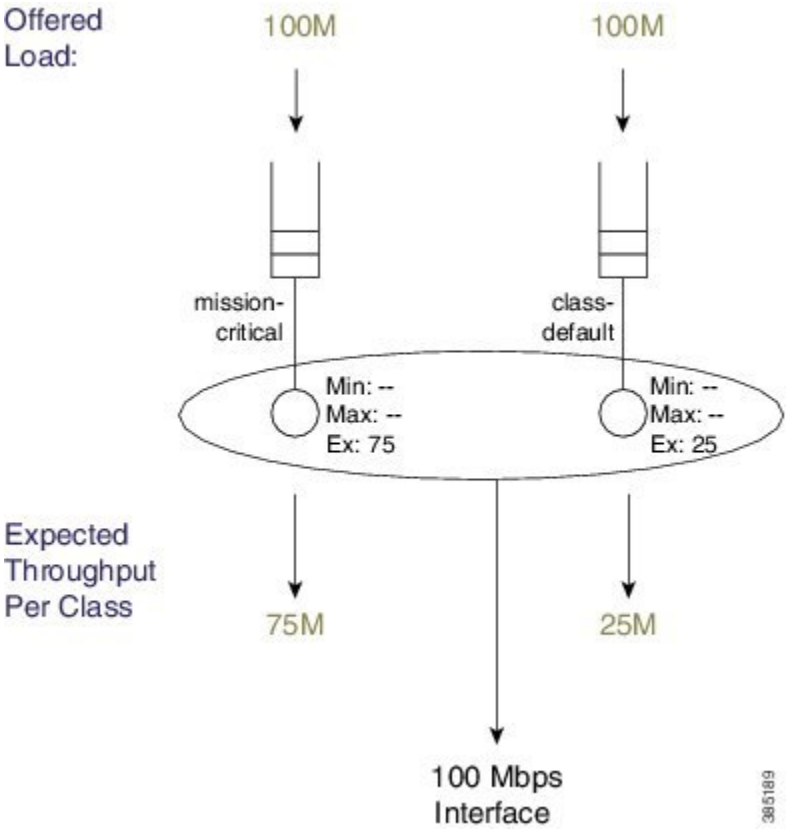
最も単純な例として、Bandwidth Remaining Ratio （65 ページ） の最初の例を考えましょう。

```
policy-map BRP-Example1
  class mission-critical
```

```
bandwidth remaining percent 75
```

スケジュール階層とクラスごとの予想スループットは、次のようになります。

図 *14* : パーセントで明示的に割り当てられた帯域幅の分割



class-default の超過ウェイトが明示的に設定されていませんが、（デフォルトで"1"から）変更されたことに注目してください。

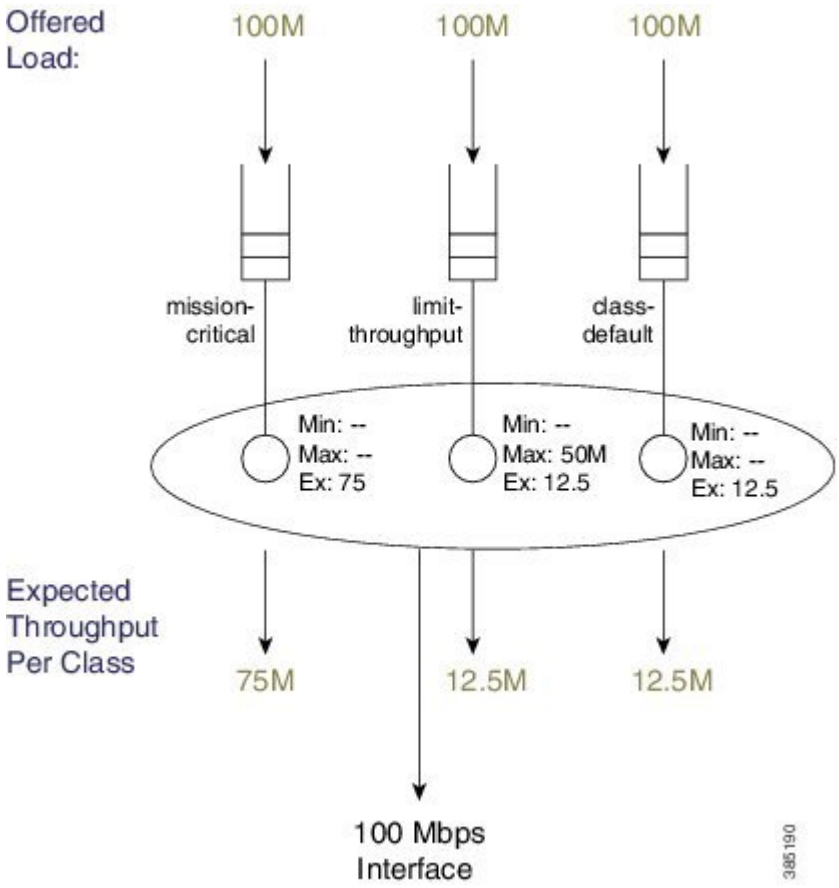
次に、残存帯域幅が明示的に設定されていないキューイングクラスを追加します。先ほどの説明にもあったように、シェーパーだけでクラスを追加します（「Bandwidth Remaining Ratio （65 ページ）」 の「比率によって明示的に割り当てられる帯域幅の分割」の図を参照してください）。

```
policy-map BRP-Example2
  class mission-critical
    bandwidth remaining percent 75
  class limit-throughput
    shape average 50m
```

この例では、class-default と明示的に割り当てられていないクラスにおける割合の分割に重点を置いています。

階層とスループットは次のようになります。

図 *15*: 追加されたシェーパーを使用して、*class-default* クラスと割り当てられていないクラス間で帯域幅の割合を分割する



2 パラメータ対 3 パラメータ スケジューリング

先ほどの説明にあったように、各帯域幅キューのスケジュール エントリには、キュー サービスを制御するための 3 つのパラメータ（Min、Max、Ex）があります。（スケジュール エントリのプログラム方法 （39 ページ） を参照）。そのため、ASR 1000 シリーズ アグリゲーション サービス ルータでは、スケジューラの実装を *3* つのパラメータによるスケジューラとして分類しています。

既存の IOS Classic の実装では、より単純な *2* つのパラメータによるスケジューラを提供します。Min と Ex の個別のエントリの代わりに、各スケジュール エントリには単一ウェイトのみとなります。また **bandwidth** または **bandwidth remaining** コマンドを使用したかどうかにかかわらず、同じ単一ウェイトを設定しました。違いを把握するために、**bandwidth** コマンドに焦点を当てた例を見てみましょう。

この例では、ポリシーマップは次のようになります。

```
policy-map bandwidth-example
  class bandwidth-class1
    bandwidth percent 5
```

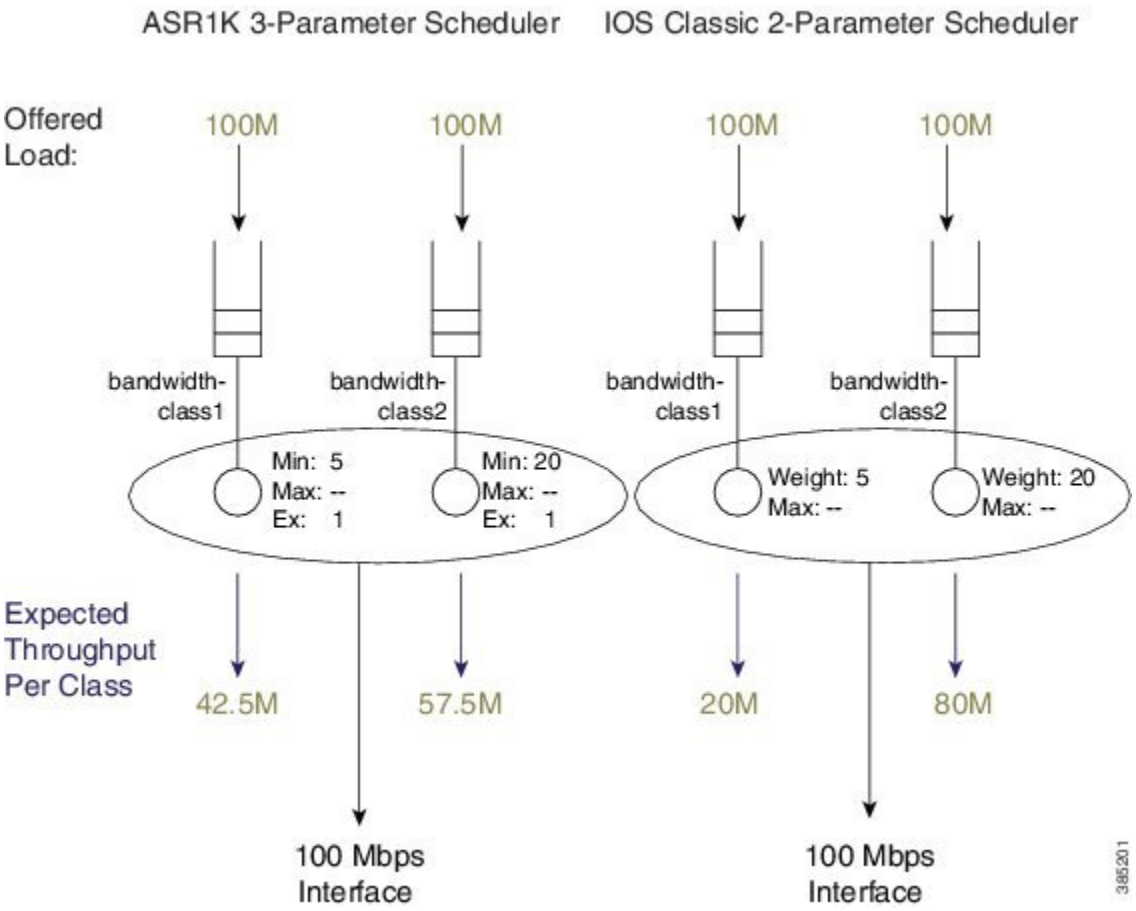
```
class bandwidth-class2
  bandwidth percent 20
```

次に、100 Mbpsインターフェイスに適用されたポリシーマップを検討し、各キューに100 Mbpsを提供します。次の図は、ASR 1000 シリーズ アグリゲーション サービス ルータ（左側）とIOS Classicイメージを実行しているルータ（例： Cisco 7200、右側）の両方で、スケジュール設定と予想スループットがどのように表示されるかを説明しています。

25 Mbps は ASR 1000 シリーズ アグリゲーション サービス ルータの 3 つのパラメータによるスケジューラに割り当てられ、最低帯域保証が遵守するために 25 Mbps を使用し、超過帯域幅が 75 Mbps 残ります。 この超過帯域幅は、各キューが受け取る既定の超過ウェイトである「1」に基づいて、2 つのキューで均等に共有されます。

この同じ設定を2つのパラメータによるスケジューラに適用すると、設定された帯域幅の値により、スケジュールエントリの単一ウェイトのパラメータが決定します。「Minスケジューリングと超過帯域幅の共有」という概念はここでは適用されません。 代わりに、各エントリでは、すべての帯域幅共有は、単一ウェイトのみに依存します。

図 *16 : IOS XE* を搭載した *ASR 1000* と *IOS Classic* を実行しているルータで同じ設定を実行



この例を見ると、IOS XE を実行している ASR 1000 シリーズ アグリゲーション サービス ルータで同じ設定をした場合、IOS のクラシック イメージを実行しているルータとは動作が大きく異なることがわかります。

（注） IOS Classic を実行しているルータと同一の動作を実現するには、超過帯域幅の共有のみを使用して設定を使用できます。IOS Classic 設定での **bandwidth percent** ステートメントを IOS XE での **bandwidth remaining percent** ステートメントに変更すると、既存の設定を簡単に移行できるようになります。

スケジュールのバースト性

スケジューリングにおけるバースト性の考えられる送信元は、パケットのバッチ処理およびスケジューラによる時間表記にあります。

パケットのバッチ処理

このソースは意図されたものであるため、問題を引き起こすことはありません。ハードウェアで実装されているように、スケジュールが 1 秒間に下すことができる決定の数を制限します。小さなパケットすべて、たとえば 64 バイトのフレームを送信する場合、10 Gbps のような高速インターフェイスでパケットごとに決定を下す場合、スケジュールはそれを維持するのに苦労する可能性があります。この負担を軽減するために、ハードウェアは小さなパケット（同じキューから最大約 512 バイト）をバッチ処理し、スケジューラにそれらを単一の決定として処理させます。

したがって、キューの先頭に 512 バイトのパケットが 1 つあれば、それを単一のパケットとしてスケジュールに送信します。反対に、5 つの 64 バイト パケットがキューの先頭にあった場合、スケジューラの観点から、それらのパケットは単一のパケットとしてバッチ処理します。つまり、5 つのパケットすべてを同時にキューからプルし、それらを 1 回のバーストとしてワイヤ上で転送します。 単一の MTU のサイズがバーストのサイズを大幅に超えると、後者はダウンストリームのバッファリングや他のキューのジッタに与える後の影響はわずかになります。

スケジューラによる時間表記

2 番目に考えられるバースト性の送信元は、ハードウェアのスケジュールがどのように時間を追跡しているかに起因します。同じポリシーマップ内に非常に小さいレート（たとえば 100K 以下）と非常に大きいレート（たとえば 100M 以上）を混在させると、高いレートで設定されたキューからのトラフィックのスケジューリングに予期しないバーストが発生する可能性があります。

リアルタイム スケジューリングを使用する場合 （**bandwidth** または **shape** コマンドのいずれかを使用して）、レートをビット/秒で指定します。 つまり、各スケジュール エントリには、リアルタイムの概念を持っている必要があり、サービスレートをそのエントリのリアルタイムと照らし合わせて監視する必要があります。時間の表記は、指定されたスケジューラのすべてのエントリにわたって一致している必要があります。

8 Kbps のシェーパー（8000 ビット/秒 = 1000 バイト/秒）について考えてみましょう。

64 バイトのパケットを送信することは、（64 バイトのパケット * 64/1000 =）64 ミリ秒ごとに 1 つのパケットを送信すること（と同等）になります。

1500 バイトのパケットを送信することは、（1500 バイトのパケット * 1500/1000 =）1.5 秒ごとに 1 つのパケットを送信すること（と同等）になります。

64 ミリ秒から始まる範囲を表記する必要があります。これを行うには、時間をカウントし、すべてのカウンタの増分が 10.5 ミリ秒のリアルタイムを表示するようにします。

今度は 10 Gbps のシェーパーについて考えてみましょう。 10.5 ミリ秒では、1500 バイトのパケットを 8750 個送信することが予想されます。

10,000,000,000 b/sec * .0105 sec = 105,000,000 ビット、つまり 105,000,000/8/1500、または 1500 バイト パケットを 8750 個

これは巨大なデータ バーストです。10.5 ミリ秒の増分で時間をカウントすると、時計が進む（進む）ときはいつでも、そのバーストを送信する必要があります。対照的に、.65 ミリ秒がリアルタイムを表示している場合、1500 バイトのパケットを 542 個送信されると予想されます（はるかに扱いやすい状況）。

時間の表記は、ポリシーマップ内で設定される最低レートによって決まります。次の表は、選択した時間とポリシーマップで設定されているレートの粒度を示します。（詳細は ESP-20 において正確ですが、ASR1K ハードウェアのすべてのバリアントに対して類似しているというだけです。）

表 *3:* 選択した時間の粒度と設定したレート

ポリシー マップのレート範囲	選択した時間の粒度
8K ～ 14K	10.5 ミリ秒
15K ～ 28K	5.2 ミリ秒
29K ～ 57K	2.6 ミリ秒
58K ～ 115K	1.3 ミリ秒
116K ～ 231K	0.65 ミリ秒
232K ～ 463K	0.33 ミリ秒
464K ～ 927K	0.16 ミリ秒
928K ～ 3094K	0.08 ミリ秒
3.1M ～ 6.1M	40 マイクロ秒
6.2M ～ 12.2M	20 マイクロ秒
12.3M ～ 24.6M	10 マイクロ秒

ポリシー マップのレート範囲	選択した時間の粒度
24.7M ～ 49.4M	5 マイクロ秒
49.5M ～ 99M	2.5 マイクロ秒
99.1M ～ 198M	1.3 マイクロ秒
198.1M ～ 396M	0.6 マイクロ秒
396.1M ～ 10G	0.3 マイクロ秒

表を見ると、ポリシーマップ内のすべてのレートが 116K 以上である場合、この時間の表記によって発生したバーストは 1 ミリ秒未満であるため、それほど重要ではないことがわかります。シェープまたは帯域幅レートを高速インターフェイスで 116K 未満に設定した場合は、意図していない結果が生じないようにする必要があります。（たとえば、ポリシーマップ内のすべてのレートが 29K ～ 57K の場合、この時間表記で発生するバーストは 2.6 ミリ秒になります。）結果として、ダウンストリーム デバイスにそのようなバーストを受信する上でのバッファリングが不十分な場合に起こるデバイスのドロップ、WRED ドロップ パケット、またはダウンストリームポリサーがバースト許容値を超えた場合に起こるパケットのドロップなどが発生する可能があります。

キューにおける最低保証サービス レート

どのキューについても、最低保証サービス レート（他のすべてのキューが輻輳している場合にキューが受け取るサービス）を計算できます。 前述のいくつかの例では、各キューにおける 100% の提供レートについて説明してきました。これらの例では、予想されるスループットが最低保証サービス レートです。 このレートにより、アプリケーションが深刻な輻輳の下でどのように動作するかを予測できる場合があります。つまり、ネットワークがシステム上過負荷となっているときに、アプリケーションが実行されることを期待できますか。

保証レートから計算できる特に有用な数値の1つとして、出力キューがいっぱいになった場合にパケットが受ける遅延があります。たとえば、オーバーサブスクライブ ビデオ キューについて考えてみましょう。そのポリシーマップは次のようになります。

```
policy-map min-service-rate-example
  class priority-class
    priority
    police cir 1m 1250
  class video
    bandwidth 1000
  class mission-critical
    bandwidth 2000
```

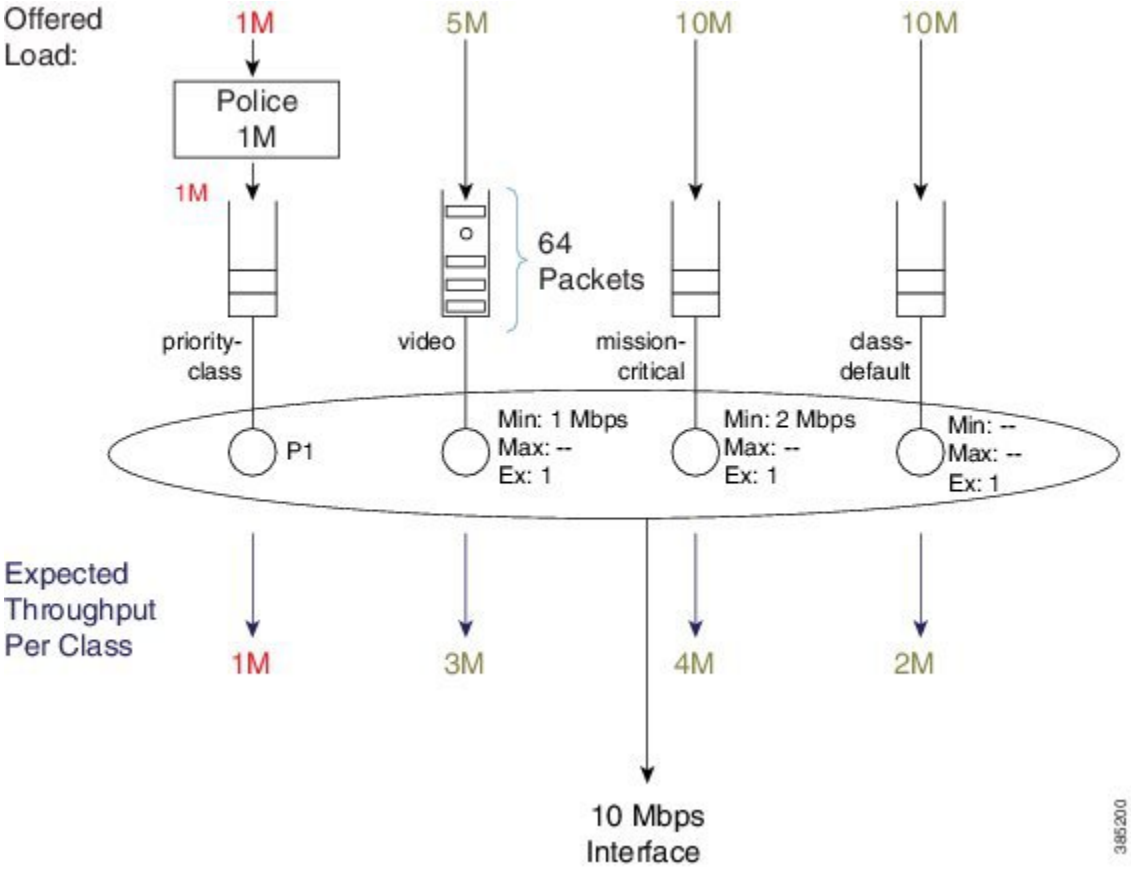
この例では、次の分析図から分かるように、ポリシーマップを10 Mbpsイーサネットインターフェイスに適用し、提供される負荷を各クラスに付加します。 （「スケジュール操作 （40 ページ）」のガイドラインに準拠します。）

Video クラスは、最低保証サービスレートである 3 Mbps（1 Mbps (**bandwidth** コマンドで設定された Min）+ 2 Mbps（常に保証されている超過帯域幅の割合) で処理されます。関連する計算をより詳しく見ましょう：

10 Mbps - 1 Mbps（P1 用）- 3 Mbps（ビデオおよびミッションクリティカル アプリケーションのための最低保証帯域）= 6 Mbps

6 Mbps の超過帯域幅（アカウンティング後の P1 および最低保証用）/3（すべてのキューにおいて 3 つに均等に割り当てる） = 2 Mbps

図 *17*：最低サービス レートと遅延の「エクスペリエンス」



最低保証サービス レートで、ビデオ キューに到着する遅延パケットの「エクスペリエンス」を計算できるようになります。デフォルトの 64 パケットのキュー制限を使用すると（オーバーサブスクリプションの場合、キューがいっぱいになり、64 パケットが含まれると予想されます）、新しいパケットはドロップされるか、キューの末尾に配置されます（パケットがビデオ キューから取り出された直後にパケットが到着した場合）。

このビデオ トラフィック用のキューを考えると、平均パケット サイズは約 1400 バイト（MPEG I フレームのサイズ）になると予想され、バッファリングされたデータ量として 716,800 ビットが生成されます。

64 パケット * 1400 バイト/パケット * 8 ビット/バイト = 716800 ビット

3 Mbps の最低レートを考えると、このキューを排出するために 239 ミリ秒が必要になります。

716800 ビット/3 Mbps = 0.239 秒（239 ミリ秒）

このように、最低保証サービス レートを設定すると、輻輳した状況でのアプリケーションの動作を予測することができます。

Pak プライオリティ

Pak プライオリティは、ネットワークの安定性にとって重要となる非常に重要な制御パケット(インターフェイス キープアライブ、BFD パケット、一部のルーティング プロトコル hello など）を保護するためのスキームを指定します。このセクションでは、これらのパケットを説明し、それらがどのようにスケジュールされるかを概説します。

（注） Pak_priority という名前は残念ながら、制御パケットがプライオリティ キューに（実際に）キューイングされず、混乱を招く可能性があります。

Pak_priority を使用すると、制御パケットの配信の保証を試行しますが、低遅延の保証はされません。制御パケットには、コントロール プレーンで最初に生成されたとき、pak_priority フラグがマークされます。このフラグはルータの外部には伝播されず、パケットを出力インターフェイスに送信するときに特別な処理が行われるようにするためにのみ使用されます。

IP カプセル化された制御パケットの DSCP を CS6 に設定して、それらが通過する必要があるネットワーク内の他のデバイスでそれらを保護していることを確認します。制御パケットを生成するルータでは、pak_priority 指定は、CS6 パケットに対して設定する保護よりもさらに強化された保護を指示します。

次の表は、内部 pak_priority フラグでマークされているパケットとプロトコルの一覧です。

pak_prority フラグでマークされているパケットおよびプロトコル

表 *4 : pak_priority* フラグでマークされた制御パケット

マークされるレベル	パケットとプロトコル
レイヤ 1 とレイヤ 2	
	ATM アドレス解決プロトコル否定応答（ARP NAK）
	ATM ARP 要求
	ATM ホスト ping 操作、管理、保守セル（OA&M）

マークされるレベル	パケットとプロトコル
	ATM 暫定ローカル管理インターフェイス（ILMI）
	ATM OA&M
	ATM ARP 応答
	Cisco Discovery Protocol
	ダイナミック トランキング プロトコル（DTP）
	イーサネット ループバック パケット
	フレーム リレー エンドツーエンド キープアライブ
	フレームリレー Inverse ARP
	フレーム リレー リンク アクセス手順（LAPF）
	フレーム リレー ローカル管理インターフェイス（LMI）
	ホット スタンバイ接続間制御パケット（HCCP）
	ハイレベル データ リンク制御（HDLC）キープアライブ
	リンク集約制御プロトコル（LACP）（802.3ad）
	ポート集約プロトコル（PAgP）
	PPP キープアライブ
	リンク制御プロトコル（LCP）メッセージ
	PPP LZS-DCP
	シリアル ライン アドレス解決プロトコル（SLARP）
	一部のマルチリンク ポイントツーポイント プロトコル（MLPP）制御パケット（LCP）
IPv4 レイヤ 3	
	プロトコル独立型マルチキャスト（PIM）hello

マークされるレベル	パケットとプロトコル
	内部ゲートウェイ ルーティング プロトコル（IGRP）hello
	OSPF hello
	EIGRP hello
	Intermediate System-to-Intermediate System（IS-IS）hello、完全シーケンス番号 PDU（CSNP）、PSNP、ラベル スイッチド パス（LSP）
	ESIS hello
	トリガーされたルーティング情報プロトコル（RIP）確認応答
	TDP および LDP hello
	リソース予約プロトコル（RSVP）
	一部の L2TP 制御パケット
	一部の L2F 制御パケット
	GRE IP キープアライブ
	IGRP CLNS
	双方向フォワーディング プロトコル（BFD）

pak_priority パケットの保護レベル

第 1 レベル

ポリサーまたは WRED は、この指定を持つパケットをドロップしません。

第 2 レベル

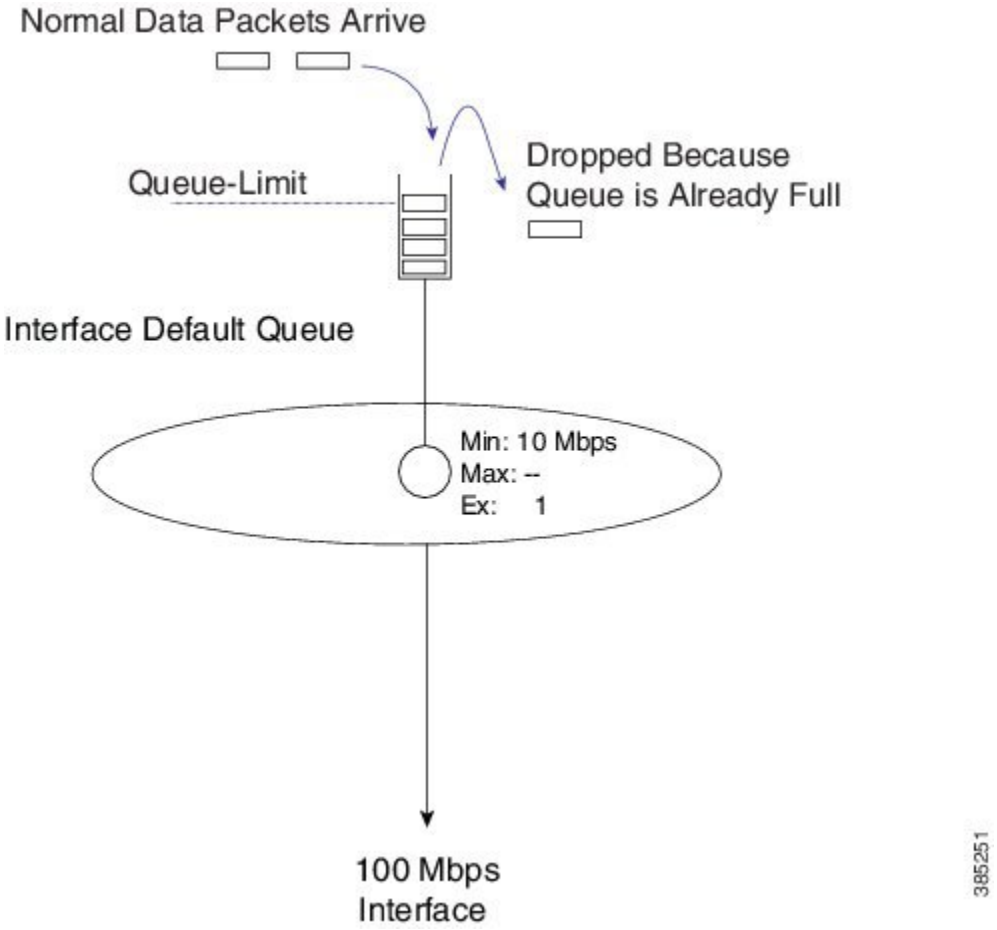
出力キューが既にいっぱいの場合でも、pak_priority パケットをエンキューします。これを理解するには、最初に、QoS が設定されていない物理インターフェイスを見てみましょう。まだキューが必要で、そのためそのキューからパケットをプルするスケジュールが必要となります。

インターフェイスに QoS が設定されていない場合、ファーストイン ファーストアウト（FIFO）キューが 1 つあります（インターフェイスのデフォルト キューと呼ばれます）。.（*class-default* キュー と混同しないでください）。

次の図では、キューがいっぱいになると通常のデータ パケットが到着します。スケジュール エントリには、標準の最小値と既定の Ex 値（インターフェイス レートの 10% および

1）が含まれていることに注意してください。キューが 1 つしか存在しないため、これらの値は無効になります。競合せずに、1 つのキューが使用可能なすべての帯域幅を受け取ります。

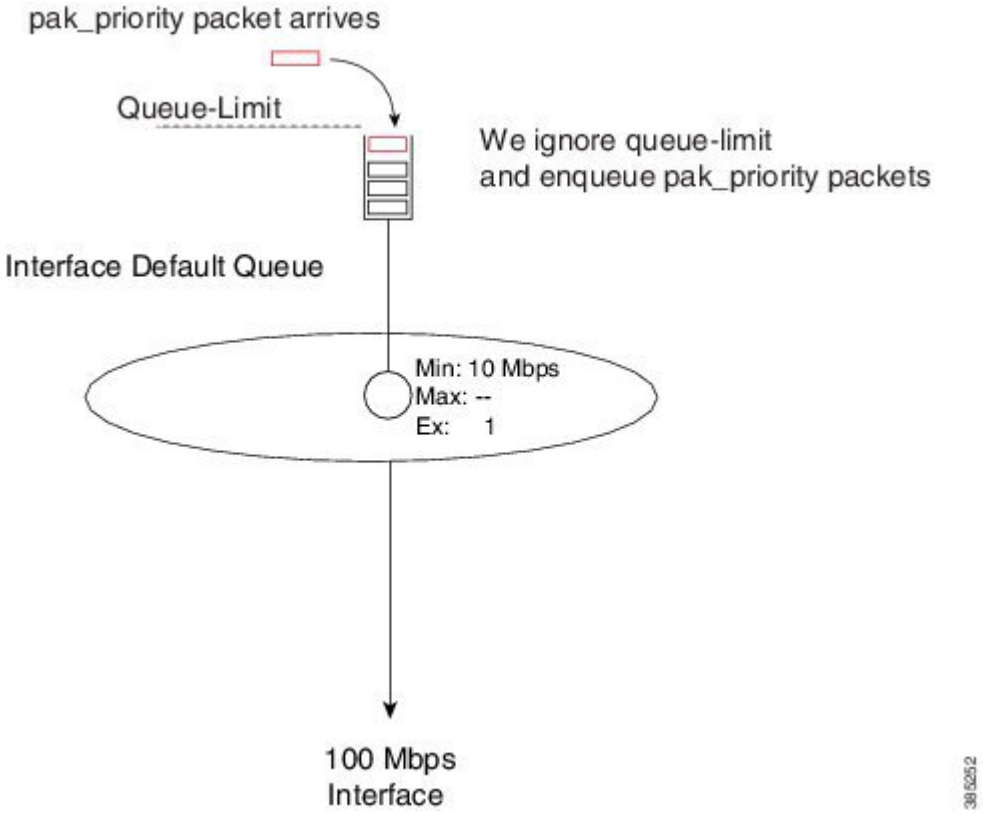
図 *18 :* キューがいっぱいになると通常のパケットが到着する



すべてのキューと同様に、キュー制限により、キューがいっぱいであるとみなされる前にバッファできるデータの量が決定します。

キューがいっぱいになると、pak_priority パケットが到着し、キュー制限を無視してエンキューします。パケットは、すべての先行パケットが送信されるまで待機する必要があります。ほとんどの ASR 1000 シリーズ アグリゲーション サービス ルータ プラットフォームでは、デフォルトのキュー制限は 50 ミリ秒です。したがって、50 ミリ秒 の pak_priority パケットを遅延させることがありますが、その配信を保証します。

図 *19 : pak_priority* パケットは、キューがいっぱいになったときに到着し、キュー制限は無視されます。

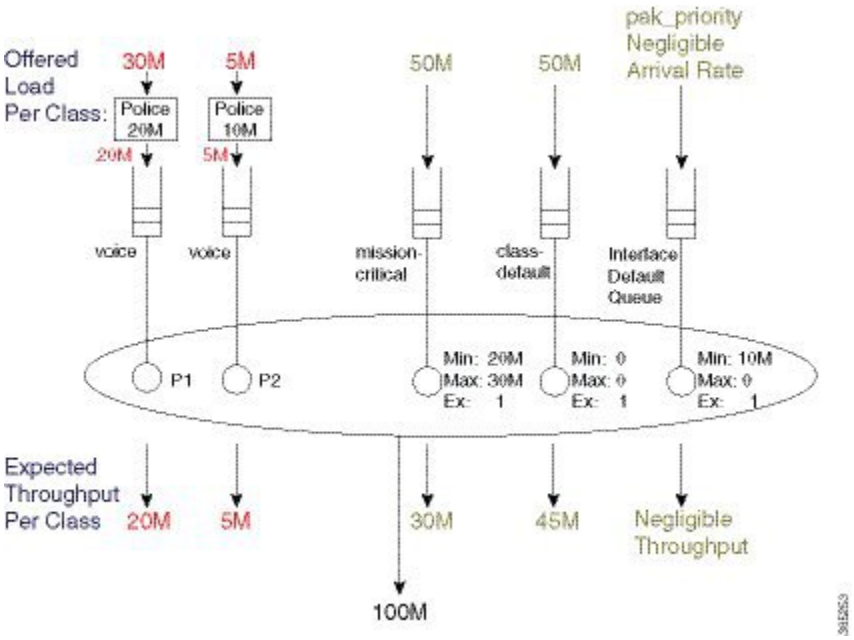


インターフェイスで QoS を設定する場合は、pak プライオリティに関する説明は少し変わってきます。それでは、この章（「シェーパーを使用したスケジュール操作 （43 ページ）」を参照）に出てくる一番最初の例の 1 つをもう一度考えてみましょう。

```
policy-map scheduling-example
  class voice
    priority level 1
    police 20m
  class video
    priority level 2
    police 10m
  class mission-critical
    bandwidth 20000
    shape average 30m
```

これは以前に説明していなかった点です。以下にあるように、ポリシーを設定するときに、インターフェイスのデフォルト キューは削除しません.

図 *20 :* 専用の *pak_priority* キューとしてのインターフェース デフォルト キューの転用



Pak_priority パケットは、インターフェイスのデフォルト キューにまだエンキューされています。基本的にキューは専用の pak_priority キューとして再利用されています。

図を見ると、Min 値の重要度（ライン レートの 10% に設定）が分かります。この値は、（Min サービスが設定された）他のキューによりこのサービスのキューが消費されることがないことを保証します。

しかし、ライン レートの 10% を Min と設定しますが、この帯域幅を消費することはありません。非常に少数の重要なパケットのみを pak_priority としてマークするため、到着するレートは無視できる程度です。他のキューの動作には影響しないことを理解した上で、Min を多めに設定しました。あまりにも多くのパケットを「pak_priority」とマークすると、この方式は機能しません

ルーティング プロトコルでは、Hello パケットをマークしますが、pak_priority によるルーティング アップデートはマークしません。したがって、CS6 パケット用の帯域幅キューを作成する必要があります。

Hello パケットはインターフェイスのデフォルト キューを通過し、ルーティング アップデートでは新しく作成された帯域幅キューが使用されます。

例外は BGP です。ここでは、Hello パケットを pak_priority としてマークしません。どうしてですか。BGP Hello パケットおよびアップデートでは、同じ TCP ストリームが共有されます。特別な処理を行うと、TCP パケットが順序どおりに届かなくなります。これらを消費することができないため、これでは利点がありません。

pak_priority パケットを（DSCP または他のフィールドと照合して）分類し、帯域幅キューに移動しようとしても、実行されません。パケットは引き続きインターフェイスのデフォルト キューに入れられます。 ただし、パケットを分類し、指定されたプライオリティ キューに移動することができます。

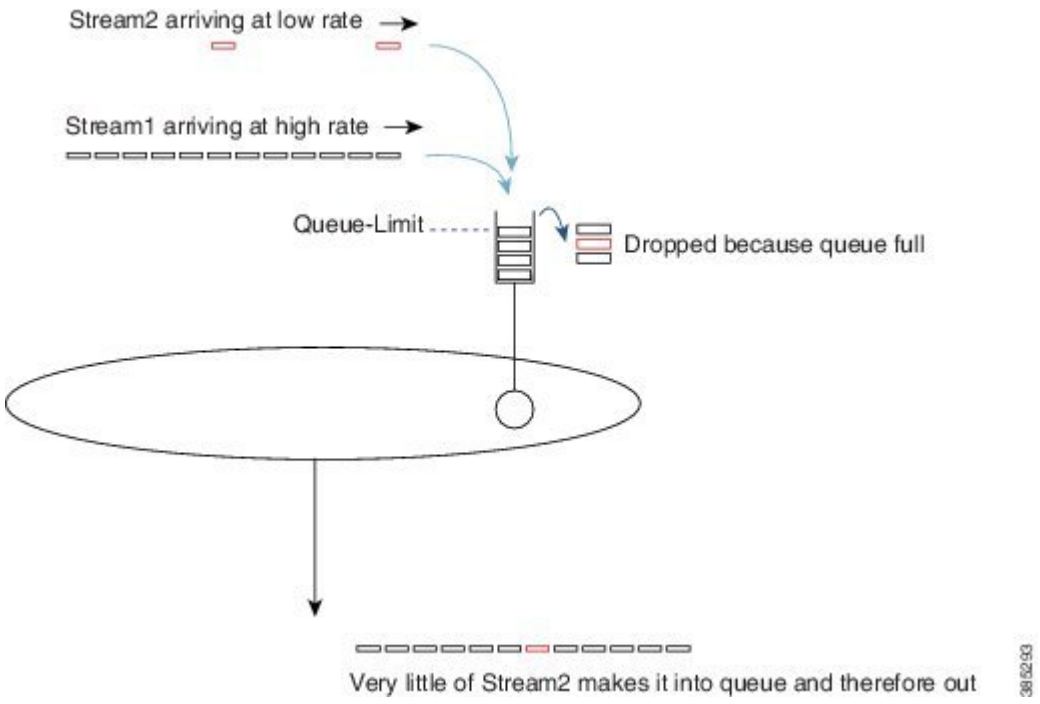
フローベース均等化キューイング

フローベースの均等化キューイングにより、ポリシーマップ内の同じクラスに属するフロー間の公平性を確保できます。これは、高速ストリームと競合するトラフィックの低速（よく動作する）ストリームを保護します。

個々のクラスがオーバーサブスクライブされている（提供レートがサービス レートを上回っている）場合、高レート フローが、低レート フローを持つサービスを消費してしまう可能性があります。これを理解するには、同じ物理キューをターゲットとする複数のストリームを考慮する必要があります。キューがいっぱいになると、パケットがさらにはドロップされます。スケジューラがキューからパケットを送信するたびに、キューの末尾に1つのスペースが開きます。次に到着するパケットのエンキューに成功しました。次の例を見ると、ストリーム 1（高レート ストリーム）からのパケットが次に到着する可能性が高いことがわかります。ストリームが公平に処理されていません。インタフェースに転送されるものは、キューに入れるためにどのパケットが管理するかによって決まります。

高レート ストリームがクラス帯域幅を不均等に割り当てられるだけでなく、低レート ストリームの遅延にも影響します。. 正常にエンキューされたパケットは常に完全なキューの末尾にあるため、他のすべてのパケットが送信されるまで待機してから転送する必要があります。

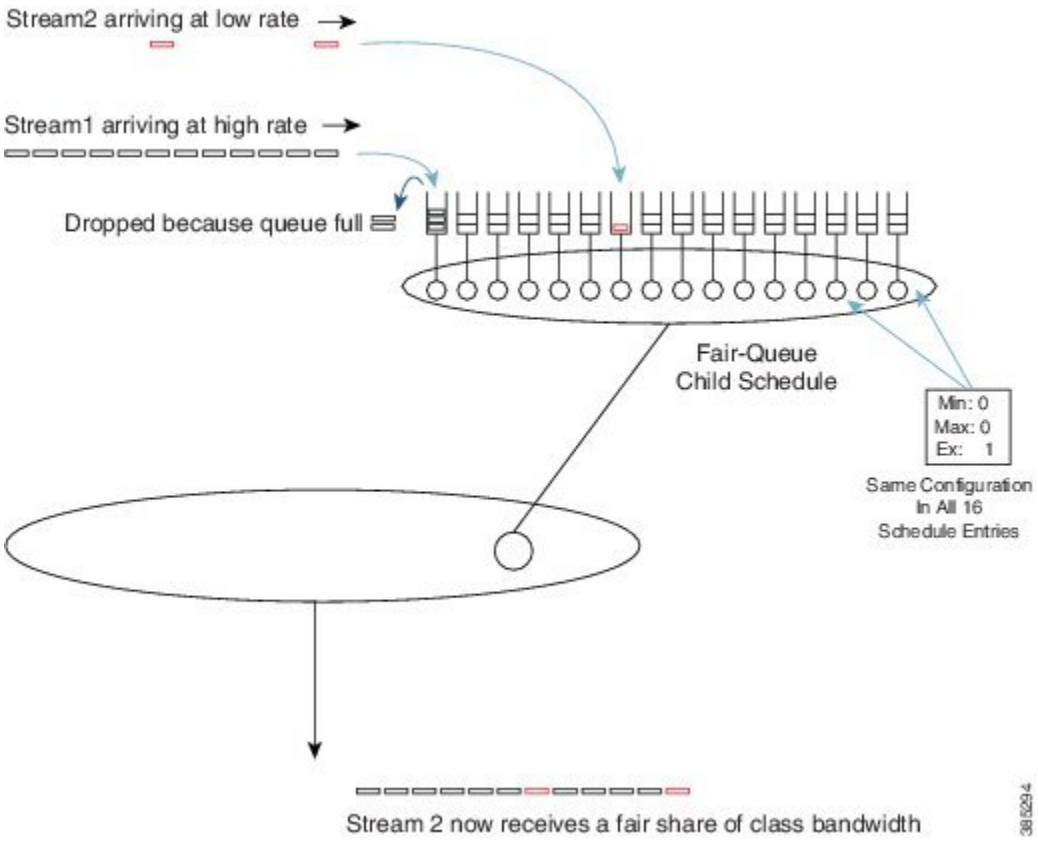
図 *21* : フローベースの均等化キューイングを使用しない場合の遅延



フローベースの均等化キューイングにより、この問題を軽減できます。この **fair-queue** コマンドを発行すると、その 1 つのクラスに対して 16 個のキューを作成するようにルータに指示しますが、これは階層スケジューリングの単純なスキームです。 .

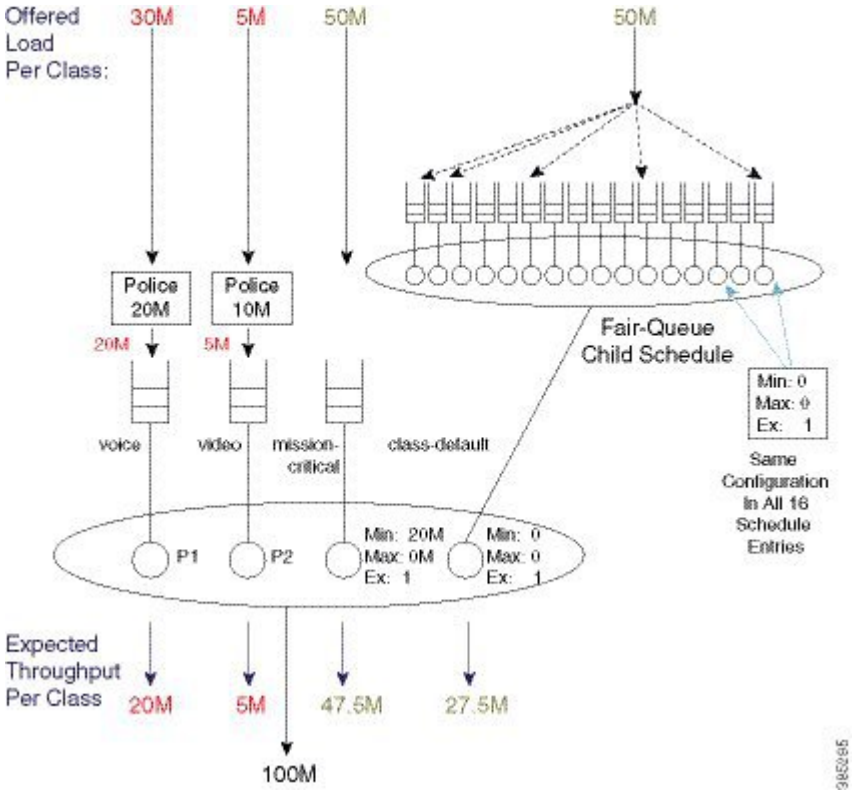
均等化キューの子スケジュールをプレソーターと見なしてください。同じクラスをターゲットとする複数のストリーム間でソーティングと均等化が実現します。

図 *22*：フローベースの均等化キューイングによって提供されるプレソートと均等化




```
policy-map fair-queue-example
  class voice
    priority level 1
    police 20m
  class video
    priority level 2
    police 10m
  class mission-critical
    bandwidth 20000
  class class-default
    fair-queue
```

図 *23 :* フローベースの均等化キューイングによるパケット フロー



確認

show policy-map interface インターフェイス コマンドを使用して、スケジューリングの操作を確認できます。このコマンドによって、長期間の傾向と、設定されたポリシーの完全なビューが表示されます。

データ プレーンは 10 秒ごとにコントロール プレーンに統計情報を送信し、コントロール プレーンは10秒ごとに独自の統計情報をリフレッシュします。これは、**show policy-map interface** コマンドの出力値が 10 秒ごとに更新されることを意味します。現在のキューの深さなど、瞬時の状態を表すカウンタの中には、あまり役に立たないものもあります。 真の瞬時の状態の情報が必要な場合は、ハードウェア カウンタを直接確認します。

次の設定は、**show policy-map interface** インターフェイス コマンドの例です。

```
policy-map show_policy-example
 class voice
  priority level 1
  police cir percent 10 bc 5 ms
 class video
   priority level 2
  police cir percent 20 bc 10 ms
 class critical-data
  bandwidth percent 50
```

このポリシーには、明示的に設定された 3 つのクラスと暗黙の class-default を含む 4 つのクラスがあります。

show policy-map interface コマンドからの出力は、設定されたポリシーを反映し、設定された各クラスのセクションがあります。各クラス内の出力は、分類セクションと設定済みアクションごとのセクションで一貫して編成されています。

プライオリティ クラスのキューイング情報は、そのクラスの他の機能（ポリサー）とは別に表示されることに注意してください。これは、複数のプライオリティ クラスが同じキューにマップされる可能性があるためです。

次に、**show policy-map interface** コマンドの出力例を示します。出力データが連続した 1 つの集合体となっていますが、出力の構造を強調するためにセクションに分割します。

`Device#show policy-map interface g1/0/4`
` GigabitEthernet1/0/4`
 	
`  Service-policy output: show_policy-example`
 	
`queue stats for all priority classes:`
`      Queueing`
`      priority level 1`
`      queue limit 512 packets`
`      (queue depth/total drops/no-buffer drops) 0/0/0`
`      (pkts output/bytes output) 39012/58518000`
	このセクションには、プライオリティ レベル 1 のキューのキュー情報が表示されています。

`    queue stats for all priority classes:`
`      Queueing`
`      priority level 2`
`      queue limit 512 packets`
`      (queue depth/total drops/no-buffer drops) 0/0/0`
`      (pkts output/bytes output) 61122/91683000`
	このセクションには、プライオリティ レベル 2 のキューのキュー情報が表示されています。

`Class-map: voice (match-all)`
`      39012 packets, 58518000 bytes`
`     5 minute offered rate 672000 bps, drop rate 0000 bps`
`      Match:  dscp ef (46)`
	このセクションでは、voice という名前のクラスの統計データが表示されています。
	
	最初の行に表示されているのは、分類統計データです。

Priority: Strict, b/w exceed drops: 0 Priority Level: 1 police: cir 10 %, bc 5 cir 100000000 bps, bc 62500 bytes conformed 39012 packets, 58518000 bytes; actions: transmit exceeded 0 packets, 0 bytes; actions: drop conformed 672000 bps, exceeded 0000 bps	プライオリティ レベルは、このクラスによって使用される上記のプライオリティ キューを示します。 キューアドミッション コントロールに使用されるポリサーの統計情報もここに表示されます。
Class-map: video (match-all) 1376985 packets, 2065477500 bytes 5 minute offered rate 9171000 bps, drop rate 0000 bps Match: dscp af41 (34)	これは、video というクラスのセクションの開始部分です。 分類統計データと基準が最初に表示されます。
police: cir 20 %, bc 10 cir 200000000 bps, bc 250000 bytes conformed 1381399 packets, 2072098500 bytes; actions: transmit exceeded 0 packets, 0 bytes; actions: drop conformed 9288000 bps, exceeded 0000 bps Priority: Strict, b/w exceed drops: 0 Priority Level: 2	このセクションでは、video という名前のクラスで設定されたアクションが表示されています。 キューアドミッション コントロール ポリサーの統計データです。 プライオリティ レベルは、このクラスからのパケットが、上記に示しプライオリティ レベル 2 のキューにエンキューされることを示します。
Class-map: critical-data (match-all) 45310 packets, 67965000 bytes 5 minute offered rate 719000 bps, drop rate 0000 bps Match: dscp af11 (10)	これは「critical-data」というクラスのセクションの開始部分です。 通常通り、分類統計データと基準が最初に表示されます。
Queueing queue limit 2083 packets (queue depth/total drops/no-buffer drops) 0/0/0 (pkts output/bytes output) 45310/67965000 bandwidth 50% (500000 kbps)	このクラスには帯域幅アクションがあるため、このクラスに対してキューが作成されます。 このセクションでは、キューに関連する設計と統計データが表示されています。

Class-map: class-default (match-any) 51513 packets, 77222561 bytes 5 minute offered rate 194000 bps, drop rate 0000 bps Match: any	これは class-default のセクションの開始部分で、すべてのポリシーに存在する暗黙のクラスです。 通常通り、最初にこのクラスに属するとみなされるパケットの統計データが表示されています。
queue limit 4166 packets (queue depth/total drops/no-buffer drops) 0/0/0 (pkts output/bytes output) 1371790/2057638061	このセクションでは、class-default のキュー情報が表示されています。

前述のように、**show policy-map interface** コマンドは 10 秒ごとにデータ プレーンから更新を受け取ります。

また、システムの動作をリアルタイムで表示するために、直接データ プレーンを確認することもできます。このコマンドでは、データ プレーンが予想どおりにプログラムされていることを確認することもできます。

パケットエンキューまたはドロップされたパケットなどの長期間イベントのカウンタは、情報がコントロール プレーンにプッシュされるたびに、10 秒ごとにクリアされます。

おそらく、データ プレーンの最も有用なカウンタは、キュー深度における即時データです。このコマンドは発行するたびに読み取られるため、キューが輻輳しているかどうか、輻輳が持続しているか、バースト動作が発生しているかどうかをリアルタイムで把握できます。

show platform hardware qfp active feature qos interface インターフェイスは、ハードウェア データ プレーンの QoS 設定と統計データを表示するコマンドです。

次に、上記の設定と **show policy-map interface** の例に対応するコマンドからの出力例を次に示します。

コマンドの出力が、ポリシーマップの構造と設定された各クラスのセクションと共に反映していることがわかります。

Device#show platform hardware qfp active feature qos interface gig1/0/4 Interface: GigabitEthernet1/0/4, QFP interface: 11 Direction: Output Hierarchy level: 0 Policy name: show_policy-example	

<pre>Class name: voice, Policy name: show_policy-example
Police:
cir: 100096000 bps, bc: 63488 bytes
pir: 0 bps, be: 0 bytes
rate mode: Single Rate Mode
conformed: 0 packets, 0 bytes; actions:
transmit
exceeded: 0 packets, 0 bytes; actions:
drop
violated: 0 packets, 0 bytes; actions:
drop
color aware: No
green_qos_group: 0, yellow_qos_group: 0
overhead accounting: disabled
overhead value: 0, overhead atm: No</pre>
	voiceという名前のクラスのセクションの開始部分です。
	ポリサー設定と 10 秒間隔で更新される統計データです。

<pre>Queue: QID: 175 (0xaf)
bandwidth (cfg) : 0
, bandwidth (hw) : 0
shape (cfg) : 0
, shape (hw) : 0
prio level (cfg) : 1
, prio level (hw) : 0
limit (pkts) : 512
drop policy: tail-drop
Statistics:
depth (pkts) : 0
tail drops (bytes): 0
, (packets) : 0
total enqs (bytes): 0
, (packets) : 0
licensed throughput oversubscription drops:
(bytes): 0
, (packets) : 0
Schedule: (SID:0x258)
Schedule FCID : 16
bandwidth (cfg) : 1050 Mbps
, bandwidth (hw) : 1050.01 Mbps
 	
shape (cfg) : 1050 Mbps
, shape (hw) : 1050.01 Mbps</pre>
	voice クラスのキュー情報
	これは、キュー深度を表す即時データで、非常に便利です。

Class name: class-default, Policy name: show_policy-example Queue: QID: 176 (0xb0) bandwidth (cfg) : 0 , bandwidth (hw) : 0 shape (cfg) : 0 , shape (hw) : 0 prio level (cfg) : 0 , prio level (hw) : n/a limit (pkts) : 4166 drop policy: tail-drop Statistics: depth (pkts) : 0 tail drops (bytes): 0 , (packets) : 0 total enqs (bytes): 3420000 , (packets) : 2280 licensed throughput oversubscription drops: (bytes): 0 , (packets) : 0	class-default のセクションの開始部分です。 このクラスのキュー情報です。 10 秒間隔で更新される即時深度と統計データです。
Class name: video, Policy name: show_policy-example Police: cir: 200064000 bps, bc: 253952 bytes pir: 0 bps, be: 0 bytes rate mode: Single Rate Mode conformed: 0 packets, 0 bytes; actions: transmit exceeded: 0 packets, 0 bytes; actions: drop violated: 0 packets, 0 bytes; actions: drop color aware: No green_qos_group: 0, yellow_qos_group: 0 overhead accounting: disabled overhead value: 0, overhead atm: No	video という名前のクラスのセクションの開始部分です。 アドミッション コントロール ポリサーの設定と統計データが最初に表示されます。

Queue: QID: 178 (0xb2) bandwidth (cfg) : 0 , bandwidth (hw) : 0 shape (cfg) : 0 , shape (hw) : 0 prio level (cfg) : 2 , prio level (hw) : 1280 limit (pkts) : 512 drop policy: tail-drop Statistics: depth (pkts) : 0 tail drops (bytes): 0 , (packets) : 0 total enqs (bytes): 0 , (packets) : 0 licensed throughput oversubscription drops: (bytes): 0 , (packets) : 0 Schedule: (SID:0x258) Schedule FCID : 16 bandwidth (cfg) : 1050 Mbps , bandwidth (hw) : 1050.01 Mbps shape (cfg) : 1050 Mbps , shape (hw) : 1050.01 Mbps	video クラスのキュー情報 10 秒間隔で更新される即時深度と統計データです。
Class name: critical-data, Policy name: show_policy-example Queue: QID: 177 (0xb1) bandwidth (cfg) : 500000000 , bandwidth (hw) : 500000000 shape (cfg) : 0 , shape (hw) : 0 prio level (cfg) : 0 , prio level (hw) : n/a limit (pkts) : 2083 drop policy: tail-drop Statistics: depth (pkts) : 0 tail drops (bytes): 0 , (packets) : 0 total enqs (bytes): 0 , (packets) : 0 licensed throughput oversubscription drops: (bytes): 0 , (packets) : 0	これは「critical-data」というクラスのセクションの開始部分です。 10 秒間隔で更新される即時キュー深度と統計データです。

コマンド リファレンス

アカウント

Account とは、独立したコマンドではなく、ユーザがそのコマンドのオーバーヘッド アカウンティングを指定できるようにするスケジューリング コマンドの拡張機能です。この章では、各スケジュール コマンドでのレプリケーションを回避するために account について説明します。

構文の説明：

スケジューリング長に加算または減算されるユーザ定義のバイト数を設定するには、次のようにします。

[**no**] **shape** | **bandwidth** *rate* **account user-defined** *value* [**atm**]

ダウンストリーム デバイスのカプセル化を指定し、オーバーヘッド アカウンティングの調整を自動的に計算するには、次のようにします。

[**no**] **shape** | **bandwidth** *rate* **account dot1q** | **qing** *encapsulation*

コマンド デフォルト：

デフォルトでは、レイヤ 3 データグラムとレイヤ 2 ヘッダがスケジューリング計算に含まれます。

使用上のガイドライン：

アカウント オプションを、ポリシーマップのスケジュール アクションを含む 1 つのクラスで使用する場合、スケジューリング アクションを含むすべてのクラスで、同じ値を持つアカウント コマンドを使用する必要があります。同様に、階層型ポリシーマップでは、ポリシーの各レベルで同じアカウント オプションを設定する必要があります。

帯域幅

bandwidth コマンドは、クラスへの最小限サービス レートを保証するために使用されます。

構文の説明：

Kbps で設定するには：

[**no**] **bandwidth** *rate* [**account** *account options*]

表示帯域幅の割合として設定するには、次のようになります。

[**no**] **bandwidth percent** *value* [**account** *account options*]

コマンド デフォルト：

デフォルトでは、キューのスケジュール エントリに設定されている最小帯域幅の値はありません。既定の超過ウェイトにより、最小限サービスが保証されることに注意してください。

使用上のガイドライン：

この **bandwidth** コマンドは、最小帯域幅の要件がわかっているアプリケーションに対して役立ちます。

帯域幅レートは 8Kbps の増分で設定でき、ASR1K はテストされ、それらのレートの 1% 以内の正確さを達成しています。

bandwidth コマンドは、リーフ スケジュール（レイヤー スケジュール クラス）でのみサポートされています。親ポリシーで帯域幅を割り当てるい場合は、**bandwidth remaining** コマンドを使用することができます。

IOS Classic プラットフォーム（2 つのパラメータによるスケジューラ）のスケジューリング動作をレプリケーションしたい場合は、設定で、すべての **bandwidth percent** 値コマンドを **bandwidth remaining percent** 値コマンドで置き換えます。

Bandwidth remaining

bandwidth remaining コマンドは、クラス間の超過帯域幅を割り当てるために使用されます。これは、単純なウェイトとして、または使用可能な帯域幅の割合として設定できます。

構文の説明：

単純なウェイトとして設定するには：

[**no**] **bandwidth remaining ratio** *value* [**account** *account options*]

割合として設定するには：

[**no**] **bandwidth remaining percent** *value* [**account** *account options*]

コマンド デフォルト：

デフォルトでは、すべての帯域幅スケジュール エントリは、リーフ スケジュールまたは親スケジュールのいずれでも、超過ウェイト 1 で設定されます。これは、そのクラスに設定されている **bandwidth remaining ratio** *1* に相当します。

使用上のガイドライン：

残存帯域幅を、1 ー 1000 の値をサポートするウェイトとして設定します。これにより、割合オプションを使用するよりも詳細な残存帯域の割り当てが可能になります。

bandwidth remaining percent 値を構成すると、2 つのパラメータによるスケジューラを使用する IOS Classic と同様の動作を生成します。

親ポリシーにシェーパー/子ポリシー上のキュー（親には class default のみ）を使用して、**bandwidth remaining ratio** 値を使用し、親ポリシーが適用されている論理インターフェイス間で帯域幅を割り当てる必要があります

Fair-Queue

fair-queue コマンドは、帯域幅キューとして設定されたクラスでフローベースの均等化キューイングを設定するために使用します。

構文の説明：

fair-queue

コマンド デフォルト：

デフォルトでは、1 つの fifo キューが各帯域幅クラスに対して設定されます。

使用上のガイドライン：

フローベースの均等化キューイングは、貪欲な単一のフローがクラスに割り当てられたすべての帯域幅を消費できないようにするために使用します。

特定のフローからのすべてのパケットは、同じフローのキューにハッシュされます。

フローキューイングは、トンネルインターフェイスに適用されたポリシーでは設定することはできません。すべてのパケットには同じ外部ヘッダーがあり、すべてのパケットは同じフローのキューにハッシュされるため、機能が無効になります。

優先度

priority コマンドはトラフィックのクラスに低遅延と低ジッターを提供するために使用されます。

構文の説明：

絶対プライオリティ キューを設定するには（注：明示的なポリサーを使用する必要があります）：

[**no**] **priority**

マルチレベル プライオリティ キューイングを使用して絶対プライオリティ キューを設定するには（注：明示的なポリサーで使用する必要があります）：

[**no**] **priority level** *1* | *2*

条件付きポリサーを使用してプライオリティ キューを設定するには：

[**no**] **priority** *rate in kbps*[**burst in bytes**]

または

[**no**]**priority percent** *rate* [**burst in bytes**]

条件付きポリサーを使用してマルチレベル プライオリティ キューイングを設定するには：

[**no**]**priority level** *1* | *2 rate in kpbs* [**burst in bytes**]

または

[**no**] **priority level** *1* | *2* **percent***rate* [**burst in bytes**]

コマンド デフォルト：

デフォルトでは、キューはプライオリティ処理の設定はされていません。

使用上のガイドライン：

プライオリティキューは、他のクラスのサービスのキューが不足することがないように、キューアドミッションコントロール（明示的なポリサーまたは条件付きポリサー）で使用する必要があります。

ポリサー適合バーストは、キュー内のアプリケーションに適した値に設定する必要があります。次に、設定例を示します。

```
policy-map always_on_policer_burst_example
```

```
class voice
  priority
  police cir 2000000 1250
```

プライオリティを階層型ポリシーの親に設定する必要はありません。プライオリティ伝播が、リーフ スケジュールによってプライオリティとしてマークされたパケットを、スケジューリング階層を通して優先的に処理されるようにします。

形状

shape コマンドを使用して、キューを処理できる最大レートを設定します。シェーパーの設定により、クラスへのスループットが保証されることはなく、クラスが処理されるレートに上限値が設定されるだけです。

構文の説明：

[**no**] **shape average** *rate* [**unit**] [**confirming burst**] [**excess burst**] [**account** *options*]

または

[**no**] **shape average percent** レート [**confirming burst**] [**excess burst**] [**account** オプション]

コマンド デフォルト：

デフォルトでは、帯域幅キューのスケジュールエントリに設定されている最大レートはありません。

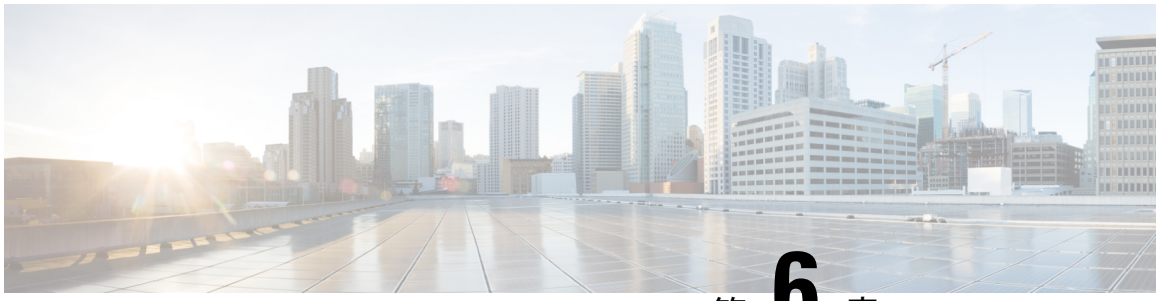
使用上のガイドライン：

shape コマンドは、トラフィックがリモート サイトに送信されるレートを制限する上で、親ポリシーで最も一般的に使用されます。

親ポリシーで使用される場合、すべてのトラフィック（プライオリティと帯域幅）に対してシェープ レートが適用されます。

shape コマンドには、適合および超過バースト サイズのオプションがありますが、これらの値は XE プラットフォームには影響しません。ASR1K プラットフォームでのハードウェア スケジューリングにより、バースト パラメータを最適化する必要はなくなります。

shape コマンドにより、クラスが処理される最大レートが適用されますが、このコマンド自体はそのクラスのスループットを保証しません。**bandwidth remaining** コマンドを shape コマンドと共に使用して、スループットを保証することができます。



第 6 章

QoS 階層型スケジューリング

この章では、スケジューリングの章で説明したように、コマンドとそのセマンティクスをさまざまな方法で組み合わせて、より複雑な結果を達成する方法を説明します。

複雑なスケジューリング階層を設定するには、階層型ポリシーマップと論理インターフェイスに適用されたポリシーマップの2つの異なるアプローチがあります。ここでは、どちらの方法で同じ結果が得られるかを示し、各アプローチの相対的な利点を説明します。

階層型スケジュールについて

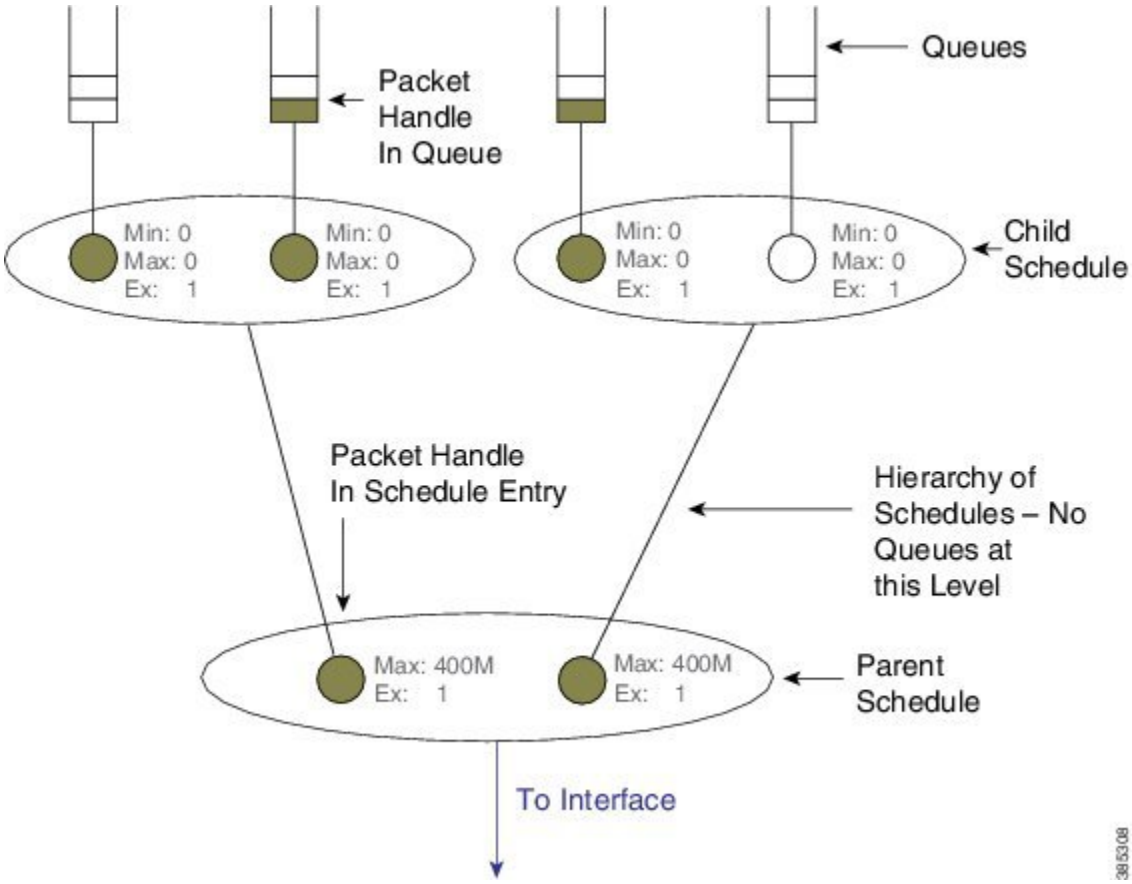
定義

スケジュールの役割と、そのエントリの子がどのように処理されるかに関する情報（パケット処理、クラス キューなど）をスケジュール エントリにどのように格納するかを理解していることを前提としています（スケジュールの章の「定義 （37 ページ）」を参照）。ここでは、その議論を基にしていきます。

ここで示した内容と前の章の根本的な違いは、子スケジュールにあり、キューまたは別のスケジュールである可能性があります。階層型スケジューリングにより、複数のレイヤで帯域幅を共有し、複雑な構造を構築できます。

次の図は、基本的な階層型スケジューリング構造を示しています。

図 *24 :* 階層型スケジューリングの定義



図から最初に気付くことは、キューの階層ではなく、スケジュールの階層を実装していることです。これは、キューが階層のリーフ レイヤのみに存在し、パケット処理（パケット表記ビークル）がキューからキューに移動することはないことを意味します。代わりに、1 つのパケット処理が親スケジュール エントリ（パケットが送信を待機していることを前提とした場合）にロードされます。

スケジューリング階層を詳述する場合、スケジュールを親または子（または実際には孫）として記述します。これらの記述は相対的なものです。親スケジュールは、階層のルートに一段近い（インタフェースに近い）スケジュールです。 スケジュールの子は、スケジュールまたはキューのいずれかになります。スケジュールをリーフ スケジュールまたは非リーフ スケジュールと呼ぶこともあります。 リーフ スケジュール には、子としてのキューのみがあります。非リーフ スケジュールには、子供として少なくとも 1 つのスケジュールがあります。

図を見ると、親スケジュール（非リーフ）のスケジュール エントリには、スケジュール エントリごとに2つのパラメータしかしかないことがわかります。最小帯域幅パラメータはリーフ スケジュールでのみサポートされ、非リーフ スケジュールではサポートされません。

スケジューリングの決定：ルートからリーフへ

次の一連の図は、階層内のスケジュールがどのように協調して機能し、さらにインターフェイスを介して送信する次のパケットを選択する際にローカルな決定を下すかを示しています。

ローカルに格納されているパケットの中で、親スケジュールはまず、インターフェイスに転送する最も適格なパケットを選択します。関連付けられたパケット処理を送信した後、独自のスケジュールエントリに空きスポットが生成されます。そのエントリの子からのパケット処理は存在しません。

子が別のスケジュールである場合、親は *pop*（「最も適格なパケットを選び、そのパケット処理を私に送ってください」と伝えるメッセージ）を送信します。子スケジュールは、各エントリの設定を確認し、次に送信するパケットを決定し、そのパケット処理を親スケジュールに転送します。これで子には、そのスケジュール エントリのフリー スポットが生成されます。子がキューであるため、決定は必要ありません。キューの先頭にあるパケット処理は、（子）スケジュール エントリにロードされます。

図 *25 :* スケジューリングの決定：ルートからリーフへ、ステップ *1* ～ *2*

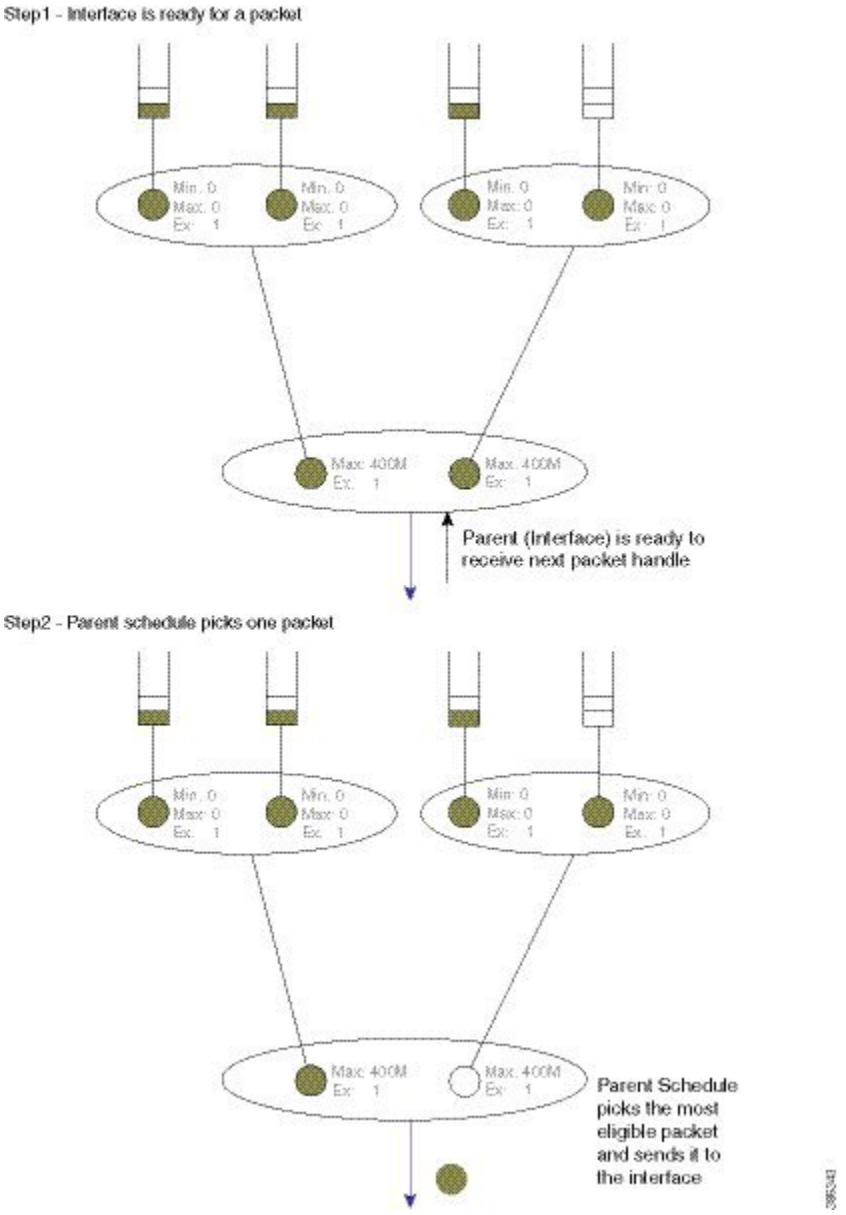


図 *26 :* スケジューリングの決定：ルートからリーフへ、ステップ *3* ～ *4*

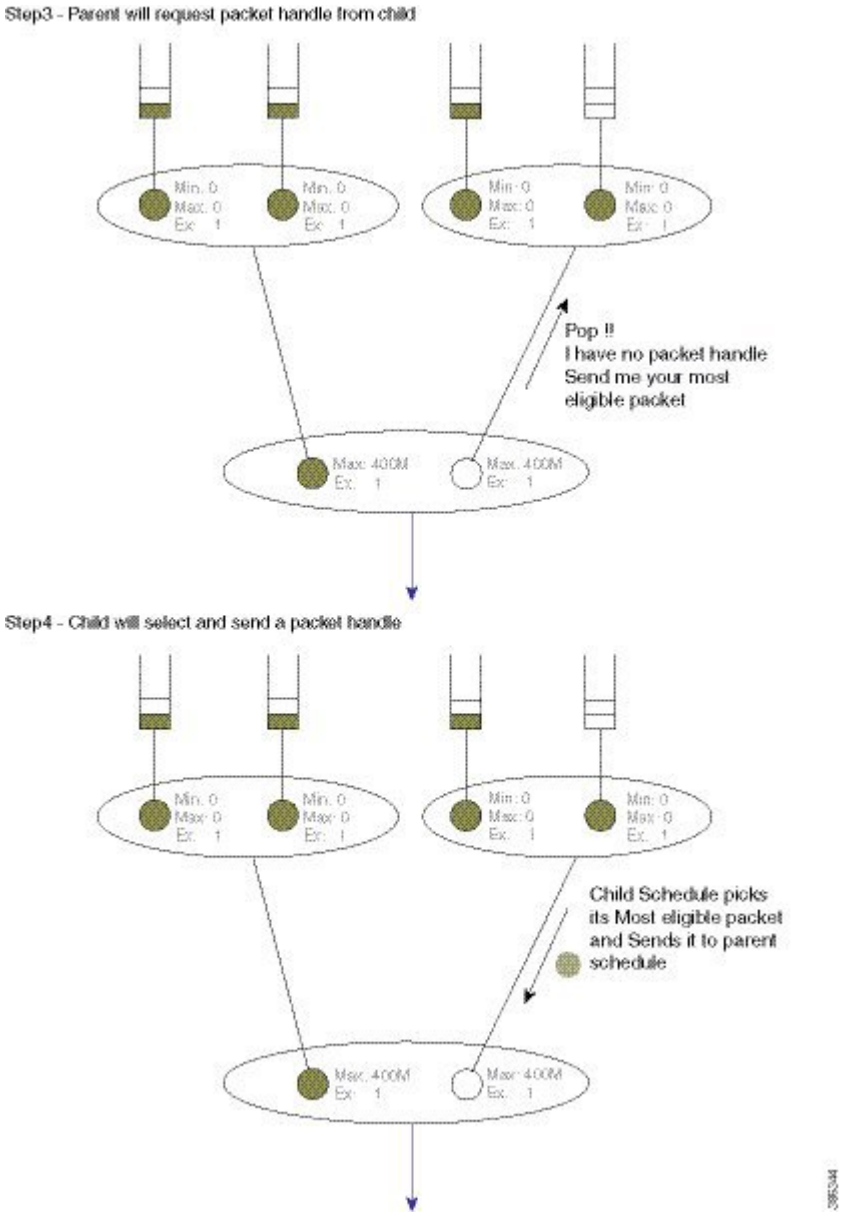
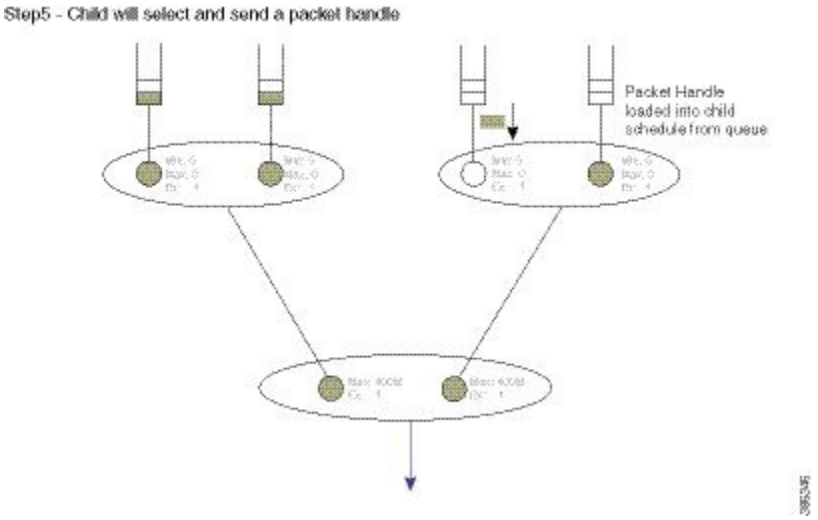


図 *27:* スケジューリングの決定：ルートからリーフへ、ステップ *5*

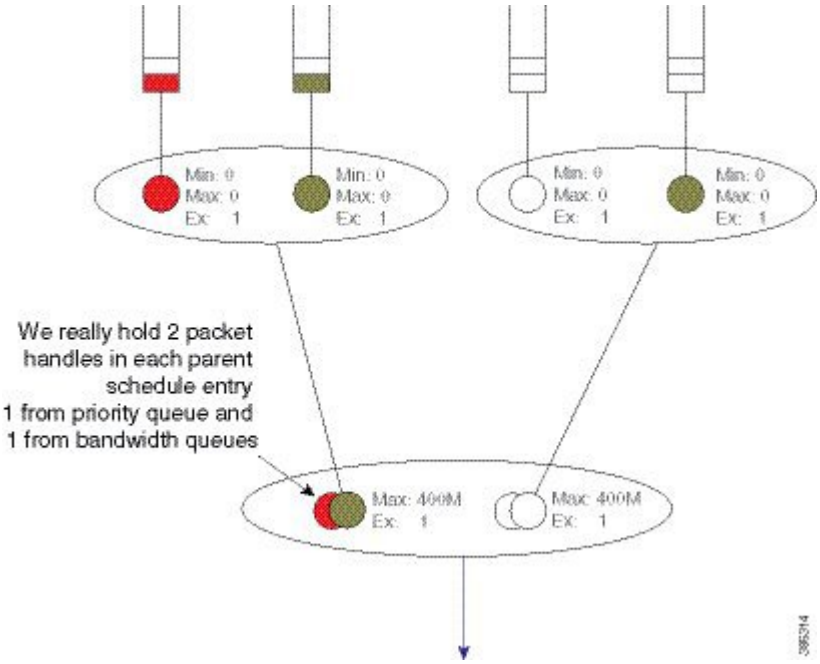


優先度の伝搬の概念

これまでの説明では、帯域幅キューのみを扱っており、やや単純でありました。実際には、各子では、親スケジュールがプライオリティ（キュー）パケットの処理 および帯域幅（キュー）パケットの処理を保持できます（この機能をレーンの通過と呼ぶ）。パケットの処理が子スケジュールから親に送信されると、それがプライオリティクラスまたは帯域幅クラスのどちらから発生したのかを示し、さらに優先度レベルも示します（この動作を優先度の伝播と呼ぶ）。

優先度の伝播については、この章の後半で説明します。ここでは、階層的なスケジューリングのルールを理解する上で概念のみを紹介します。

図 *28 :* 親スケジュールはプライオリティおよび帯域幅の処理（レーンの通過）を保持できます。



この階層では、プライオリティサービスは親スケジュールエントリの設定を必要としません。（親）スケジュールエントリには、最大ウェイトと超過ウェイトの2つのパラメータしかありません。

階層型スケジューリングの操作

スケジューリングの章では、物理インタフェースに適用されたフラットポリシーのスケジューリング決定がどのように行われるかについて説明します。ここでは、リーフスケジュール（子としてキューのみを持つスケジュールに対処するシナリオ）のスケジューリングルールについて説明します。それらの規則はまだ有効です。

親/子のインタラクションのルールを含めるために、ここでその説明をさらに掘り下げていきます。

- プライオリティトラフィックは、親スケジュールで設定されたMax（**shape** コマンド）に対してカウントされます。
- プライオリティ トラフィックは、親で設定されたEx（**bandwidth remaining ratio** コマンド）によって変更されることはありません。
- 親スケジュールのプライオリティパケットは、常に帯域幅パケットの前にスケジュールされます。
- 優先度は、親で設定されたシェープ レートに比例してスケジュールされます。この点はトピックを全体的にカバーするために説明していますが、優先の対象となる負荷がインターフェイスをオーバーサブスクライブできない限り、あまり重要な要素ではありません。

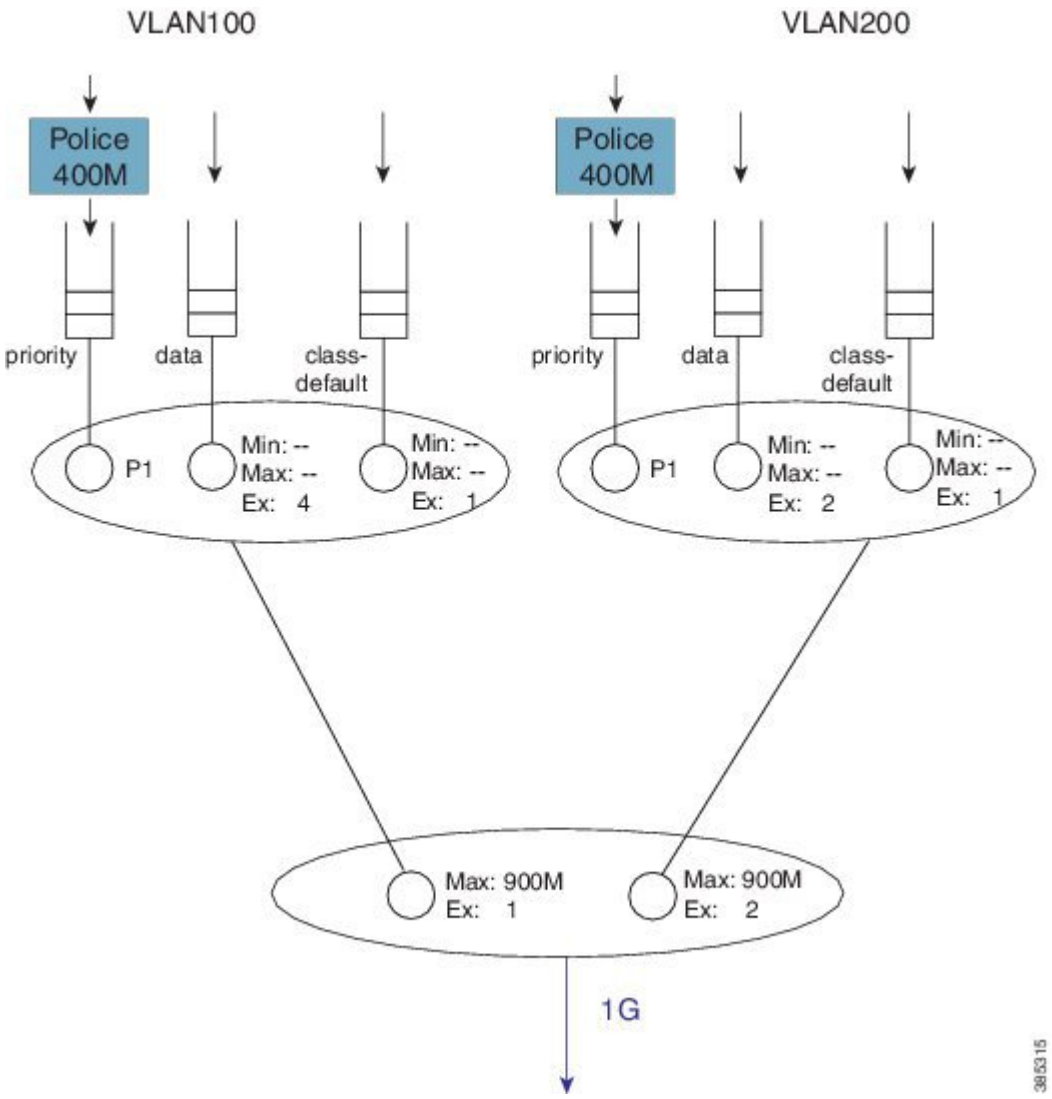
- 優先度の伝播では、親は、パケットがプライオリティ キューから来たことを認識しますが、それが P1（プライオリティ レベル 1）または P2（プライオリティ レベル 2）であるかどうかは認識できません。
- **bandwidth** または **bandwidth remaining** コマンドで設定されたキューからのトラフィックは、親側でも同等に処理されます（最小帯域幅の伝播なし）。これ以降の章では、帯域幅キューからのトラフィックを帯域幅トラフィックと呼びます。.
- 親側での超過ウェイト設定は、プライオリティ トラフィックによって消費されない物理帯域幅をめぐって競合する複数の子からの帯域幅トラフィック間の均等性を制御します。

これらのルールを理解するには、次の設定例を見てみましょう。後で、設定がデータ パス設定にどのようにマッピングされるかを詳しく説明します。現時点では、図と図に示されているスケジュール エントリから動作を充分に理解できます。

```
policy-map child100
  class priority
    priority
    police cir 400m
  class data
    bandwidth remaining ratio 4
!
policy-map parent100
  class class-default
    shape average 900m
    service-policy child100
!
policy-map child200
  class priority
    priority
    police cir 400m
  class data
    bandwidth remaining ratio 2
!
policy-map parent200
  class class-default
    shape average 900m
    bandwidth remaining ratio 2
    service-policy child200
!
int g1/0/4.100
  encaps dot1q 100
  service-policy out parent100
!
int g1/0/4.200
  encaps dot1q 200
  service-policy out parent200
```

次の図は、以前の設定に関連付けられているスケジューリング階層を示しています。前に説明したように、親ポリシーの **shape** コマンドは Max パラメータを設定し、**bandwidth remaining ratio** コマンドはスケジュール エントリに Ex パラメータを設定します（ルール 1 および 2）。明示的に設定されていない場合、後者はデフォルトの 1 になります：

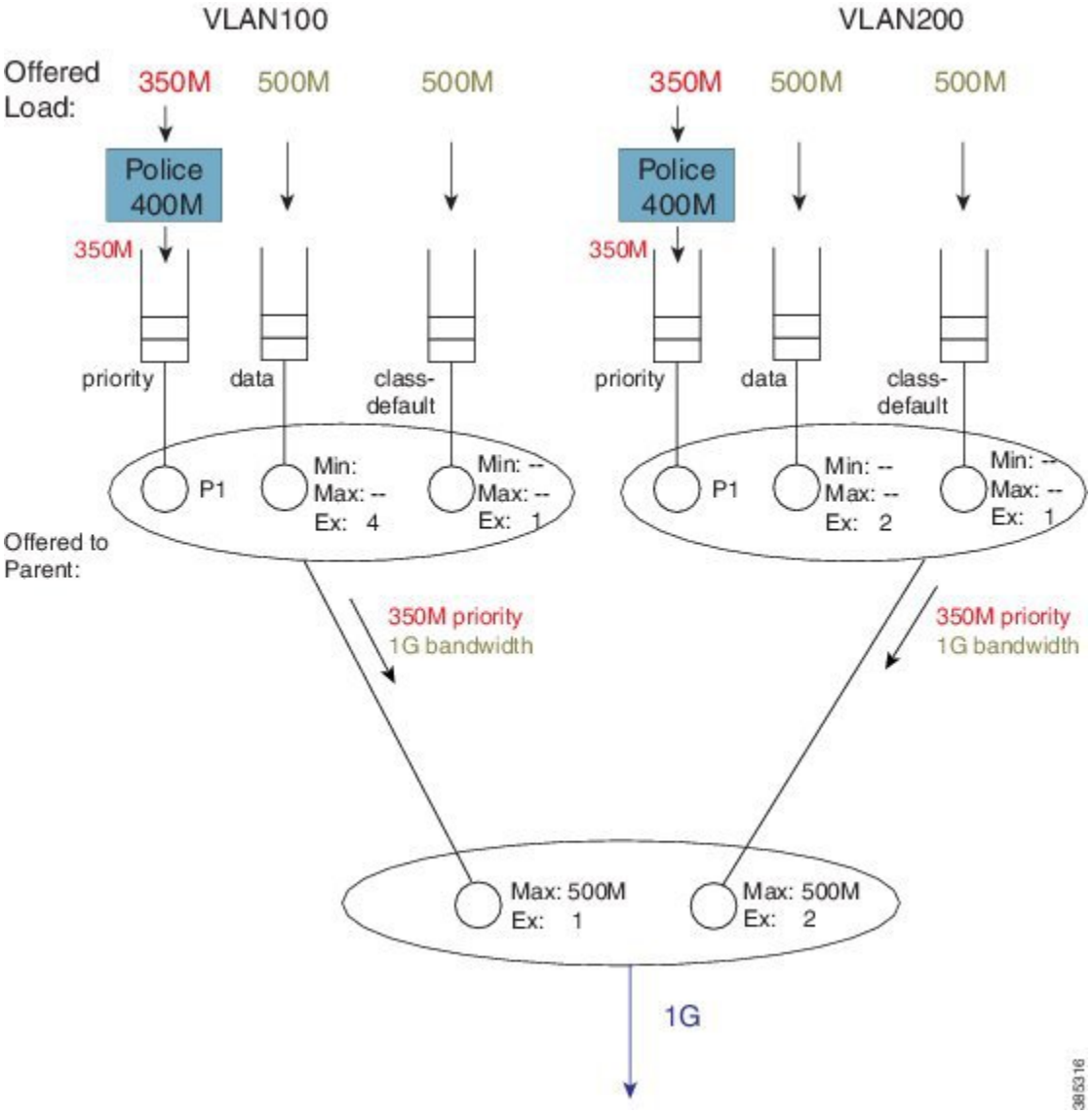
図 *29 :* スケジュール階層の例：プライオリティ負荷全体を転送



それでは、供給負荷に対して予想されるスループットを見てみましょう。

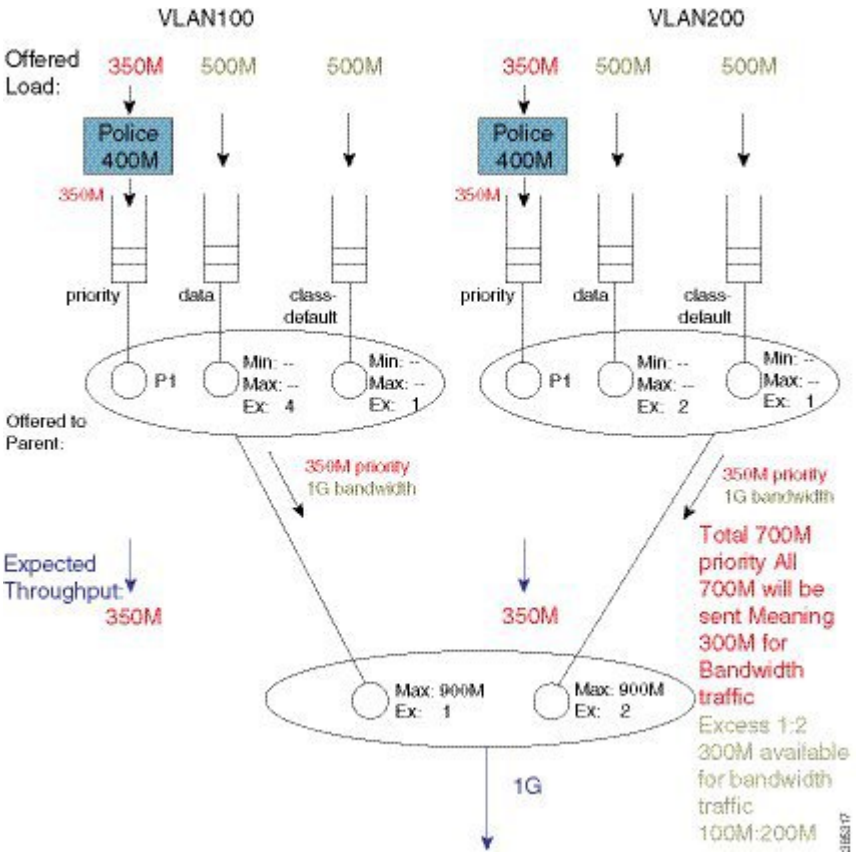
（注） 次の例では、オーバーヘッドアカウンティングを無視しています。これらは、非主要な詳細点とは無関係に、予想されるスループットを計算する方法を説明することのみを目的としています。

図 *30 :* 各子スケジュールから親に提供されるものを計算する



予想されるスループットを計算する際の最初のステップは、クラスごとに供給される負荷を目測することです。次のステップでは、それらを集約して、親に供給されるプライオリティクラスと帯域幅クラスからの合計負荷を観察します。

図 *31 :* 帯域幅キューの残存帯域幅の計算



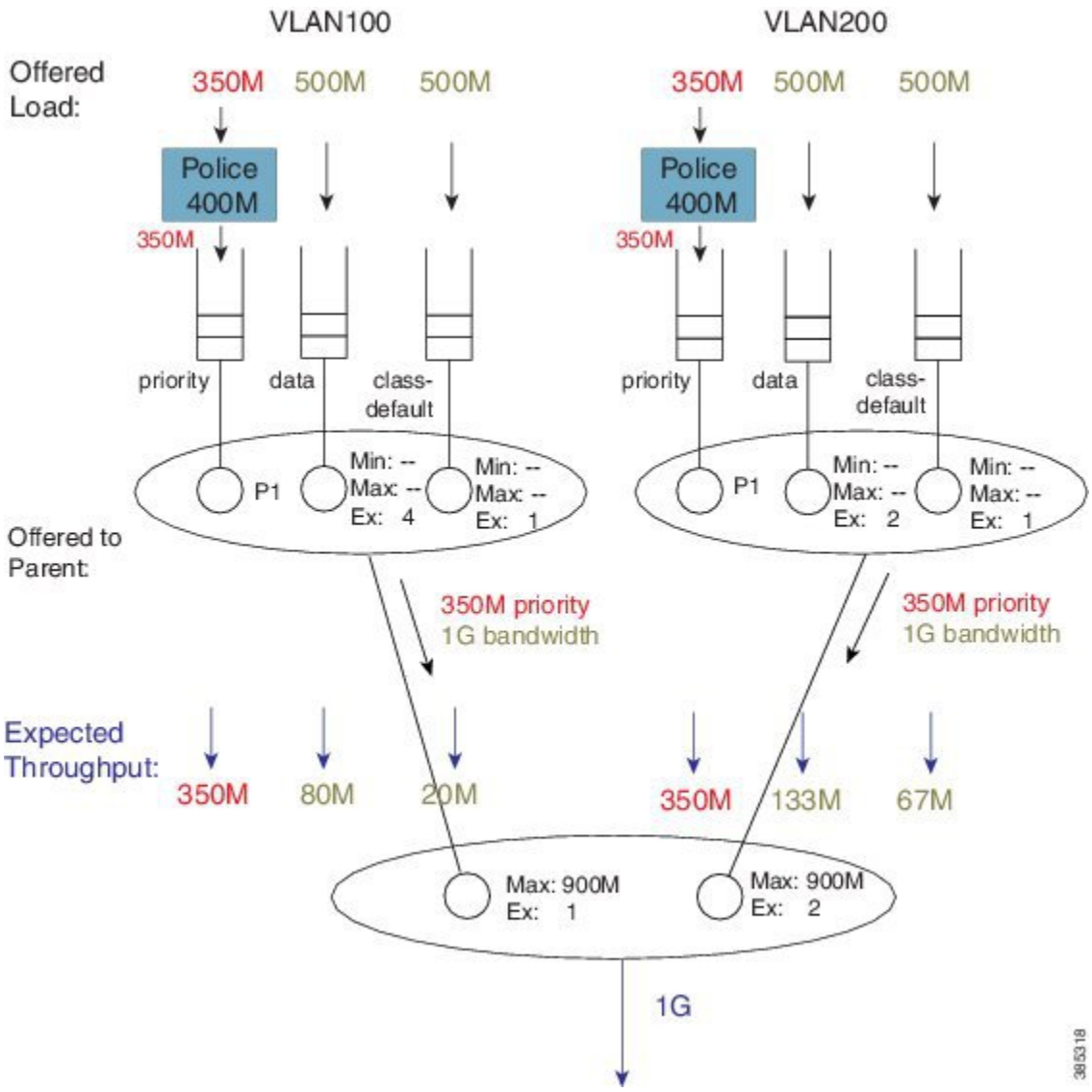
各子スケジュールは、350 Mbps のプライオリティ トラフィックを親に提供しています。インターフェイスには 1 Gbps の利用可能な帯域幅があるため、700 Mbps のプライオリティ負荷全体が転送されます。

ルール 3 に従って、帯域幅トラフィックの前にプライオリティ トラフィックをスケジュールします。各親スケジュール エントリの最大値（レート）が、そのエントリの子スケジュールから供給されるプライオリティ負荷を超えているため、350 Mbps のトラフィック全体が転送されます。

スケジュールされたプライオリティ負荷（350 mbps + 350 mbps のトラフィック）を使用すると、帯域幅（キュー）トラフィックの（残存）帯域幅を計算できるようになります（プライオリティ負荷によって消費された 700 Mbps、総帯域幅 300 Mbps または 1 Gbps）。

親スケジュールは Ex 設定を使用して 300 Mbps の（残存）帯域幅を割り当てます。 VLAN 100 と VLAN 200 の Ex 値が 1 と 2 の場合、それぞれ帯域幅は 1 : 2 で共有されます。VLAN 100 は 100 Mbps を受信し、VLAN 200 は 200 Mbps の帯域幅トラフィック スループットを受信します。

図 *32 :* 子スケジュールの超過ウェイトに基づく帯域幅の共有



この 100 Mbps がどのように割り当てられるかを計算するために、VLAN 100 の帯域幅キューのスケジュール エントリ（スケジュール内）を調べることができるようになりました。

最低帯域保証が設定されていないため（**bandwidth** コマンドは親スケジュールではサポートされていない）、共有はすべて子スケジュールのスケジュール済みEx値に依存します。設定（クラス データに 4、class-default に 1）に基づいて、100 Mbps が 4 : 1 で共有されます（クラス データは 80 Mbps を受信し、class-default は 20 Mbps を受信します）。

VLAN200 で同じアプローチに従うと、利用可能な 200 Mbps は 2 : 1 に分割されます。クラス データは 133 Mbps を受信し、class-default は 67 Mbps を受信します。

おそらくすべてのクラスがオーバーサブスクライブされていることに気づいたでしょう。これは、計算した予想スループットが各クラスの最低保証サービス レートでもあったことを意味します。階層型スケジューリングでは、親スケジュールでの帯域幅共有により、子スケジュール

に送信待ちのパケットがない場合でも帯域幅を浪費しないようにします。フラットポリシーでの帯域幅の共有と同様に、子が使用していない帯域幅は他の子が利用できます。

優先度の伝播

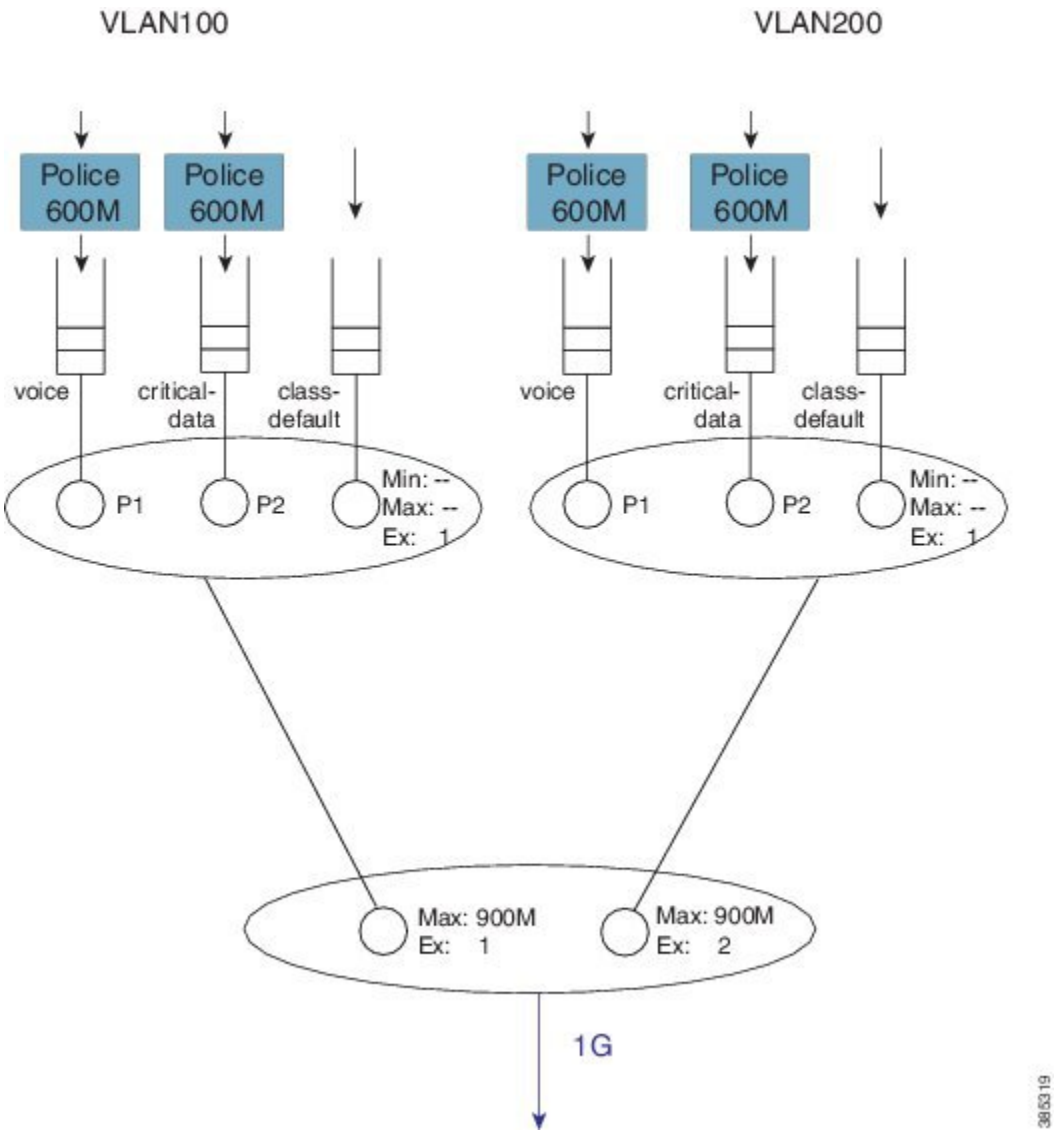
「優先度の伝搬の概念（100ページ）」に関しては、ここでいくつかの点を強調するために次の設定例を使用します。

```
policy-map child
  class voice
    priority
    police cir 600m
  class video
    priority
    police cir 600m
!
policy-map parent100
  class class-default
    shape average 900m
    service-policy child
!
policy-map parent200
  class class-default
    shape average 900m
    bandwidth remaining ratio 2
    service-policy child
!
int g1/0/4.100
  encaps dot1q 100
  service-policy out parent100
!
int g1/0/4.200
  encaps dot1q 200
  service-policy out parent200
```

（注） 両方の親ポリシーマップで同じ子ポリシーを使用しています。一意のポリシーマップは、どのレベルでも不要です。要件が一致する場合は、子ポリシーマップ、さらには親ポリシーマップを共有できます。

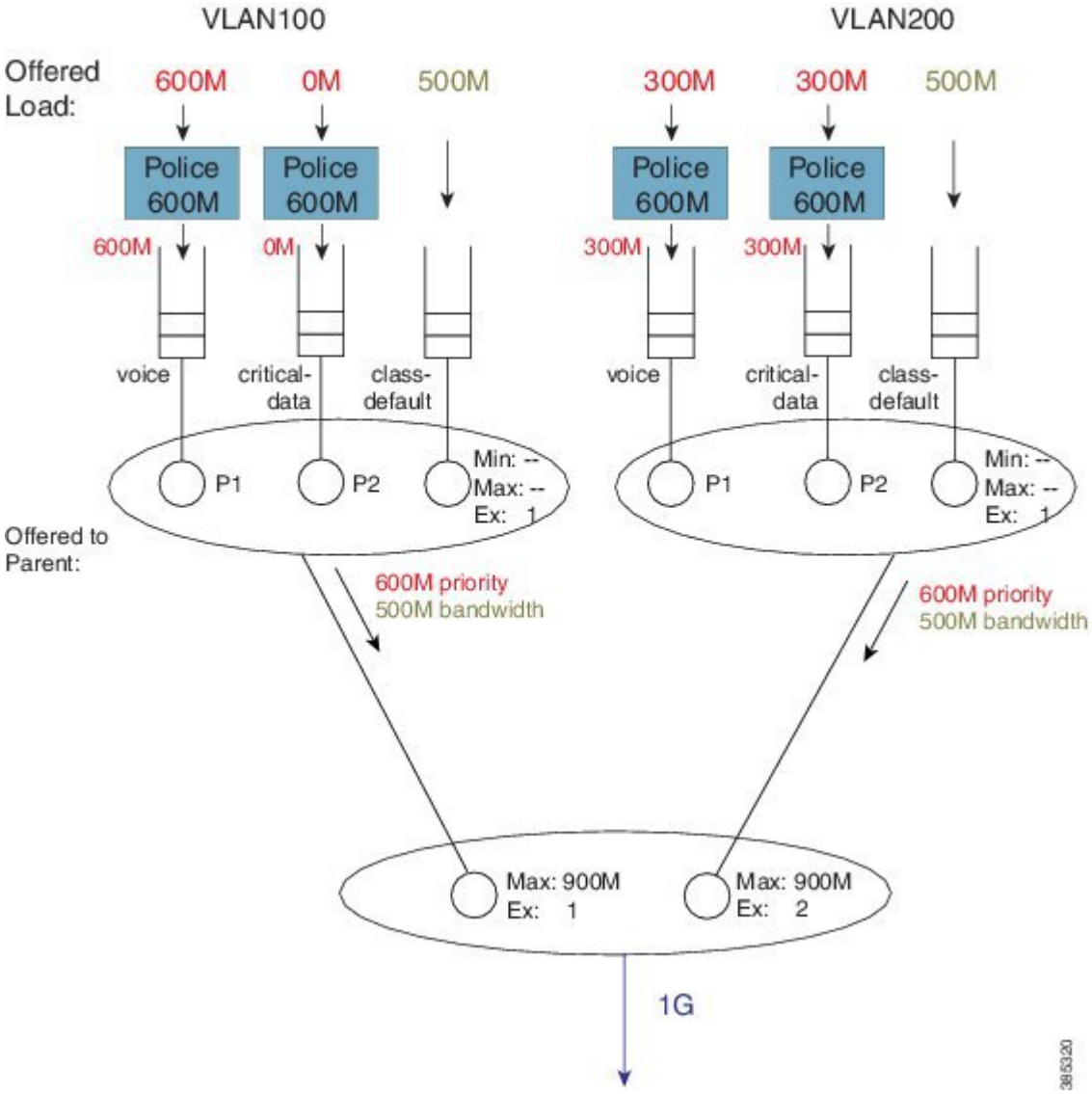
この設定用に作成された階層は次のようになります。

図 *33 :* スケジューリング階層の例：子スケジュールにおけるマルチレベル プライオリティ キューイング



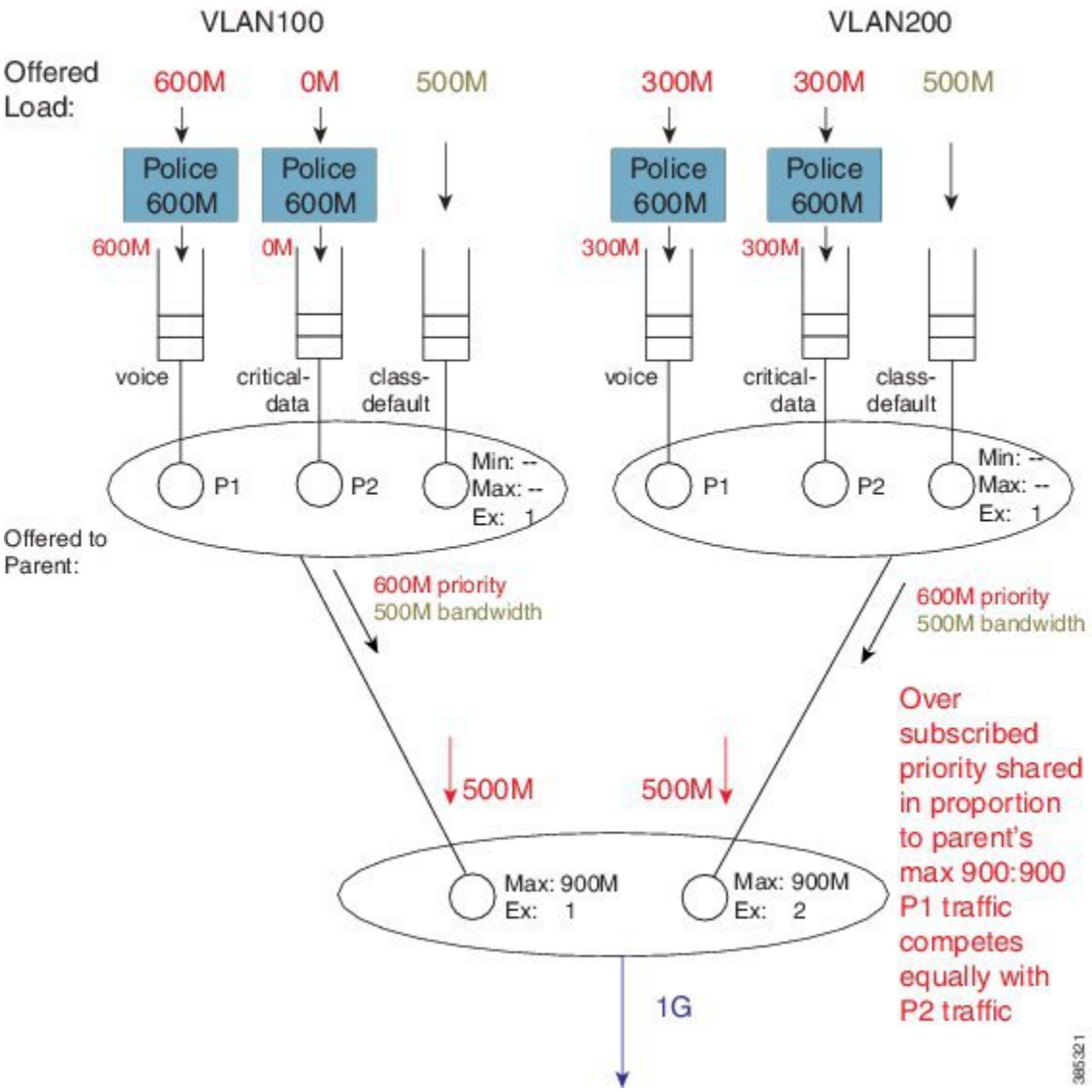
シナリオは、「優先度の伝搬の概念（100 ページ）」によって異なります。子スケジュールでマルチレベルプライオリティキューイングがサポートされるようになりました（例、P1 [プライオリティ レベル 1] および P2 [プライオリティ レベル 2] クラス）。 次の図は、各クラスに供給される負荷を示しています。

図 *34 :* マルチレベル プライオリティ キューイング：各クラスに供給される負荷



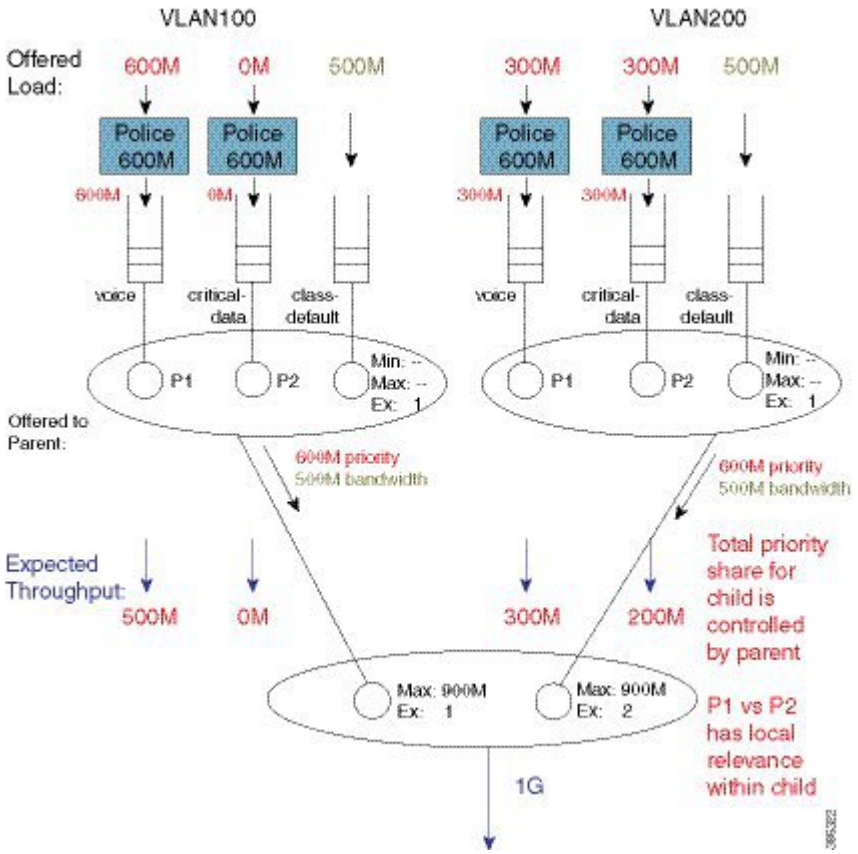
それでは、（各子の）プライオリティ キューと帯域幅キューから親に供給される総負荷を見てみましょう。

図 *35 :* 親の最大比率に対して共有されるオーバーサブスクライブされたプライオリティ キュー



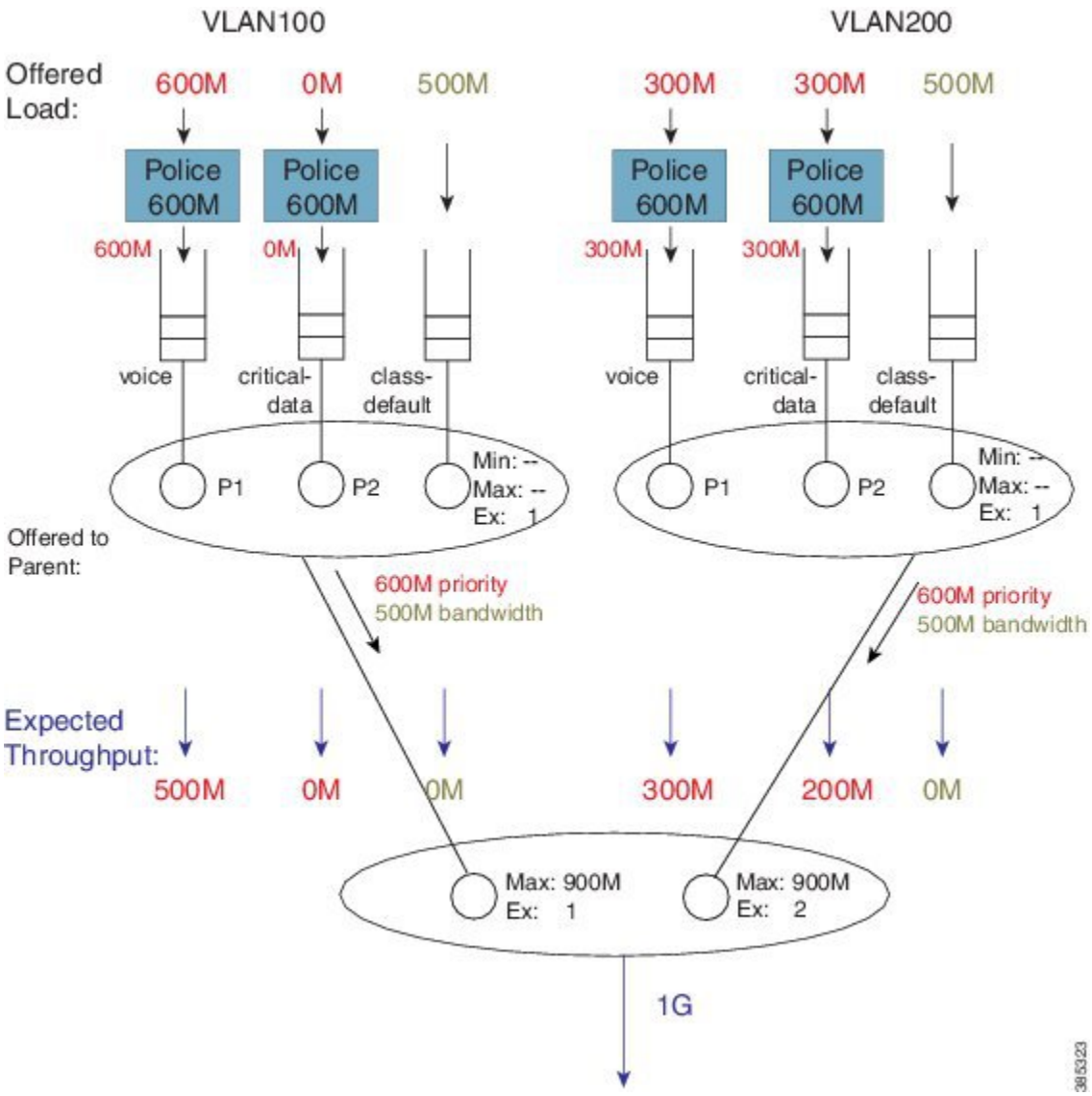
ルール4では、階層型スケジューリングの操作（101ページ）親がスケジュールエントリで設定されたシェープ レートに比例して提供されるプライオリティ負荷をスケジュールすることを規定しています。ここでは、各子は 900 Mbps の最大レート（親ポリシーにおける「シェーピング」）を有し、600M のプライオリティ トラフィック（つまり、1 Gbps しか利用できない場合は（1.2 Gbps [600M + 300M + 300M トラフィック]）を提供します。親スケジュールは各子に 500 Mbps を割り当てます。ここで注意する必要がある重要な点は、VLAN 100 からの P1 が VLAN 200 からの P2 トラフィックと同等に競合することです。（ルール 5 で説明があったように、優先度の伝搬により、パケットがプライオリティ キューから発生したにもかかわらず、優先順位を示していないことが親に警告されます。）

図 *36 :* 親が子の優先度共有の合計を制御する



親スケジュールは、VLAN200 から 500 Mbps のプライオリティ負荷を受け入れます。 子スケジュールは、その 500 Mbps の帯域幅を割り当てます。子ポリシーは、音声クラスで P1 を設定しました。これは、子スケジュールが常にそのキューから常にパケットを最初に選ぶことを意味します（つまり、スケジュールでは、プライオリティ レベルはローカルで有効となります）。VLAN 200 の音声クラスの予想されるスループットは 300 Mbps です。 critical-data クラスは 200 Mbps（この例では 500 Mbps ー 300 Mbps が未使用の割合）を受信します。

図 *37*：親スケジュールから受け取った帯域幅を子スケジュールが割り当てる



帯域幅キューから予想されるスループットはどうですか。供給されたプライオリティ負荷が利用可能な物理帯域幅を超えたため、帯域幅キューには何も残りませんでした。この例は、プライオリティクラスが帯域幅キューを完全に消費する可能性があることを効果的に強調しています。制御パケットがプライオリティキューに入っていないと、ネットワークが不安定になる可能性があります。事実、プライオリティ キューに制御パケットを配置できなかった場合は、設定ミスが考えられます。

（注） 利用可能な物理帯域幅がすべてのプライオリティ クラス ポリサーの合計を超えていないかを確認してください。そうすれば、後者により他のサービスのキューが消費されることはありません。

優先度の伝播の概念が活用できるのは、スケジューリング階層のみではないことに留意してください。パケットをプライオリティ クラスから発生したものとしてマークすると、そのタグは出力インターフェイスに転送されます。出力キャリア カードまたはインターフェイス カードでは、通過するレーンにより、優先パケットができるだけ早くインターフェイスに到達できる場所が複数あります。

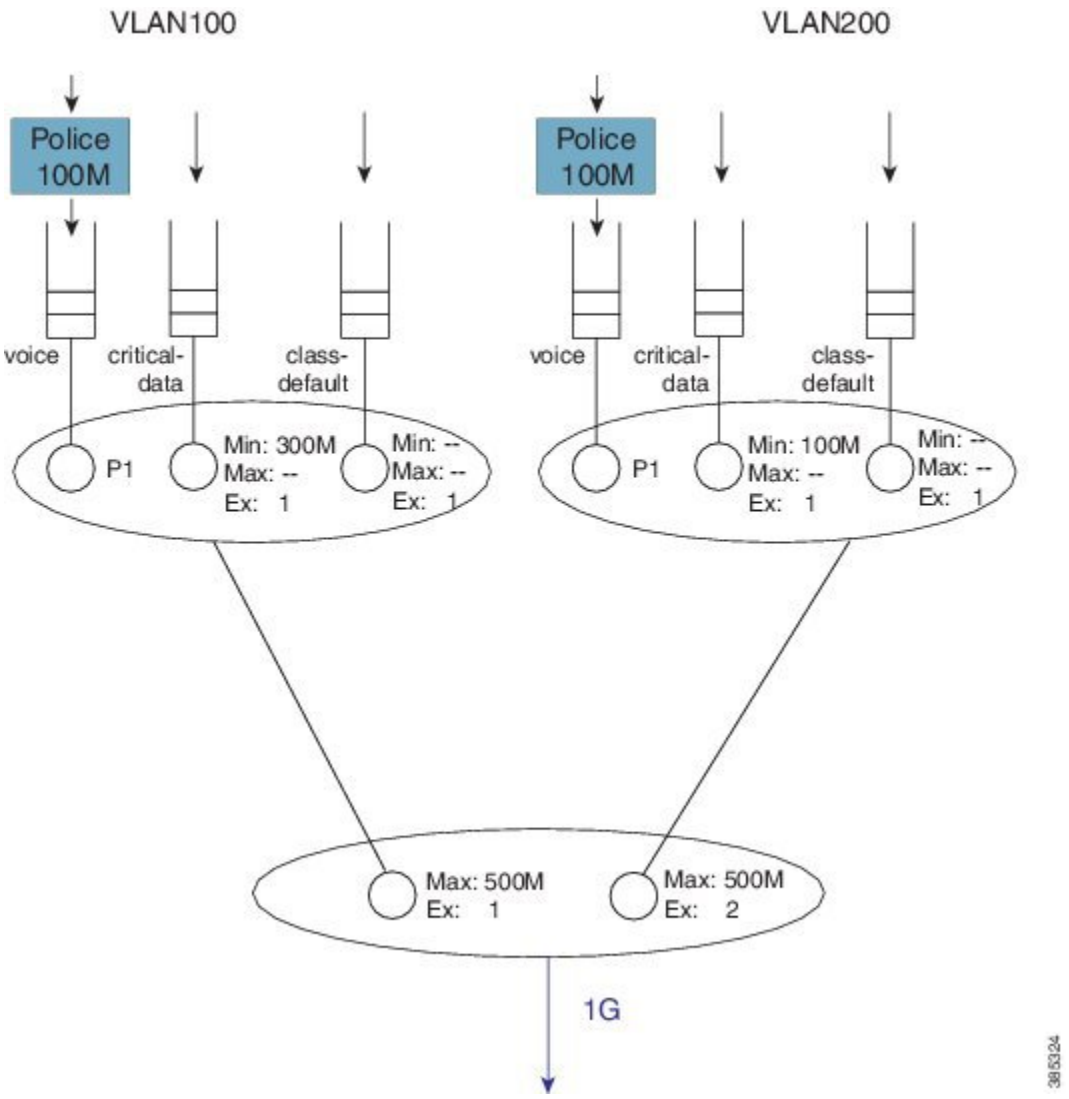
リーフ スケジュールにおける bandwidth コマンド

bandwidth コマンドは親スケジュールではサポートされていませんが（Min 設定はありません）、リーフ スケジュールではサポートされています。次の設定では、子ポリシーマップで **bandwidth** コマンドの動作について説明します。（アスタリスクが付いている行は、この設定が「優先度の伝播（107 ページ）」に示されているものとどのように比較されるかを示しています。）

```
policy-map child100
  class voice
    priority
    police cir 100m
  class critical-data
    bandwidth 300000                  ****
!
policy-map child200
  class voice
    priority
    police cir 100m
  class critical-data
    bandwidth 100000                 ****
!
policy-map parent100
  class class-default
    shape average 500m
    service-policy child100
!
policy-map parent200
  class class-default
    shape average 500m
    bandwidth remaining ratio 2
    service-policy child200
!
int g1/0/4.100
  encaps dot1q 100
  service-policy out parent100
!
int g1/0/4.200
  encaps dot1q 200
  service-policy out parent200
```

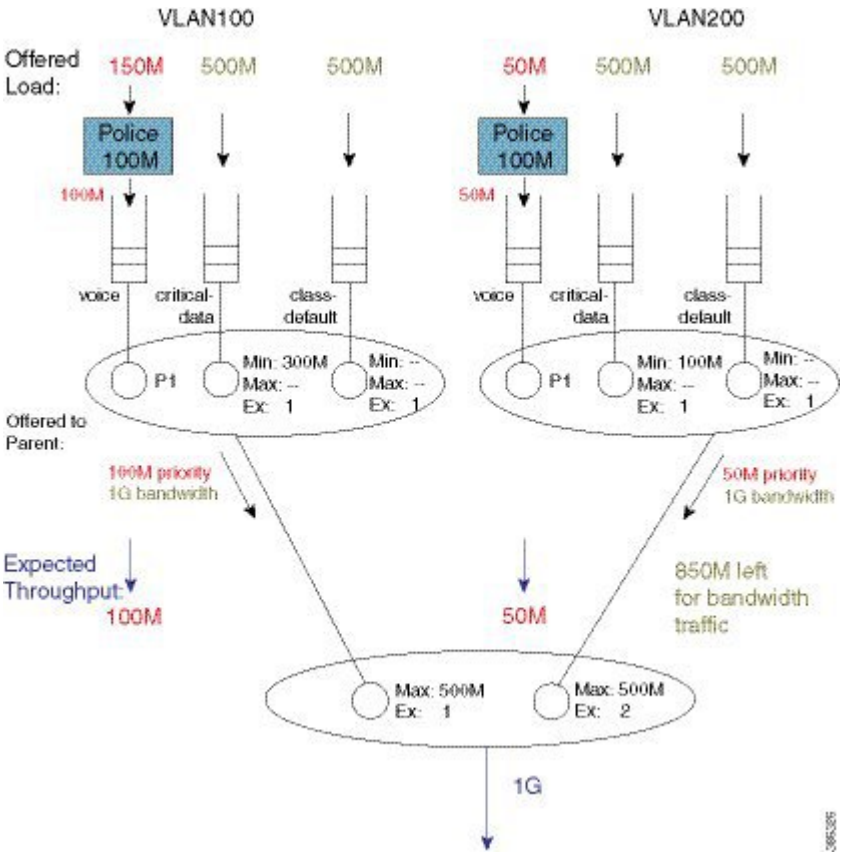
この設定用に作成された階層は次のようになります。

図 *38 :* リーフ スケジュールにおける *bandwidth* コマンドの応用



この階層の動作を説明する上で、（各クラスに）供給される次の負荷を考えてみましょう。

図 *39 :* 各クラスに供給される負荷

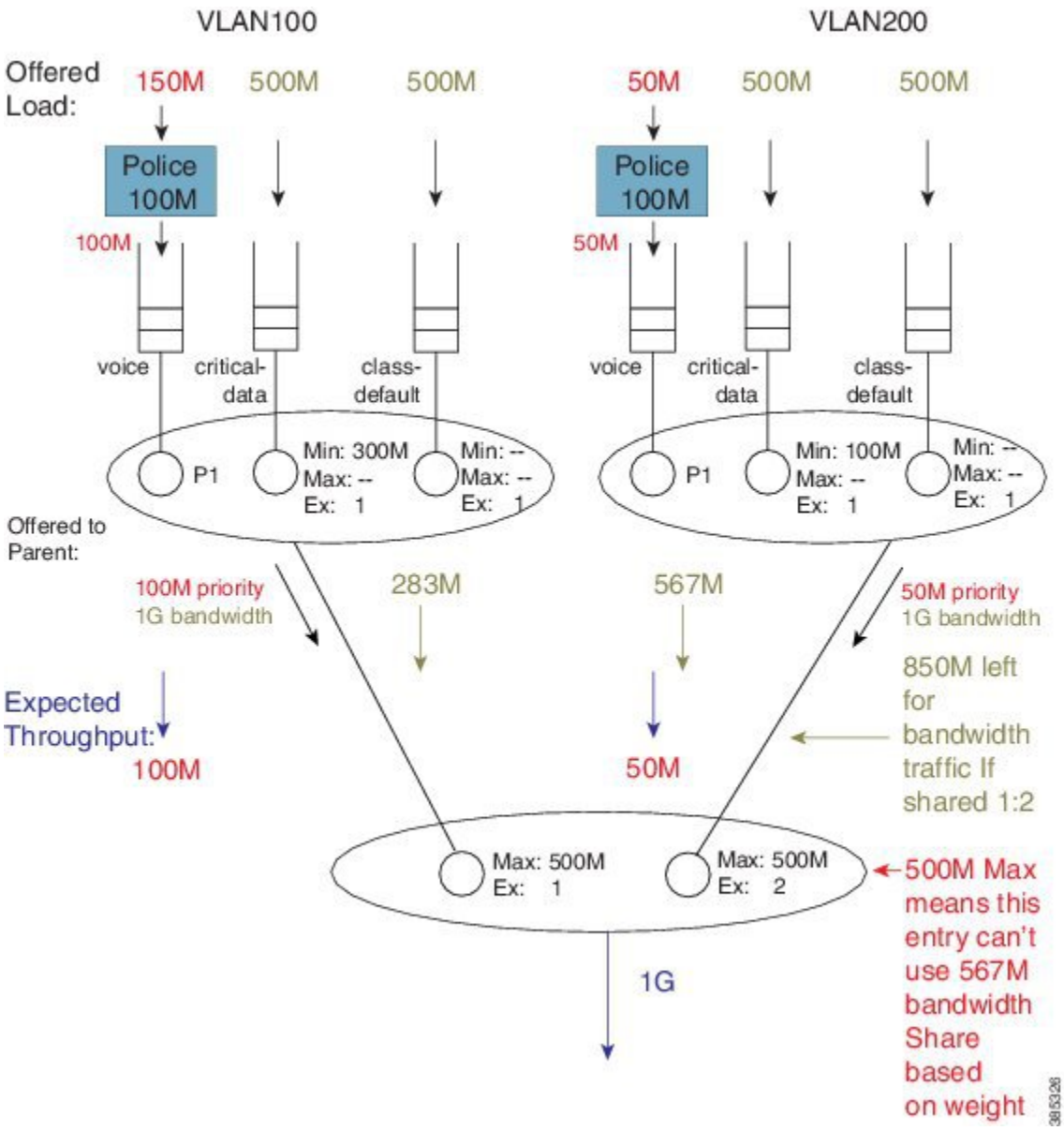


先ほどの例と同様に、（各子の）プライオリティ キューと帯域幅キューが親に供給される総負荷について見てみましょう。

この例での総プライオリティ度負荷は 150M です。各子は、最大レート（親ポリシーのシェーピング）よりも小さい値を供給しており、総プライオリティ負荷は 1 Gbps の合計使用可能帯域幅を下回っています。（提供されたプライオリティ トラフィックの合計が利用可能な合計帯域幅を超えた「スケジュール操作」の例を思い出してください。）これは、各子から供給されるプライオリティ負荷すべてが転送されることを意味します。プライオリティ キューからスケジュールされた 150 Mbps があるため、帯域幅キューでは 850 Mbps が使用可能です。

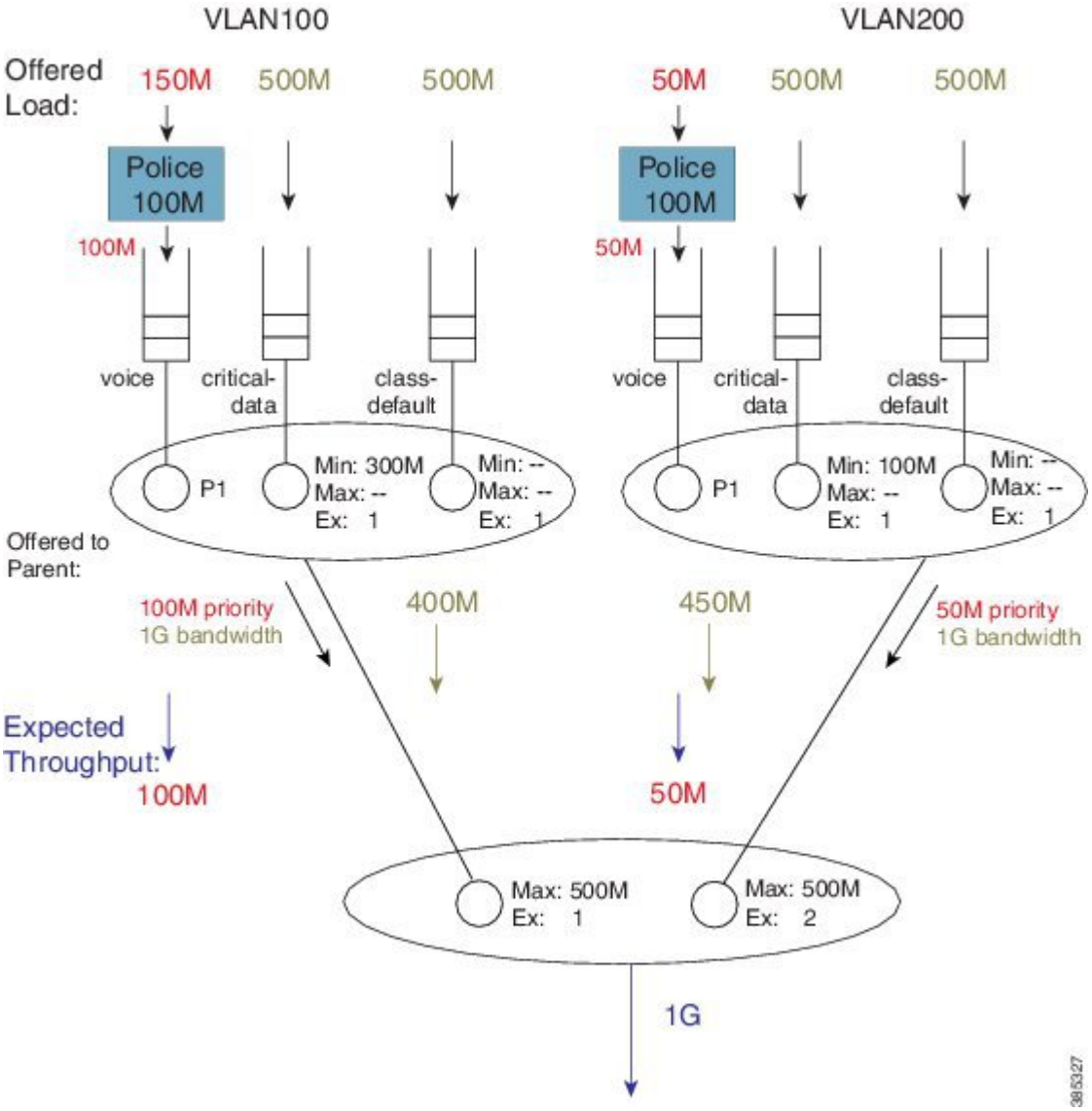
各子間の帯域幅の配分方法を計算するには、まず、親の各スケジュール エントリに設定されている超過ウェイトを見てみましょう。

図 *40 :* 子間での帯域幅の共有



超過ウェイトに焦点を当てる場合、VLAN200 には 567 Mbps のインターフェイス帯域幅（850 Mbps の 2/3）が割り当てられます。 ただし、スケジュール エントリに設定されている最大値（500 Mbps）も考慮する必要があります。これには、その子からの 50 Mbps のプライオリティ トラフィックも含まれます。つまり、VLAN 200 は実際には 450 Mbps の帯域幅トラフィックを転送し、VLAN 100 は 400 Mbps の帯域幅トラフィックを転送します（VLAN 200 では 850 Mbps ー 450 Mbps）。

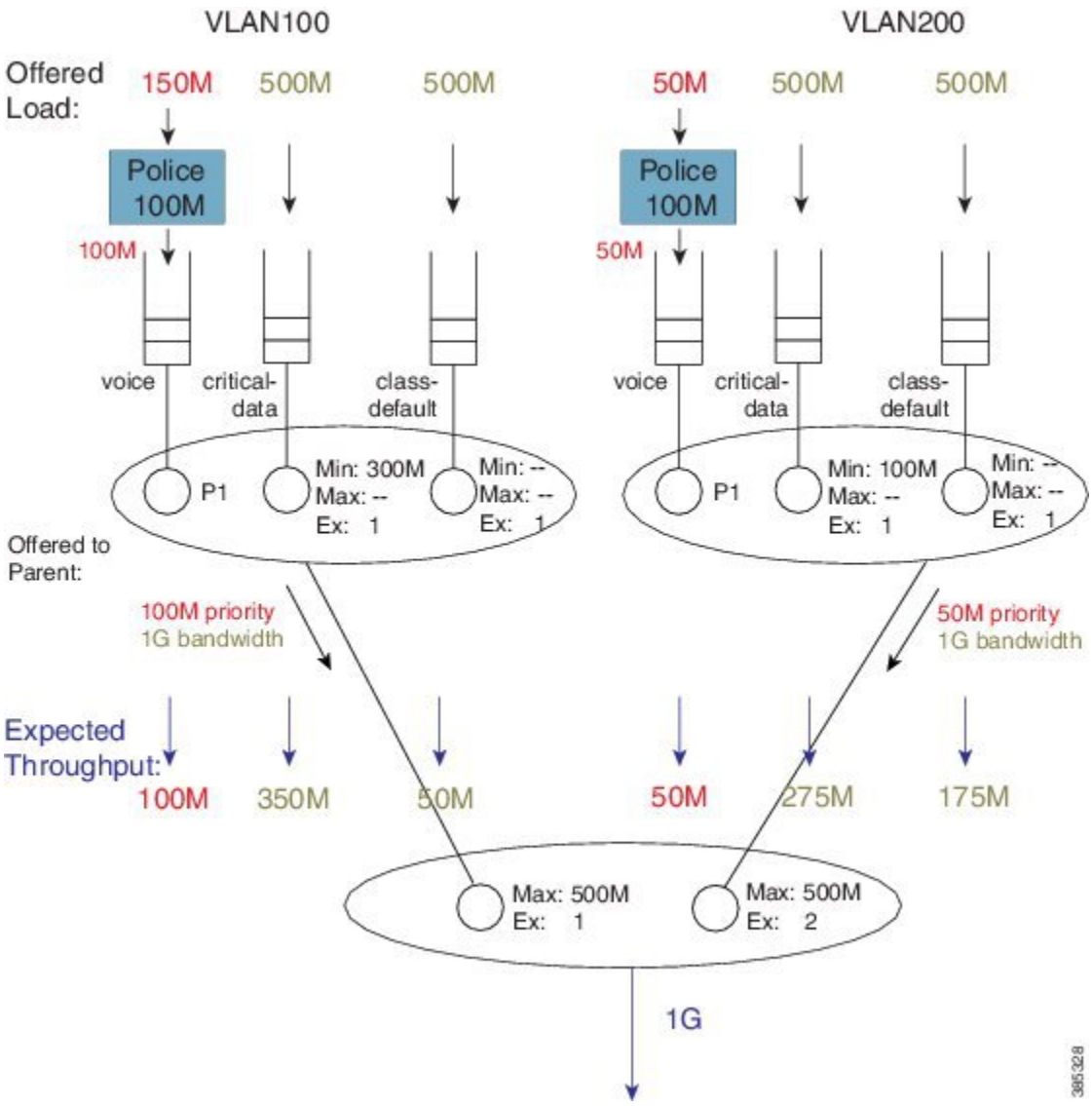
図 *41* : 親のスケジュール エントリの最大値が帯域幅の割り当てに与える影響



親レベルの最大値の合計が利用可能な物理帯域幅以下であるため、親ポリシーの Ex 値により値は追加されません。各子はそのシェープ レートと一致する合計スループットを受け取ります（たとえば、VLAN 100 の場合、100M + 400M = 500M [シェープ レート]）。このような設定では、ある子が使用していない帯域幅は他の子が使用できないことに注意してください。すべての子は常に設定された最大値に制限されます。

各子の帯域幅クラスの合計スループットを使用して、その子の各クラスのスループットを計算できます。「スケジュール操作」で説明したように、最低帯域幅保証が常に最初に処理され、超過帯域幅は Ex 値に基づいて共有され、常に 1 がデフォルトになります。

図 *42 :* 各子スケジュール内で帯域幅を割り当てる上での合計スループットの因数分解



たとえば、VLAN200 の critical-data クラスに割り当てられた帯域幅は、275M（Ex の比率 1：1 から「1/2」が算出され、100M（最低帯域保証）+ 1/2（450M-100M））になります。

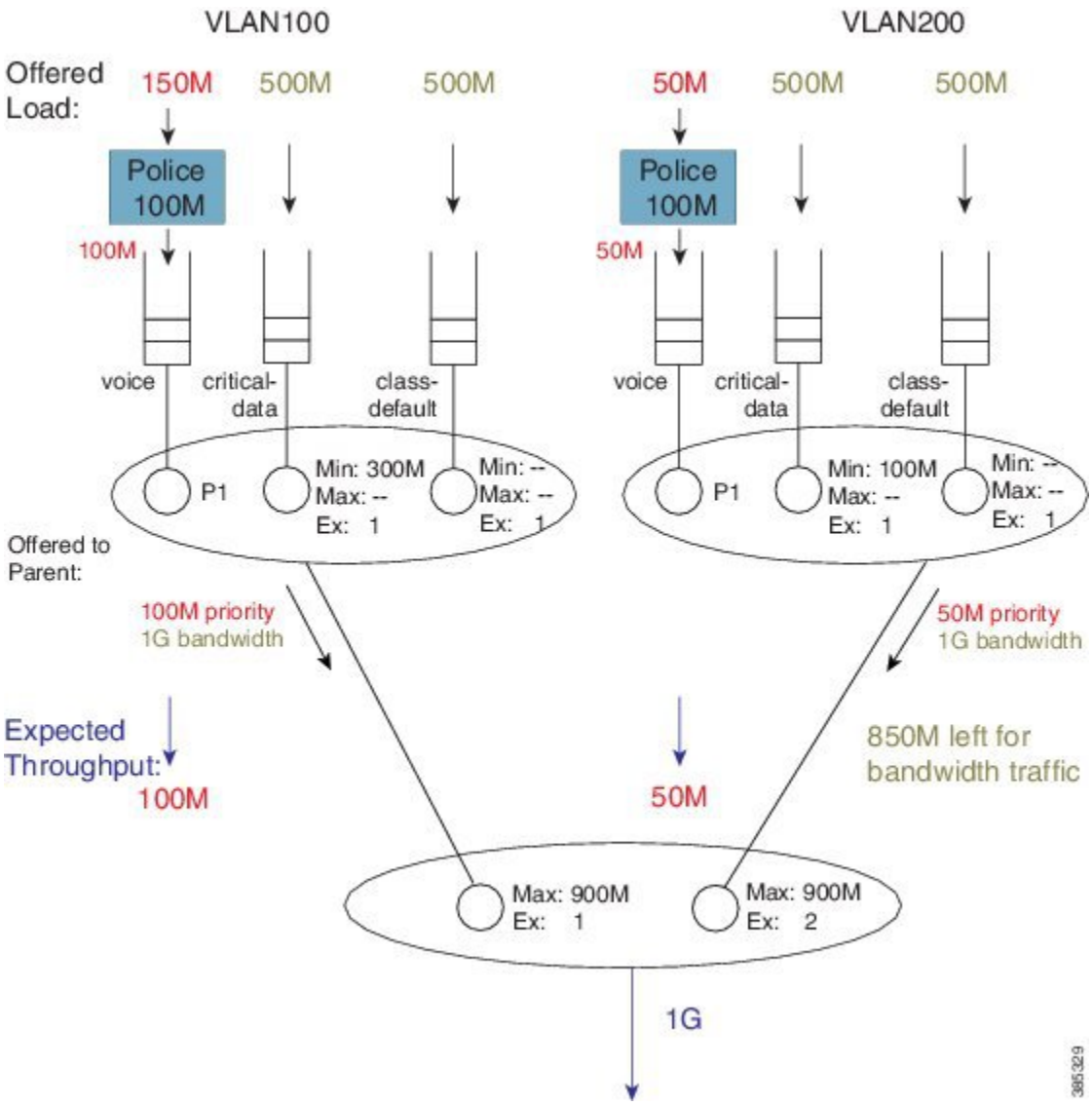
Bandwidth コマンドはローカルでのみ有効

階層型ポリシーで**bandwidth** コマンドを使用するリスクを説明する上で、親シェーパーを増やし、これらが制約要因ではなくなるように、前の設定例を変更します。修正された設定では、親シェーパーの合計数が、利用可能な物理帯域幅をオーバーサブスクライブします。（アスタリスクが付いているコマンドは、この設定が「リーフ スケジュールにおける bandwidth コマンド （113 ページ）」で説明された設定とどのように違っているかを示しています。）

```
policy-map child100
  class voice
    priority
    police cir 100m
  class critical-data
    bandwidth 300000
!
policy-map child200
  class voice
    priority
    police cir 100m
  class critical-data
    bandwidth 100000
!
policy-map parent100
  class class-default
    shape average 900m                          ****
    service-policy child100
!
policy-map parent200
  class class-default
    shape average 900m                          ****
    bandwidth remaining ratio 2
    service-policy child200
!
int g1/0/4.100
  encaps dot1q 100
  service-policy out parent100
!
int g1/0/4.200
  encaps dot1q 200
  service-policy out parent200
```

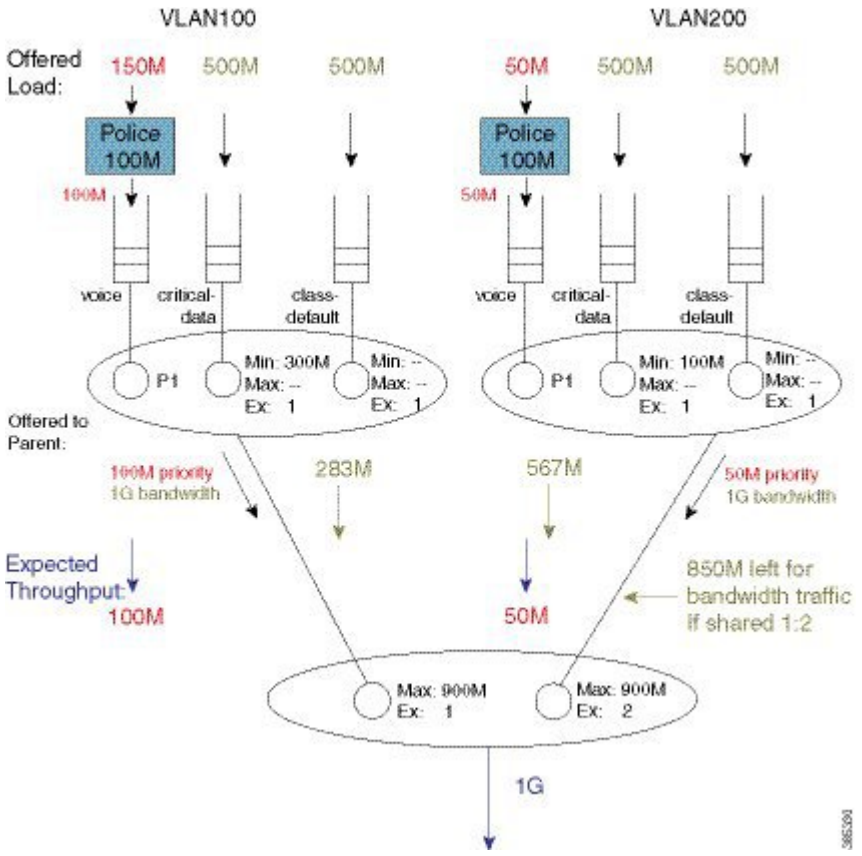
リーフ スケジュールにおける bandwidth コマンド （113 ページ）供給された負荷プロファイルを適用した場合、階層と負荷プロファイルは次のようになります。

図 *43 :* スケジューリング階層の例：親シェーパーは制約を受けなくなった



前の例と同様に、850 Mbps が帯域幅キューに使用できます（残存）。（各子に割り当てられたプライオリティ負荷と帯域幅トラフィックの合計割合を調べると、各子スケジュールの最大値を超えていないことがわかります。）親のスケジュールで設定されている超過ウェイトに基づいて、各子が親スケジュールから受け取る帯域幅の割合を計算します。283 Mbps と 567 Mbps になります。

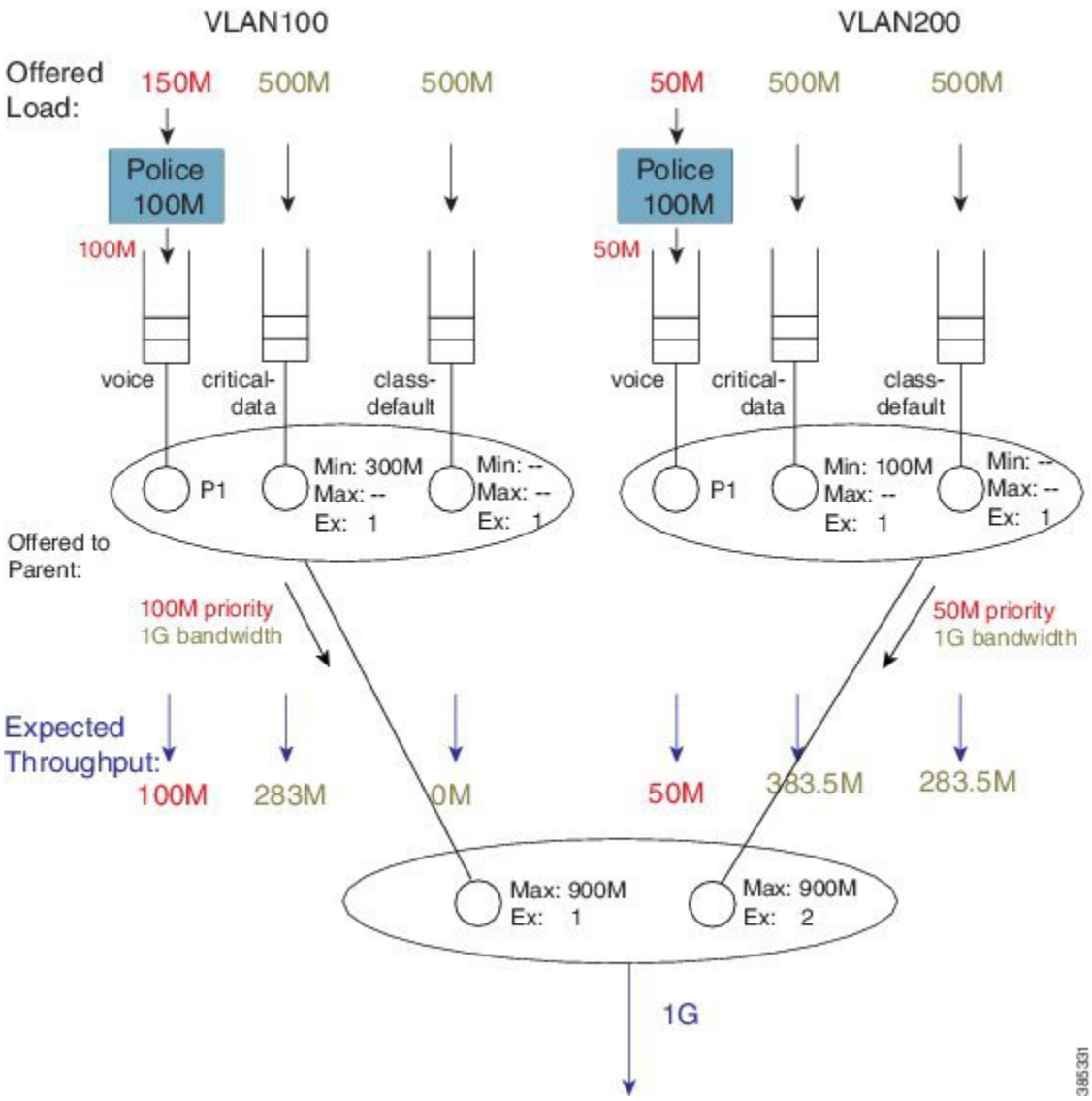
図 *44 :* 親スケジュールで設定された超過ウェイトに基づいて、各子が受け取る帯域幅分を計算する



（注） 前の例とは対照的に、シェープ値はもはや制約されていないため、各子の合計スループットはシェープ_レートとは一致しません。

各子スケジュールのエントリを調べて、各クラスに帯域幅がどのように割り当てられるかを確認しましょう。

図 *45 :* 子エントリにどのように帯域を割り当てるかを指定する方法



VLAN100 の子スケジュールを確認すると、critical-data クラスのスケジュール エントリに 300 Mbps の最小値が設定されていることがわかります。 このスケジュールに割り当てられた 283 Mbps の帯域幅は、この保証を満たすには不十分です。

ここで重要なポイントは、最低帯域幅保証はローカルにのみ関連しているということです。最小帯域幅の伝播はされません。子スケジュールからのトラフィックは、別のスケジュールからの超過トラフィックと同等に競合します。

また、スケジューリング階層で Min を使用すると、他のサービス クラス（この例では VLAN 100 の class-default）のキューが消費される可能性があります。これを回避するには、子ポリシーの **bandwidth remaining** コマンドのみを使用します。

Tip

階層ポリシーで親シェーパーをオーバーサブスクライブし、一部のサービス クラスが消費されないようにするには、プライオリティ キューのポリサーの合計が利用可能な帯域幅を超えないようにしてください。さらに、**bandwidth** コマンドの **bandwidth remaining** 使用を検討してください。

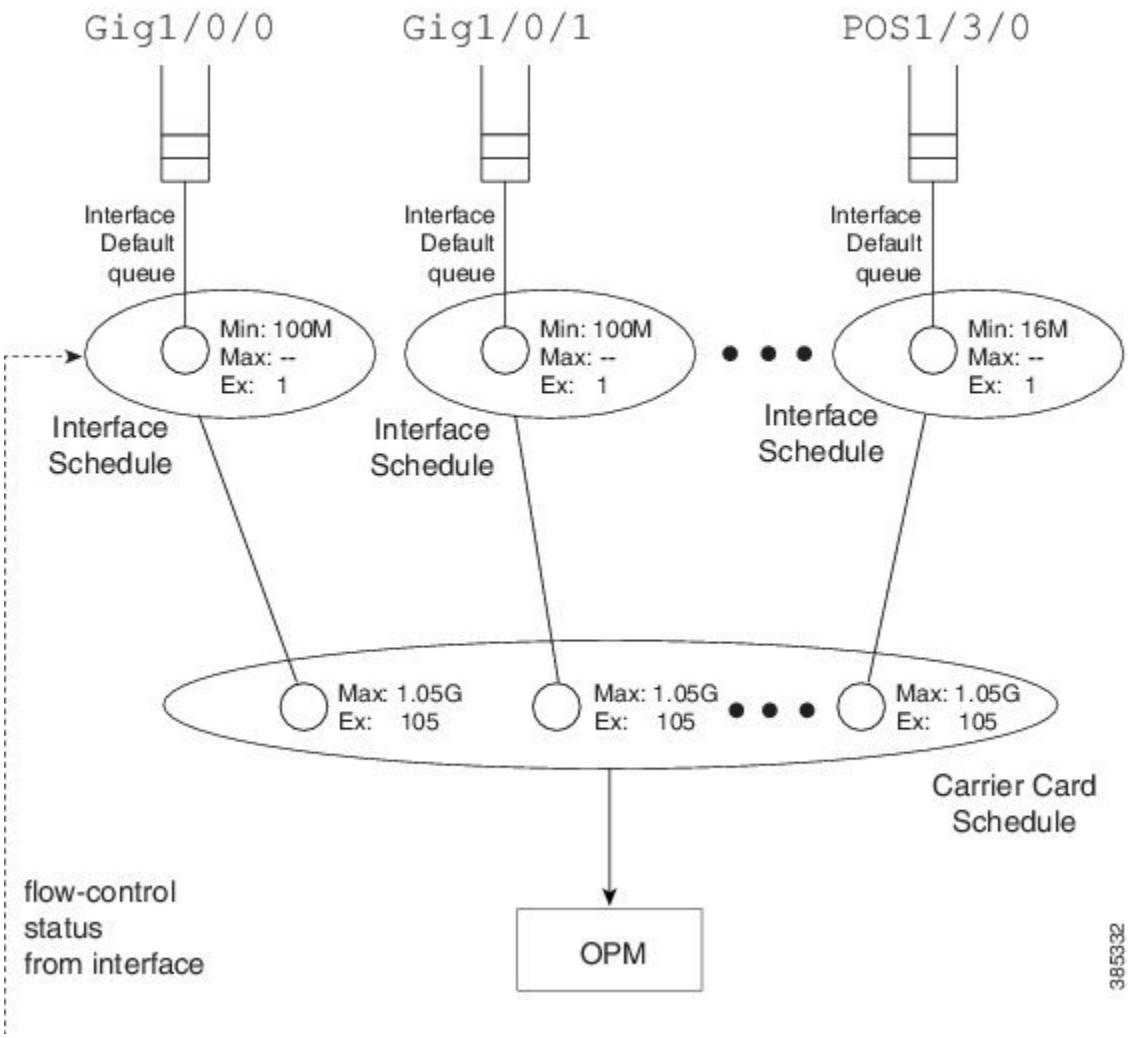
論理インターフェイスに適用されたポリシーマップ

この章の前半では、スケジューリング階層を作成するための 2 つの主な方法、論理インターフェイスに適当された QoS ポリシーと階層型ポリシーマップについて説明しました。 前の例では、論理インターフェイスに適用されるポリシーについて概説しました。 このシナリオをさらに詳しく見ていきましょう。

インターフェイス スケジューリング

論理インターフェイス上のポリシーによって階層がどのように変更されるかを考える前に、QoS ポリシーが適用される前に存在するインターフェイス スケジュールと階層を注意深く調べる必要があります。

図 *46 : QoS* ポリシーの適用前のインターフェイス スケジュールと階層



OPM（出力パケット モジュール）は、スケジューリング階層のルートにあります。パケット処理を受信すると、実際のパケットをメモリから取得し、それを物理インターフェイスにプッシュします。

（意思決定の観点から）OPM 層の直下に、キャリア カードのスケジュールが表示されます。. モジュラー プラットフォームでは、スロットごとにそのようなスケジュールが 1 つありますが、固定システムではシステム全体に対して 1 つあります。

ここで、モジュラ型シャーシについて考えてみてください。スロットが 1 つ搭載され、バックプレーンを介して 10 Gbps のリンクを持つ SIP10 を ESP（組み込み式サービス プロセッサ、転送プロセッサとも呼ばれる）に収容しています。. SIP10 には 4 つの SPA（共有ポート アダプタ）が搭載可能で、 それぞれ最大で 10 Gbps 容量のインターフェイスを搭載可能です。SIP 内で SPA を組み合わせ、バックプレーンの容量を超えると、そのリンクが輻輳ポイントになる可能性があります。万が一これが発生した場合、キャリア カードのスケジュールにより、インタフェース間の均等性が保証されます。各インターフェイスの超過ウェイトはインターフェイス速度に比例します。

プラットフォーム内のトラフィックを調整するために、 各インターフェイスのキャリア カードスケジュールの最大値において、インターフェイスの帯域幅をわずかに超えるように設定します。物理的なインターフェイスに向けて十分なトラフィックを送信する必要があるため、そのインターフェイスを消費させないようにします。さらに、出力バッファがいっぱいになったことがインターフェイスに表示されるたびに送信を中止する必要があります。これは、インターフェイスがダウンストリームデバイスから一時停止フレームを受信するときに発生する可能性があり、シリアルインターフェイスはビットまたはバイトスタッフィングなどによりデータを拡張します。

ここが重要な点で、物理的なインターフェイスが常にネットワークに送信できるデータを持てるように、そのインターフェイスに向けてトラフィックをプッシュし、十分なデータがバッファリングされていることがインターフェイスに表示されるたびに一時的に送信を一時停止する必要があります。

インターフェイスは、フロー制御メッセージによるトラフィックの送信を停止するように指示します。. 設計上、スケジュール（スケジュール エントリではない）がこのメッセージに応答します。送信を停止します。このため、ボックス内のすべての物理インターフェイスに対して常にインターフェイスのスケジュールを設定する必要があります。インターフェイスデフォルト キュー（QoS がないときに使用されるキュー）は、このインターフェイスのスケジュールの子です。

各インターフェイスは、プライオリティ トラフィックと帯域幅トラフィック用に個別のバッファとキューを維持しながら、異なる優先度の高いおよび低いフロー制御メッセージを（インターフェイス スケジュールに）送信できます。

帯域幅トラフィック バッファがいっぱいになっているというメッセージをスケジュールが受信すると、そのようなトラフィックは一時停止されますが、優先度の高いトラフィックは転送され続けます。

プライオリティ バッファがいっぱいになっているというメッセージを受信した場合、輻輳が解消されるまでパケットの送信を一時停止します。

この方式は優先度の伝播の概念を物理インタフェースに活用し（これは、パケット処理がプライオリティ クラスまたは帯域幅クラスのどちらに派生するのかを意味します）、遅延の影響を受けやすいトラフィックのジッタを業界最先端レベルまで最小に抑えます。

親ポリシーでのシェーピング/子ポリシー上のキューイング

次に、論理インターフェイスに適用されている可能性のある通常のポリシーを見てみましょう（コンストラクトは「親ポリシーでのシェーピングー/子ポリシーでのキューイング」を参照）：

```
policy-map child100
  class voice
    priority
    police cir 100m
  class critical-data
    bandwidth 300000
!
policy-map parent100
  class class-default
    shape average 900m
    service-policy child100
```

```
!
int g1/0/0.100
  encaps dot1q 100
  service-policy out parent100
```

このコンストラクトでは、親ポリシーでシェーパーを設定する必要があります（シェーパーの平均 900M）。このコンストラクトの本来の目的は、各論理インターフェイスに帯域幅を割り当てることでした。シェーピング（Max）レートを、その論理インターフェイスが所有する帯域幅であるとみなし、子ポリシーがその所有共有内の帯域幅を分配できるようにします。

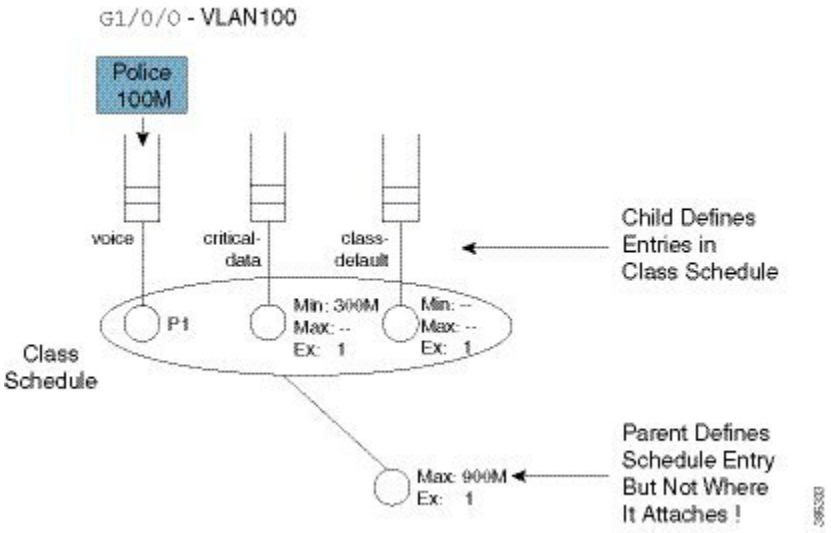
このコンストラクトの便利な応用方法の1つは、リモートサイトのトラフィックを調整することです。たとえば、所属企業のハブに GigabitEthernet リンクがあり、T1 接続でリモート ブランチにトラフィックを送信しているとします。リモートブランチが受信できるレートでトラフィックを送信する必要があります。そのブランチにサービスを提供するプロバイダのデバイス内のパケットがドロップする可能性を回避するには、親シェーパーを T1 レートで設定し、パケットをハブでキューイングします。これにより、そのブランチのリンクが輻輳ポイントであった場合に最初に転送されるものが制御されます。

顧客は、論理インタフェース（個々のサブスクライバーまたはリモートサイトのいずれかを表す）上でシェーパをオーバープロビジョニングするようリクエストしています。すべての論理インタフェースが常にアクティブであるとは限らないということが前提です。個々のサブスクライバのスループットを制限する必要があるため、個々の論理インターフェイスに割り当てられた帯域が消費されていない場合は、その帯域幅を浪費しないようにする必要があります。

それでは、オーバーサブスクライブはどうでしょうか。オーバーサブスクライブする場合、輻輳時に均等性を実現するために、超過ウェイト値を使用して親の帯域幅余剰比率を設定する必要があります。さらに、輻輳時に個々の論理インターフェイスがどのサービスを受けるかに注意してください。

それでは設定の話に戻ります。階層は次のようになります。

図 *47*:「親ポリシーでのシェーピング/子ポリシーでのキューイング」コンストラクト

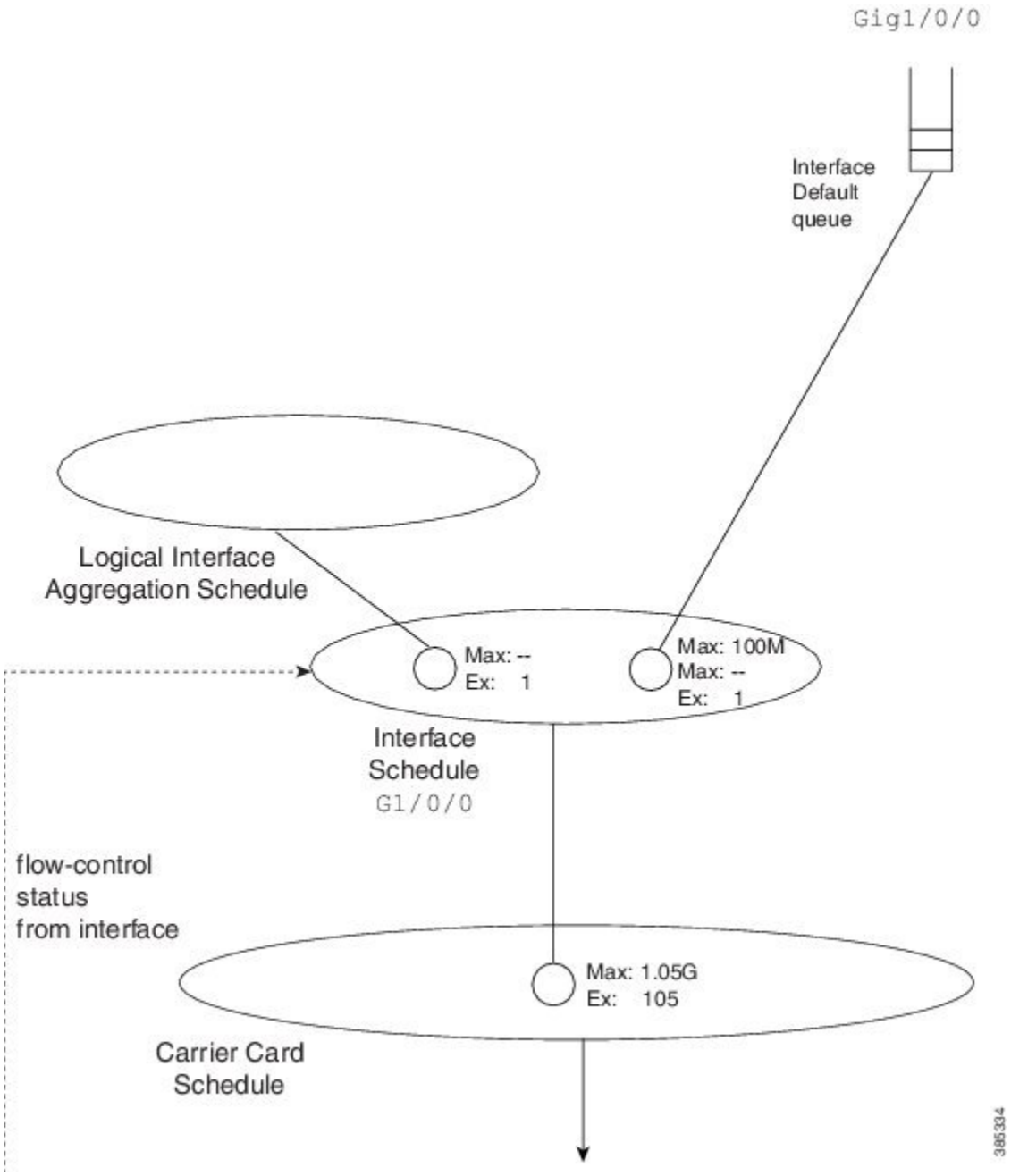


前述のように、子ポリシーは論理インターフェイス内の帯域幅の共有を定義します。通常、ここでは（音声などの）キューをクラス キュー(ポリシーマップ内のクラスによって定義された処理を含む）と呼び、このレイヤでのスケジュールをクラス レイヤ スケジュールと呼びます。.

親ポリシーでは、親シェーパー（最大：900m）と、暗黙の帯域幅の共有である'1'（Ex：1）を定義します。QoS設定では、この論理インターフェイスを既存のインターフェイス階層に接合する場所が明示的に指定されない（アタッチされていないスケジュールエントリに注意）ため、ルータは論理インターフェイスがどの物理インターフェイスに関連付けられているかを認識する必要があります。

VLAN 上のポリシーでは、どのインターフェイスが関連しているかは明らかです。サブインターフェイス設定に（論理インターフェイス）ポリシーを適用します。他のインターフェイスの種類（トンネルインターフェイスなど）については、ルーティング情報を調べてその特定の論理インターフェイスに対して、出力物理インターフェースを指定する必要があります。

図 *48 :* 既存のインタフェース階層（接合前）

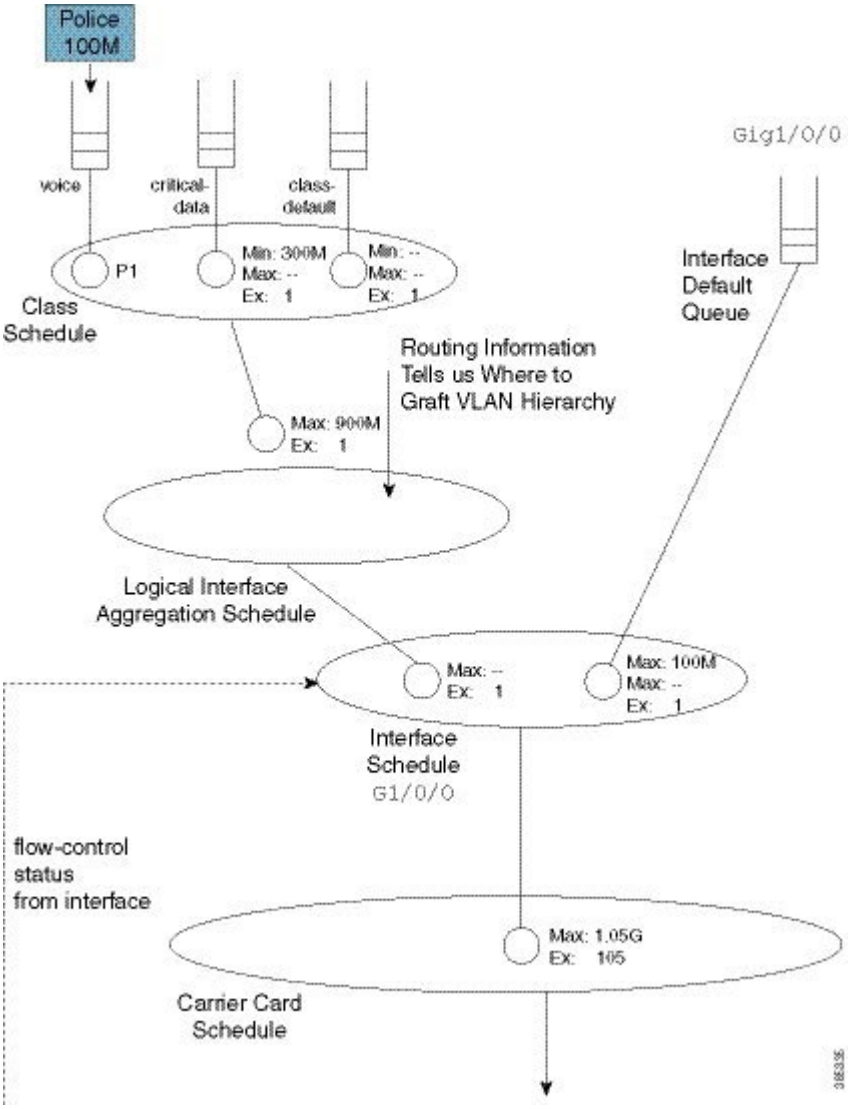


どのインターフェースが関連していることがわかったら、そのインターフェースの階層を変更できます。最初に、「親ポリシーでのシェーピング（または子ポリシーでのキューイング）」で定義された論理インターフェイス階層の接合場所として機能するスケジュール（論理インターフェイス集約）を作成します。

当初、インターフェース スケジュールには単一の子、インターフェース デフォルト キューしかありませんでした。今日では2番目の子である、論理インターフェイス集約スケジュールを

作成できるようになりました。 このスケジュールの超過ウェイトが、インターフェイスのデフォルトキューのウェイトとどのように一致するかを確認します。通常どおり、デフォルトの「1」になります。

図 *49 :* 既存のインターフェース階層（接合後）



親ポリシーでのシェーピングでは、子ポリシーを含む class-default しかありません。

```
policy-map parent100
  class class-default
    shape average 900m
    service-policy child100
```

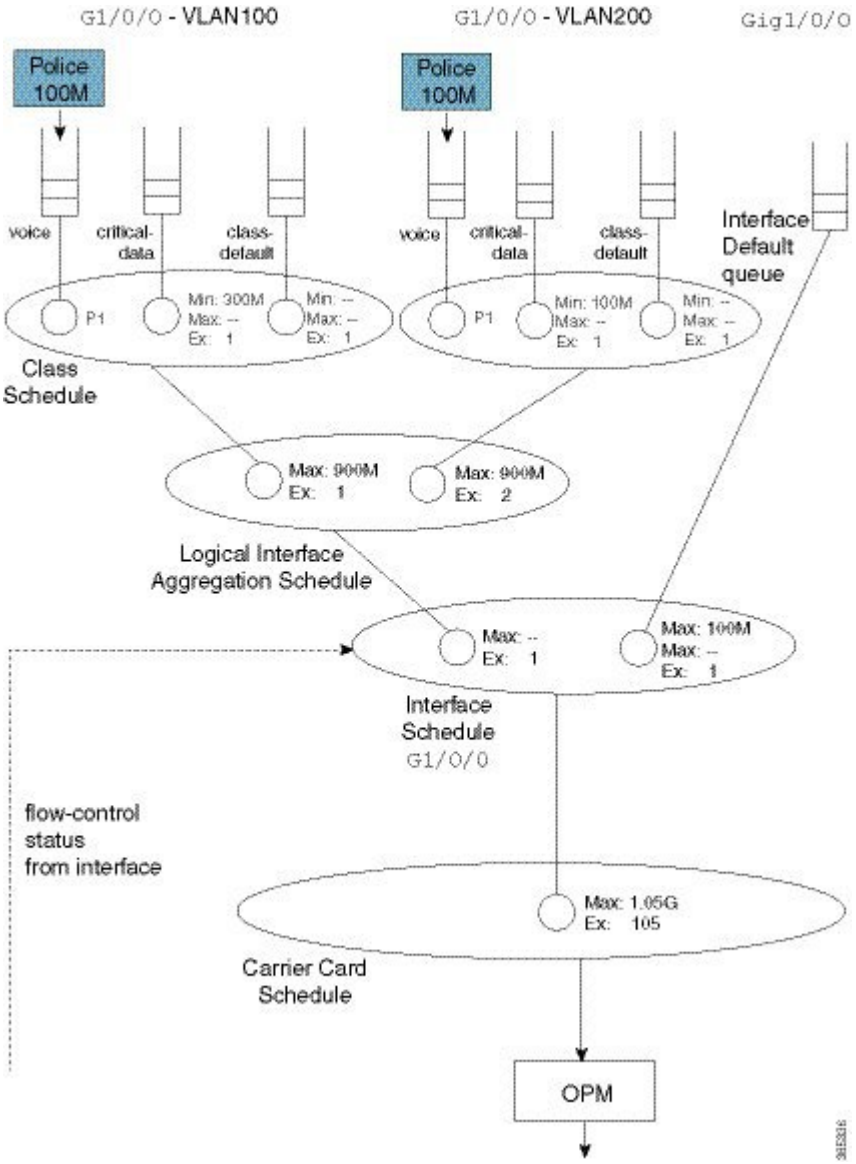
これは特殊なケースで、このポリシーのスケジュールを作成するのではなく、単にスケジュール エントリを定義します。このエンティティを *collapsed class-default* と呼びます.

この概念の重要性を理解するために、別のVLAN（VLAN200）にポリシーを追加してみましょう。（トピックの冒頭に記載されている`policy-map parent100`に関連して、アスタリスクを追加しました）：

```
policy-map child200
  class voice
    priority
    police cir 100m
  class critical-data
    bandwidth 100000
!
policy-map parent200
  class class-default
    shape average 900m                    ****
    bandwidth remaining ratio 2
    service-policy child200
!
int g1/0/0.200
  encaps dot1q 200
  service-policy out parent200
```

完全なスケジューリング階層は、次のようになります。

図 *50 :* 輻輳を処理し、帯域幅の浪費を回避するための完全な階層型スケジューリング フレームワーク



2 番目の親ポリシー（VLAN200 へのポリシー）では、帯域幅余剰比率に 2 を指定し、VLAN 間の均等性を制御しています。 Qos スケジューリングの章で説明したように、フラット ポリシーの親ポリシーにはピアが存在します。これにより、**bandwidth remaining ratio** または **bandwidth remaining percent** コマンドを使用して超過重量を指定することができます。「親ポリシーでのシェーピング」コンストラクトでは、ピアは存在しません. QoS ポリシーマップを設定すると、QoS は論理インターフェイス集約スケジュールでピアとして具体化するものを認識できません。したがって、**bandwidth remaining ratio** も **bandwidth remaining percent** コマンドもサポートされていません。

この完全なスケジューリング階層は、シスコ モジュラ QoS CLI（MQC）および階層型スケジューリング フレームワーク（HQF）の利点を真に強調しています。どのようなインタフェースでも、階層は決定論的です。次に転送されるパケットは明確にわかっています。すべての輻

輳ポイントを処理するスケジュールがあるため、輻輳が発生する可能性がある場所に関係なく、帯域幅は無駄になりません。

論理インターフェイス上のポリシーの利点

ポリシーマップを論理インターフェイスに適用する機能には、スケーリングされた環境での管理と簡単な設定という大きな利点があります。論理インターフェイスごとに、ポリシーマップを再利用または作成できます。つまり、イーサネットタイプのインターフェイスに設定されている 1000 個の VLAN それぞれにポリシーマップを適用できます。個々の論理インターフェイスの QoS 統計データを確認するには、 **show policy-map interface** *interface-name* を発行します。

利点と危険性は紙一重となる場合があります。利用可能な物理帯域幅が親シェーパーの合計を超える場合は、単一の論理インターフェイスを単独で確認するだけで十分です。 ただし、親シェーパーの合計が使用可能な物理帯域幅を超える場合は、論理インターフェイス間の競合、および個々のインターフェイスに対して本当に保証されている帯域幅を考慮する必要があります。個別のインターフェイスを単独で確認すると、誤解を招く可能性が高くなります。

複数のポリシー定義と制限

複数ポリシー（*MPOL*）を使用し、ポリシーマップが論理インターフェイスに適用されている場合に、その論理インターフェイスがバインドされている物理インターフェイス（VLAN サブインターフェイスや物理イーサネットインターフェイスなど）にポリシーマップを同時に適用している状況について説明します。

MPOL は、ポリシーマップが同じ物理インターフェイスにバインドされている異なる論理インターフェイス タイプに付加されているインスタンスを参照することもできます。たとえば、VLAN サブインターフェイスとトンネル インターフェイスの両方に適用されたポリシーがあるとします。この場合、両方とも同じ物理インターフェイスから出ます。

現在、ASR 1000 シリーズ アグリゲーション サービス ルータは MPOL の非常に限られた実装に対応しています。論理インターフェイスに適用されたポリシーマップがある場合、物理インターフェイスに適用できる唯一のポリシーは、次の例のように class-default とシェーパーのみが設定されたフラットです。このトポロジは、（プロバイダーからの）サービス レートが物理アクセス レートと異なる場合に役立ちます。たとえば、GigabitEthernet インターフェイスを介してプロバイダと接続しており、200 Mbps のサービスのみを支払っているとします。サービス プロバイダはそのレートを超えるトラフィックを監視するため、送信するすべてのものを 200 Mbps にシェーピングし、その帯域幅をローカルに割り当てる必要があります。

（注） ポリシーを論理インターフェイスに適用する前に、物理インターフェイスに適用する必要があります。 さらに、単一の物理インターフェイスにバインドされた複数の論理インターフェイス タイプにポリシーマップを適用することはできません。

それでは、2 つの VLAN サブインターフェイスに適用されたポリシーの前の例（「親ポリシーでのシェーピング/子ポリシー上のキューイング （125 ページ）」を参照）に戻り、物理イン

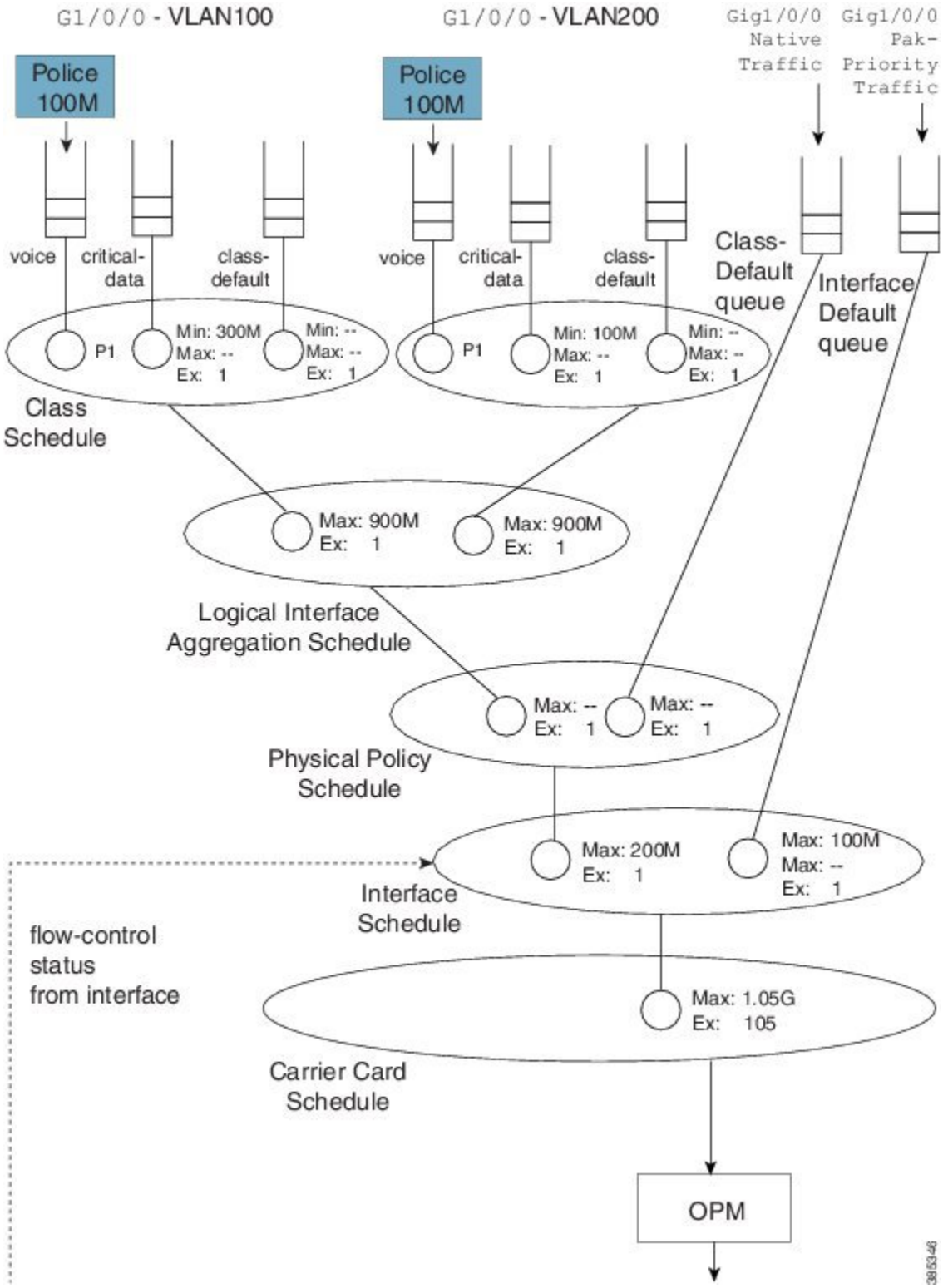
ターフェイスに 200 Mbps のシェーパーを追加します。完全な設定は次のようになります。アスタリスクは、これと以前の設定との違いを示しています。

```
policy-map physical-shaper                                  ****
  class class-default                                       ****
    shape average 200m                                      ****
 !
policy-map child100
  class voice
    priority
    police cir 100m
  class critical-data
    bandwidth 300000
!
policy-map child200
  class voice
    priority
    police cir 100m
  class critical-data
    bandwidth 100000
!
policy-map parent100
  class class-default
    shape average 900m
    service-policy child100
!
policy-map parent200
  class class-default
    shape average 900m
    bandwidth remaining ratio 2
    service-policy child200
!
! Note - must attach physical policy before logical policies
!
int g1/0/0                                                  ****
  service-policy output physical-shaper                     ****
!
int g1/0/0.100
  encaps dot1q 100
  service-policy out parent100
!
int g1/0/0.200
  encaps dot1q 200
  service-policy out parent200
```

物理インターフェイスを介して送信されるユーザー トラフィックに使用されるキューと同様に、別のスケジュールについて学んだことを忘れないでください。論理インターフェース集約スケジュールは、インターフェース スケジュールの直接の子としてではなく、物理ポリシー スケジュールの子として作成されました。これで、論理インターフェイスを通過するトラフィックと物理インターフェイスを通過するユーザ トラフィックの組み合わせは、200 Mbps にシェープされました。

完全なスケジューリング階層は次のようになります。

図 *51* : 物理ポリシー スケジュールの子としての論理インターフェイス集約の作成



階層型ポリシーマップ

前の章では、ポリシーマップが論理インターフェイスに適用されている場合の階層の作成方法を学びました。2 番目のアプローチは、階層型ポリシーマップを使用し、希望する階層を明示的に作成します。この方法を使用すると、ある程度の柔軟性が得られますが、ある程度の拡張性は失います。（論理インターフェイスに関するポリシーを使用すると、スケーリングされた環境で管理できることを思い出してください。）ASR 1000 シリーズ アグリゲーション サービス ルータは、ポリシーマップで最大 1,000 のクラスをサポートします。つまり、表記できる論理インターフェイスの最大数は 1,000 です。

階層型ポリシーマップ内のクラスに属するには、パケットは子および（すべての）親の分類規則と一致する必要があります。 以前の VLAN の例では、親クラスで VLAN ID ベースの分類を使用し、子クラスで DSCP ベースの分類を使用する方法を学びました。

次の設定では、MPOL 物理シェーパー（「複数のポリシー定義と制限（132 ページ）」を参照）と同じような動作を実現する方法を示しています。 ここでは、3 レベルの階層型ポリシーマップ（サポートしている最大レイヤ数）を使用します。

親ポリシーには class-default しかありません。つまり、インターフェイスを通過するすべてのトラフィックはこのクラスに属します。

```
policy-map physicalshaper
  class class-default
    shape average 200m
    service-policy vlansharing
```

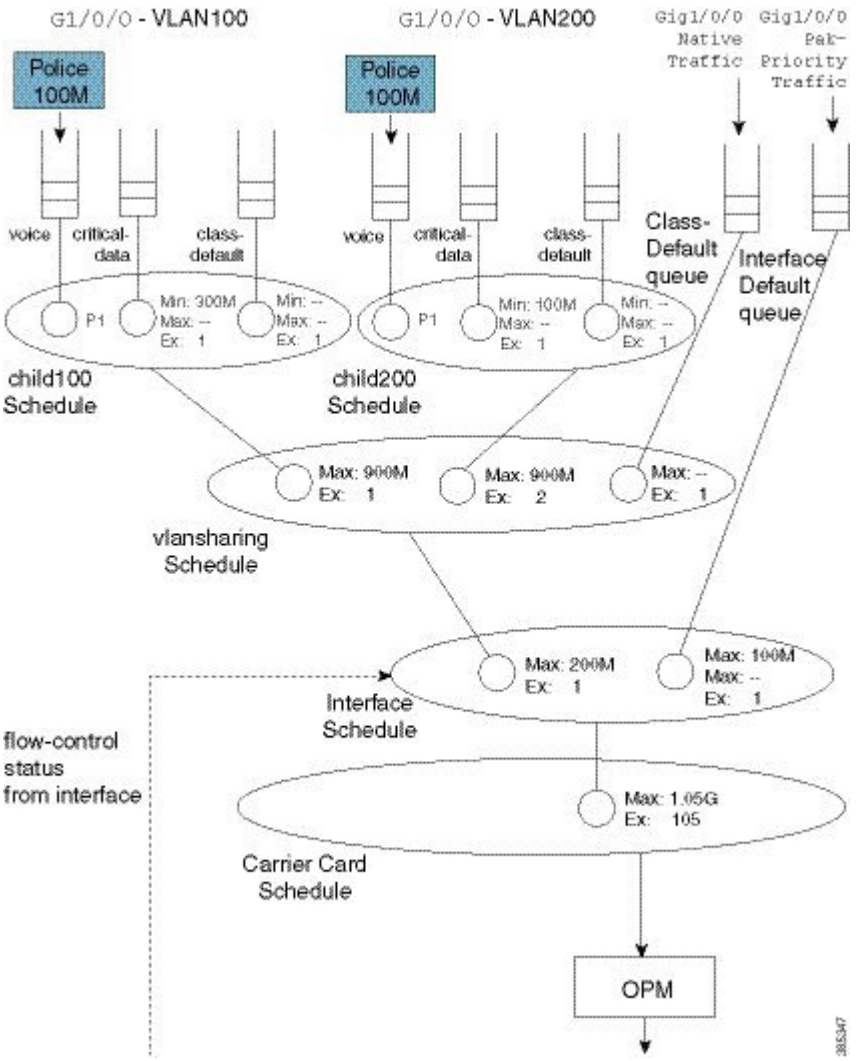
子レベルには VLAN ベースの分類があります。 VLAN 100 または VLAN 200 に属するトラフィックは、ユーザ定義クラスの 1 つに分類されます。（さらに、このポリシーには、他の VLAN からのトラフィックまたは VLAN タグなしのトラフィックをキャプチャする暗黙の class-default があります。）各 VLAN クラスには、DSCP に基づいてトラフィックをさらに分類するポリシーがあります。

```
class-map vlan100
  match vlan 100
class-map vlan200
  match vlan 200
class-map voice
  match dscp ef
class-map critical-data
  match dscp af21
        !
policy-map child100
 class voice
   priority
   police cir 100m
 class critical-data
   bandwidth 300000
!
policy-map child200
 class voice
   priority
   police cir 100m
 class critical-data
   bandwidth 100000
```

```
!
policy-map vlansharing
 class vlan100
   shape average 900m
   bandwidth remaining ratio 1
   service-policy child100
 class vlan200
   shape average 900m
   bandwidth remaining ratio 2
   service-policy child200
!
policy-map physicalshaper
   class class-default
     shape average 200m
    service-policy vlansharing
 !
int g1/0/0
  service-policy output physicalshaper
```

上記の設定に基づいて作成された階層は、次のようになります。

図 *52 :* 階層を明示的に作成するための階層型ポリシーマップ



この階層を前の MPOL の例（図 25）と比較すると、若干の違いがあることがわかります。

まず、ネイティブ インターフェイス トラフィック（VLAN 100 と 200 のどちらでもないトラフィック）が、各 VLAN のスケジュール エントリと vlansharing スケジュールを共有するようになりました。MPOLの例では、ネイティブトラフィックは、すべての（両方の）VLAN（使用可能帯域幅の 1/2）と同等の帯域を受け取りました。 これとは対照的に、この階層では、同じスケジュールで VLAN と競合するため、使用可能な帯域幅の 1/(1 + 2 + 1) だけが保証されます。

次に、物理インターフェイス上の単一のポリシーマップでは、単一の VLAN の統計データしか見ることができなくなりました。MPOL の例とこのコードを比較してください。

```
int g1/0/0
  service-policy output physical-shaper
!
int g1/0/0.100
  encaps dot1q 100
```

```
  service-policy out parent100
!
int g1/0/0.200
  encaps dot1q 200
  service-policy out parent200
```

次を使用すると：

```
int g1/0/0
  service-policy output physicalshaper
```

show policy-map interface GigabitEthernet1/0/0 コマンドの出力には、階層型ポリシーマップのすべてのレベルが反映されます。

階層型ポリシーマップにより、論理インターフェイス上のポリシーマップでは実現不可能な柔軟性が強化されます。次の例は、この動作について説明しています。

例 1 異なるトラフィック クラスにキューを追加する

MPOL の例（および以下のコード）では、物理インターフェイス ポリシーには class-default とそのクラスのシェーパーしか含めることがないことをを学びました。

```
policy-map physical-shaper
  class class-default
    shape average 200m
```

つまり、ネイティブインターフェイスを介して転送された固有のクラスのトラフィック（VLAN タグなしのトラフィック）に対して異なる方法で処理することはできません。

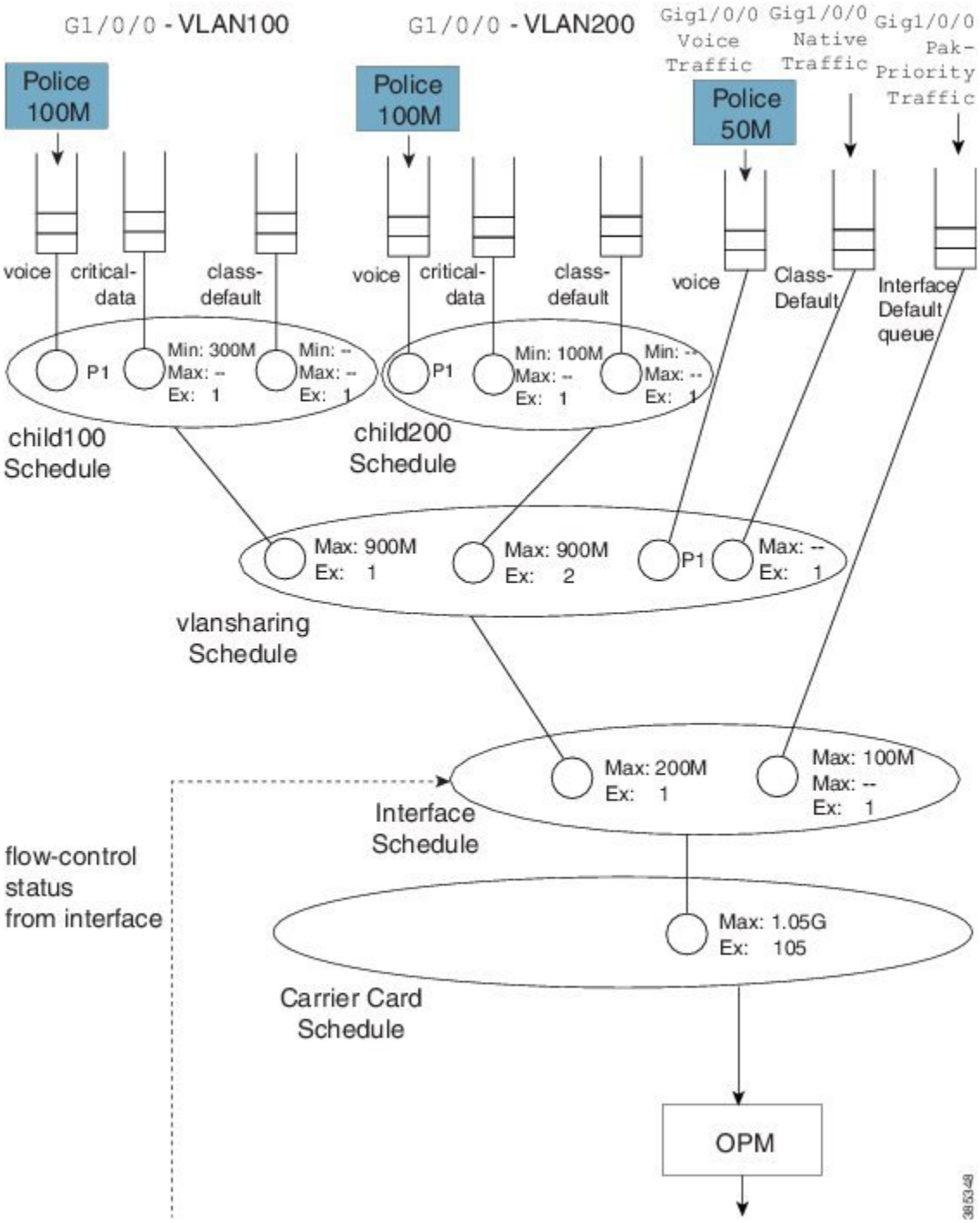
これとは対照的に、階層コンストラクトでは、（物理インターフェイスを介して）転送するトラフィックのクラスごとにキューを追加できます。たとえば、物理インターフェイス上の音声トラフィックにプライオリティ クラスを追加する場合は、vlansharing ポリシーマップを次のように変更できます（アスタリスクを参照）。

```
class-map vlan100
  match vlan 100
class-map vlan200
  match vlan 200
class-map voice
  match dscp ef
class-map critical-data
  match dscp af21
        !
policy-map child100
 class voice
   priority
   police cir 100m
 class critical-data
   bandwidth 300000
!
policy-map child200
 class voice
   priority
   police cir 100m
 class critical-data
   bandwidth 100000
!
```

```
policy-map vlansharing
  class vlan100
    shape average 900m
    bandwidth remaining ratio 1
    service-policy child100
  class vlan200
    shape average 900m
    bandwidth remaining ratio 2
    service-policy child200
  class voice                                  ****
    priority                                   ****
    police cir 50m                             ****
!
policy-map physicalshaper
  class class-default
    shape average 200m
    service-policy vlansharing
 !

int g1/0/0
  service-policy output physicalshaper
```

この設定の階層は、次のようになります。

図 *53 :* 階層コンストラクトで異なるトラフィック クラスのキューを表す



EF の DSF コードポイントでマークされているが、VLAN ID 100 または 200 でタグ付けされていないトラフィックをキャプチャする新しいキャプチャに留意してください。

この階層では、ローカル キューからの P1 トラフィック（Gig1/0/0 音声トラフィック）が、VLAN共有スケジュール内の優先度伝播トラフィックと競合することに注意してください（「優

先度の伝搬の概念（100ページ）」を参照）。そのような場合には、プライオリティが設定されたローカルエントリは優先度伝播トラフィックの前に処理されます。つまり、物理インターフェイス（Gig1/0/0）からの音声パケットは、VLAN 100 または 200 からの音声パケットよりわずかに高い優先順位を持ちます。他のクラスのキューが消費されるのを回避するために、プライオリティ キューでアドミッション コントロールを使用します。

例 2 異なる論理インターフェイス タイプへのポリシーの適用

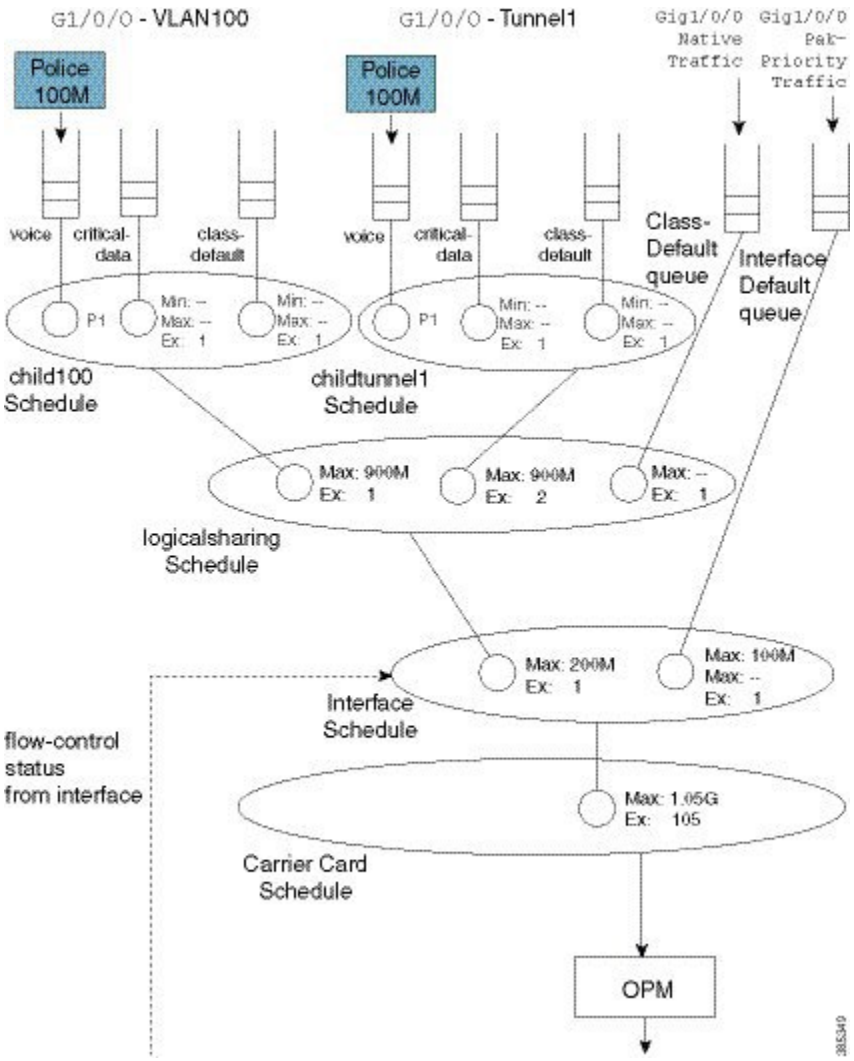
「論理インターフェイスに適用されたポリシーマップ（123ページ）」では、同じ物理インターフェイス上の異なる論理インターフェイス タイプにポリシーを適用できないことを学びました。この制限は、階層型 class-map には適用されません。

両方とも同じ物理インターフェイスを通過するトンネルで、VLAN 100 用に 1 つの子、QoS 用に 1 つの子が必要だとします。 同じポリシーマップ内で、アクセス リストを使用してトンネル トラフィックを分類し、VLAN ID を使用して VLAN トラフィックを分類できます（アスタリスクを参照）。

```
ip access-list extended tunnel1traffic
  permit ip host 192.168.1.1 host 10.0.0.1
!
class-map vlan100
  match vlan 100
class-map tunnel1traffic
  match access-group name tunnel1traffic
!
class-map voice
  match dscp ef
class-map critical-data
  match dscp af21
        !
policy-map child
  class voice
    priority
    police cir 100m
  class critical-data
    bandwidth remaining ratio 1
!
policy-map logicalsharing                  ****
  class vlan100
    shape average 900m
    bandwidth remaining ratio 1
    service-policy child
  class tunnel1traffic
    shape average 900m
    bandwidth remaining ratio 2
    service-policy child
!
policy-map physicalshaper
  class class-default
    shape average 200m
    service-policy vlansharing
 !

int g1/0/0
  service-policy output physicalshaper
```

この設定の階層は、次のようになります。

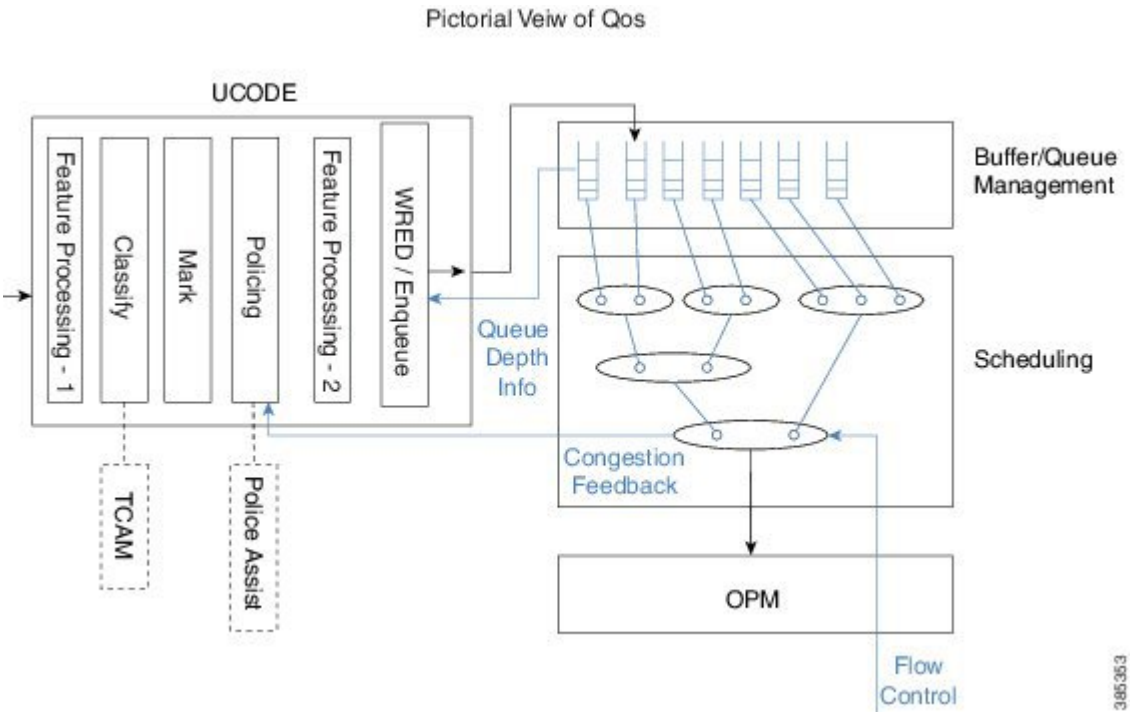
図 *54* : 異なる論理インターフェイス タイプへのポリシーの適用



オーバーヘッド アカウンティングにおける留意点

ポリシングの章では、ポリシング長（ポリサーが設定済みポリシング レートへの適合性を評価するときに、パケットの長さを認識する方法。「ポリサー レート計算（オーバーヘッド アカウンティング）に含まれるもの（311ページ）」を参照）の概念について学びました。同様に、スケジューリングの章では、スケジューリング長（設定済みのスケジューラ レートへの適合性を評価する際のパケットの長さを考慮する方法。「スケジューリング レート計算（オーバーヘッド アカウンティング）に含まれるもの （45 ページ）」を参照）の概念について学びました。慣例により、どちらの場合も、レイヤ 2 ヘッダーとデータグラムの長さ、および CRC またはパケット間のオーバーヘッドを除外します。

階層型スケジューリング構成では、ポリシングとスケジューリングの長さが異なる場合があります。 これを理解するために、機能の実行順序について考慮します。



ASR 1000 シリーズアグリゲーション サービス ルータでは、キューイングとスケジューリングはハードウェアで実行されます。パケットをエンキューした後、ハードウェアにより制御が実行され、それ以上の処理は実行されません。パケットにはすべてのヘッダーがあり、ワイヤを通過する準備ができている必要があります。予想どおり、非キューイング機能は、処理要素のうちの 1 つのマイクロコードで実行され（場合によってはハードウェア支援を用いて）ます。

2 つのシナリオについて考えてみます。

GRE トンネルでの QoS キューイング ポリシーの設定

（最終的に外部 IP/GRE ヘッダーにカプセル化される）着信 IP パケットを分類するときは、元の IP パケットのみを確認します。その結果、分類統計データは、その時点で欠落しているため、外側の IP/GRE ヘッダーを除外します。 図で示されているように、ここではポリサーのマーキングと評価を行います。分類長と同様に、ポリシング長には外部 IP/GRE ヘッダーも出力レイヤ2ヘッダーも含まれません。パケットがどの物理インターフェイスまたはカプセル化タイプを出力するのかまだわからないためです。QoS 非キューイング機能の後、外部 IP/GRE ヘッダーと最終的な出力インターフェイス用の適切なレイヤ2ヘッダーを追加して、パケットの処理を続けます。すべての処理が終了したら、パケットを WRED/Enqueue ブロックにパスします。この動作により、すべてのヘッダーが追加されたパケットが、ハードウェア内の適切な出力キューに配置されます。これで、スケジューリング長には、外側の IP/GRE とレイヤ2ヘッダーが含まれるようになりました。

物理出力インターフェイスでの QoS ポリシーの設定

結果は異なります。トンネル上の機能を確認すると、QoS が設定されていないため、機能処理に進みます。QoS ポリシーに到達する前に、すべてのトンネル処理を完了し、出力ヘッダーを追加します。これで、分類統計データとポリシング長には外部ヘッダーが含まれるようになり、ポリシングとスケジューリング長が一致するようになります。

確認

すべての QoS 設定作業において、階層型スケジューリングの設定を検証するための主要ツールは、**show policy-map interface** *interface-name* コマンドです。このコマンドの出力は、構成を階層化した方法を反映して、階層的に編成されています。

たとえば、物理インターフェイスに付加された階層型ポリシーでは、**show policy-map interface** インターフェイス名 **| include Class** を使用してその階層を表示します。

```
show policy-map int g1/0/0 | inc Class
```

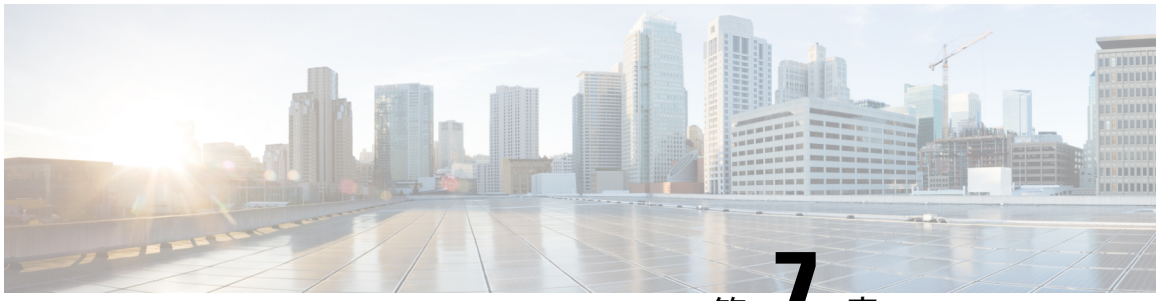
```
    Class-map: class-default (match-any)
        Class-map: vlan100 (match-all)
            Class-map: voice (match-all)
            Class-map: critical-data (match-all)
            Class-map: class-default (match-any)
        Class-map: vlan200 (match-all)
            Class-map: voice (match-all)
            Class-map: critical-data (match-all)
            Class-map: class-default (match-any)
        Class-map: vlan300 (match-all)
            Class-map: voice (match-all)
            Class-map: class-default (match-any)
        Class-map: voice (match-all)
        Class-map: class-default (match-any)
```

この例では、インターフェイス GigabitEthernet1/0/0 に3 レベルの階層型ポリシーを適用しました。 class-map のインデントはその階層を表します。子ポリシーを含むクラス内の `Service-policy: <ポリシーマップ名>` は、次にインデントされているセクションが子ポリシーに関連することを示します。

```
  Class-map: vlan100 (match-all)
          0 packets, 0 bytes
          5 minute offered rate 0000 bps, drop rate 0000 bps
          Match: vlan  100
          Queueing                                                  ****
          queue limit 3748 packets                                  ****
          (queue depth/total drops/no-buffer drops) 0/0/0           ****
          (pkts output/bytes output) 0/0
          shape (average) cir 900000000, bc 3600000, be 3600000
          target shape rate 900000000
          bandwidth remaining ratio 1

          Service-policy : child100

            queue stats for all priority classes:
              Queueing
              queue limit 512 packets
              (queue depth/total drops/no-buffer drops) 0/0/0
              (pkts output/bytes output) 0/0
```

```
Class-map: voice (match-all)
  0 packets, 0 bytes
  5 minute offered rate 0000 bps, drop rate 0000 bps
  Match:  dscp ef (46)
  Priority: Strict, b/w exceed drops: 0

  police:
      cir 100000000 bps, bc 3125000 bytes
    conformed 0 packets, 0 bytes; actions:
      transmit
    exceeded 0 packets, 0 bytes; actions:
      drop
    conformed 0000 bps, exceeded 0000 bps

Class-map: critical-data (match-all)
  0 packets, 0 bytes
  5 minute offered rate 0000 bps, drop rate 0000 bps
  Match:  dscp af11 (10)
  Match:  dscp af21 (18)
  Queueing
  queue limit 1249 packets
  (queue depth/total drops/no-buffer drops) 0/0/0
  (pkts output/bytes output) 0/0
  bandwidth 300000 kbps

Class-map: class-default (match-any)
  0 packets, 0 bytes
  5 minute offered rate 0000 bps, drop rate 0000 bps
  Match: any

  queue limit 3748 packets
  (queue depth/total drops/no-buffer drops) 0/0/0
  (pkts output/bytes output) 0/0
```

階層型スケジューリングを含むポリシーの**show**コマンド出力については、親クラス内のキューに関連するすべての情報は有効ではないことに注意してください（上記の例ではアスタリスクで強調表示されています）。**show policy-map interface** コマンドの出力形式は、IOS がキューの階層をソフトウェアに実際に実装したときに作成されました。 ASR 1000 シリーズ アグリゲーション サービス ルータ ハードウェアは、リーフにのみ存在するスケジュールとキューの階層を実装します。IOS コントロールプレーンにより、キュー制限がまだ計算され、表示されますが、それを使用することはありません。したがって、この値をチューニングしても意味がありません。



第 7 章

レガシー QoS コマンドの廃止予定

これらの隠しコマンドによって提供される機能は、モジュラ QoS CLI（MQC）経由で提供される同様の機能に置き換えられています。MQC は、シスコ プラットフォームで QoS を設定するための、プラットフォームに依存しないコマンドです。これは、今すぐ、トラフィック クラスを定義して、それらのクラスを含むトラフィック ポリシーを作成し、それらのポリシーを必要なインターフェイスに適用することによって、QoS をプロビジョンする必要があることを意味します。このマニュアルでは、隠しコマンドとそれらの代替 MQC コマンドを一覧表示します。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「Bug Search Tool」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

レガシー QoS コマンドの廃止予定に関する情報

MQC を使用して適用した QoS 機能

MQC 構造を使用すると、トラフィック クラス（クラス マップとも呼ぶ）を定義したり、トラフィック ポリシー（ポリシーマップとも呼ぶ）を作成したり、インターフェイスへトラフィック ポリシーを適用したりすることができます。これは、次の 3 つレベルのステップから構成されています。

1. **class-map** コマンドを使用して、トラフィック クラスを定義します。トラフィック クラスは、トラフィックの分類に使用します。
2. **policy-map** コマンドを使用して、トラフィック ポリシーを作成しますトラフィック ポリシーには、1 つのトラフィック クラスと、そのトラフィック クラスに適用する 1 つ以上の QoS 機能が含まれます。トラフィック ポリシー内の QoS 機能によって、分類されたトラフィックの処理方法が決まります。
3. **service-policy** コマンドを使用して、インターフェイスにトラフィック ポリシーを適用します。

ステップ 1 および 3 には、レガシー QoS の隠しコマンドが含まれていません。つまり、このドキュメントの対象範囲に入っていません。この 2 つのステップの詳細については、『*Quality of Service Solutions Configuration Guide*』の「Applying QoS Features Using the MQC」モジュールを参照してください。

隠しレガシー コマンド

次の表に、隠しコマンドまたは削除済みコマンドのリストを示します。また、代替コマンド（または一連のコマンド）も示します。

表 *5:* 代替コマンドへの隠しコマンド、削除済みコマンド、またはサポート対象外のコマンドのマップ

隠しコマンド、削除済みのコマンド、またはサポート対象外のコマンド	代替 MQC コマンド シーケンス
重み付けランダム早期検出または分散重み付けランダム早期検出のパラメータ グループの設定	

隠しコマンド、削除済みのコマンド、またはサポート対象外のコマンド	代替 MQC コマンド シーケンス

コマンド
• random-detect-group
• random-detect（VC 単位）
（注） このコマンドは、Cisco IOS リリース 15.0(1)S ではサポートされていません。
コマンドの使用方法
`Router(config)# random-detect-group group-name [dscp-based|prec-based]`
`Router(config)# interface atm type number`
`Router(config-if)# pvc [name] vpi/vci`
`Router(config-if-atm-vc)# random-detect [attach group-name ]`
	コマンドの使用方法
	なし（この機能は存在しません）。

重み付けランダム早期検出の設定	

隠しコマンド、削除済みのコマンド、またはサポート対象外のコマンド	代替 MQC コマンド シーケンス

コマンド
• random-detect
• random-detect dscp
• random-detect（dscp-based キーワード）
• random-detect flow
• random-detect exponential-weighting-constant
• random-detect（prec-based キーワード）
• random-detect precedence
 	
コマンドの使用方法
 	
`Router(config)# interface type number`
`Router(config-if)# random-detect [number]`
`Router(config-if)# random-detect exponential-weighting-constant exponent`
`Router(config-if)# random-detect flow`
`Router(config-if)# random-detect precedence {precedence|rsvp} min-threshold max-threshold max-probability-denominator`
`Router(config-if)# random-detect prec-based`
`Router(config-if)# random-detect dscp-based`
`Router(config-if)# random-detect dscp dscp-value min-threshold max-threshold[max-probability-denominator]`
	コマンドの使用方法
	
	`Router(config)# policy-map`
	
	`policy-map-name`
	`Router(config-pmap)# class class-default`
	`Router(config-pmap-c)# random-detect dscp dscp-value min-threshold max-threshold[ mark-probability-denominator]`
	`Router(config-pmap-c)# random-detect clp clp-value min-threshold max-threshold[mark-probability-denominator]`
	`Router(config-pmap-c)# random-detect cos cos-value min-threshold max-threshold[mark-probability-denominator]`
	`Router(config-pmap-c)# random-detect discard-class discard-class-value min-threshold max-threshold[ mark-probability-denominator]`
	`Router(config-pmap-c)# random-detectprecedence ip-precedence min-threshold max-threshold[mark-probability-denominator]`
	`Router(config-pmap-c)# random-detect precedence-based`
	`Router(config-pmap-c)# random-detect ecn`
	`Router(config-pmap-c)# random-detect exponential-weighting-constant exponent`
	`Router(config-pmap-c)# random-detect cos-based`
	`Router(config-pmap-c)# random-detect dscp-based`

隠しコマンド、削除済みのコマンド、またはサポート対象外のコマンド	代替 MQC コマンド シーケンス

コマンド
• random-detect flow
• random-detect flow average-depth-factor
• random-detect flow count
コマンドの使用方法
`Router(config)# interface type number`
`Router(config-if)# random-detect [number]`
`Router(config-if)# random-detect flow`
`Router(config-if)# random-detect flow count number`
`Router(config-if)# random-detect flow average-depth-factor scaling-factor`
	コマンドの使用方法
	なし（この機能は存在しません）。

帯域幅割り当ての設定	

コマンド
• max-reserved-bandwidth
コマンドの使用方法
`Router(config)# interface type number`
`Router(config-if)# max-reserved-bandwidth percentage`
	コマンドの使用方法
	`Router(config)# policy-map policy-map-name`
	`Router(config-pmap)# class class-default`
	`Router(config-pmap-c)# bandwidth{bandwidth-in-kbps |remaining percent percentage | percent percentage}`

カスタム キューイングの設定	

コマンド
• custom-queue-list
（注） このコマンドは、Cisco IOS リリース 15.0(1)S ではサポートされていません。
コマンドの使用方法
`Router(config)# interface type number`
`Router(config-if)# custom-queue-list[list-number]`
	コマンドの使用方法
	`Router(config)# policy-map policy-map-name`
	`Router(config-pmap)# class class-default`
	`Router(config-pmap-c)# bandwidth{ bandwidth-in-kbps |remaining percent percentage |percent percentage}`

プライオリティ キューイングの設定	

隠しコマンド、削除済みのコマンド、またはサポート対象外のコマンド	代替 MQC コマンド シーケンス

コマンド
• ip rtp priority
• ip rtp reserve
コマンドの使用方法
`Router(config)# interface type number`
`Router(config-if)# ip rtp priority starting-port-number port-range bandwidth`
`Router(config)# interface type number`
`Router(config-if)# ip rtp reserve lowest-udp-port range-of-ports [maximum-bandwidth] 1000`
	コマンドの使用方法
	`Router(config)# policy-map policy-map-name`
	`Router(config-pmap)# class class-name`
	`Router(config-pmap-c)# priority`

重み付け均等化キューイングの設定	

コマンド
• fair-queue (WFQ)
コマンドの使用方法（Cisco IOS リリース 15.0(1)S）
`Router(config)# interface type number`
`Router(config-if)# fair-queue`
コマンドの使用方法（Cisco IOS リリース 15.1(3)T）
`Router(config)# interfacetype number`
`Router(config-if)# fair-queue [congestive-` *discard-threshold* `[dynamic-queue-count [`*reserved-queue-count*`]]]`
	コマンドの使用方法（Cisco IOS リリース 15.0(1)S）
	`Router(config)# policy-map policy-map-name`
	`Router(config-pmap)# class class-default`
	`Router(config-pmap-c)# fair-queue`
	コマンドの使用方法（Cisco IOS リリース 15.1(3)T）
	`Router(config)# policy-map policy-map-name`
	`Router(config-pmap)# class class-default`
	`Router(config-pmap-c)# fair-queue[dynamic-queues ]`

インターフェイスへのポリシー グループの割り当て	

隠しコマンド、削除済みのコマンド、またはサポート対象外のコマンド	代替 MQC コマンド シーケンス

コマンド
• priority-group
（注） このコマンドは、Cisco IOS リリース 15.0(1)S ではサポートされていません。
コマンドの使用方法
`Router(config)# interface type number`
`Router(config-if)# priority-group list-number`
	コマンドの使用方法
	`Router(config)# policy-map policy-map-name`
	`Router(config-pmap)# class class-default`
	`Router(config-pmap-c)# priority`
	`Router(config-pmap-c)# priority bandwidth-in-kbps [burst-in-bytes]`
	`Router(config-pmap-c)# priority percent percent [burst-in-bytes]`
	`Router(config-pmap-c)# priority level level`
	`Router(config-pmap-c)# priority level level [bandwidth-in-kbps [burst-in-bytes]]`
	`Router(config-pmap-c)# priority level level[percent percent [burst-in-bytes]]`

切り替えられた PVC トラフィック シェーピング キューからの DE パケットを廃棄するためのしきい値の設定	

コマンド
• frame-relay congestion threshold de
コマンドの使用方法
`Router(config)# map-class frame-relay map-class-name`
`Router(config-map-class)# frame-relay congestion threshold de percentage`
	コマンドの使用方法
	`Router(config)# policy-map policy-map-name1`
	`Router(config-pmap)# class class-default`
	`Router(config-pmap-c)# random-detect discard-class-based`
	`Router(config-pmap-c)# random-detect discard-class discard-class min-threshold max-threshold`
	`Router(config-pmap-c)# exit`
	`Router(config-pmap)# exit`
	`Router(config)# policy-map shape`
	`Router(config-pmap)# class class-default`
	`Router(config-pmap-c)# shape average rate`
	`Router(config-pmap-c)# service-policy policy-map-name1`
	`Router(config-pmap-c)# exit`
	`Router(config-pmap)# exit`
	`Router(config)# policy-map policy-map-name2`
	`Router(config-pmap)# class class-name`
	`Router(config-pmap-c)# set discard-class`discard-class

仮想回線に対するフレーム リレー カスタム キューイングの設定	

コマンド
• frame-relay custom-queue-list
コマンドの使用方法
`Router(config)# map-class frame-relay map-class-name`
`Router(config-map-class)# frame-relay custom-queue-list list-number`
	コマンドの使用方法
	`Router(config)# policy-map policy-map-name`
	`Router(config-pmap)# class class-default`
	`Router(config-pmap-c)#`
	bandwidth{*bandwidth-in-kbps* | **remaining percent** *percentage* | **percent***percentage*}

隠しコマンド、削除済みのコマンド、またはサポート対象外のコマンド	代替 MQC コマンド シーケンス
フレーム リレー ECN ビットしきい値の設定	

コマンド
• frame-relay congestion threshold ecn
コマンドの使用方法
`Router(config)# map-class frame-relay map-class-name`
`Router(config-map-class)# frame-relay congestion threshold ecn percentage`
	コマンドの使用方法
	なし（この機能は存在しません）。
	最も近いものは、（ECN に基づかない）MQC トラフィック シェーピングです。
	`Router(config)# policy-map policy-map-name`
	`Router(config-pmap)# class class-default`
	`Router(config-pmap-c)# shape average rate`

フレーム リレー重み付け均等化キューイングの設定	

コマンド
• frame-relay fair-queue
コマンドの使用方法
`Router(config)# map-class frame-relay map-class-name`
`Router(config-map-class)# frame-relay fair-queue [discard-threshold [dynamic-queue-count[reserved-queue-count [buffer-limit]]]]`
	コマンドの使用方法
	`Router(config)# policy-map policy-map-name`
	`Router(config-pmap)# class class-default`
	`Router(config-pmap-c)# fair-queue`
	`Router(config-pmap-c)# fair-queue queue-limit packets`
	（注） **queue-limit** *packets* キーワードと引数のペアは、Cisco IOS リリース 15.1(3)T ではサポートされていません。

PVC 上でのフレーム リレー プライオリティ キューイングの設定	

コマンド
• frame-relay ip rtp priority
コマンドの使用方法
`Router(config)# map-class frame-relay map-class-name`
`Router(config-map-class)# frame-relay ip rtp priority starting-port-number port-range bandwidth`
	コマンドの使用方法
	`Router(config)# policy-map policy-map-name`
	`Router(config-pmap)# class class-name`
	`Router(config-pmap-c)# priority bandwidth-in-kbps [burst-in-bytes]`

マップ クラスに関連付けられた仮想回線へのプライオリティ キューの割り当て	

隠しコマンド、削除済みのコマンド、またはサポート対象外のコマンド	代替 MQC コマンド シーケンス

コマンド
• frame-relay priority-group
コマンドの使用方法
`Router(config)# map-class frame-relay`map-class-name
`Router(config-map-class)# frame-relay priority-group group-number`
	コマンドの使用方法
	`Router(config)# policy-map policy-map-name`
	`Router(config-pmap)# class class-default`
	`Router(config-pmap-c)# priority`
	`Router(config-pmap-c)# priority bandwidth-in-kbps [burst-in-bytes]`
	`Router(config-pmap-c)# priority percent percentage [burst-in-bytes]`
	`Router(config-pmap-c)# priority level level [percent percentage [burst-in-bytes]]`
	（注） **priority level** コマンドは、Cisco IOS リリース 15.1(3)T ではサポートされていません。

BECN に対するフレーム リレー レート調整の設定	

コマンド
• frame-relay adaptive-shaping（becn キーワード）
コマンドの使用方法
`Router(config)# map-class frame-relay map-class-name`
`Router(config-map-class)# frame-relay adaptive-shaping becn`
	コマンドの使用方法
	なし（この機能は存在しません）。最も近いものは、（BECN に基づかない）MQC トラフィック シェーピングです。
	`Router(config)# policy-map policy-map-name`
	`Router(config-pmap)# class class-default`
	`Router(config-pmap-c)# shape adaptive rate`

ForeSight メッセージに対するフレーム リレー レート調整の設定	

コマンド
• frame-relay adaptive-shaping（foresight キーワード）
コマンドの使用方法
`Router(config)# map-class frame-relay map-class-name`
`Router(config)# frame-relay adaptive-shaping foresight`
	コマンドの使用方法
	なし（この機能は存在しません）。

BECN としてのフレーム リレー トラフィック シェーピング FECN のイネーブル化	

隠しコマンド、削除済みのコマンド、またはサポート対象外のコマンド	代替 MQC コマンド シーケンス

コマンド
• frame-relay fecn-adapt
コマンドの使用方法
`Router(config)# map-class frame-relay map-class-name`
`Router(config-map-class)#frame-relay fecn-adapt`
	コマンドの使用方法
	なし（この機能は存在しません）。最も近いものは、（FECN/BECNに基づかない）MQCトラフィック シェーピングです。
	`Router(config)# policy-map policy-map-name`
	`Router(config-pmap)# class class-default`
	`Router(config-pmap-c)# shape average rate`

フレーム リレー拡張ローカル管理インターフェイスの設定	

コマンド
• frame-relay qos-autosense
（注） このコマンドは、Cisco IOS リリース 15.0(1)S では隠されていません。
コマンドの使用方法
`Router(config)# interface type numberRouter(config-if)#no ip address`
`Router(config-if)# encapsulation frame-relay`
`Router(config-if)# frame-relay lmi-typeansi`
`Router(config-if)# frame-relay traffic-shaping`
`Router(config-if)# frame-relay qos-autosense`
	コマンドの使用方法
	なし（この機能は存在しません）。

フレーム リレー最小認定情報レート（MINCIR）の設定	
コマンド • frame-relay mincir **コマンドの使用方法** `Router(config)# frame-relay mincir {in | out} bps`	**コマンドの使用方法** なし（この機能は存在しません）。
相手先固定接続（PVC）に対するフレーム リレー プライオリティの設定	

隠しコマンド、削除済みのコマンド、またはサポート対象外のコマンド	代替 MQC コマンド シーケンス
コマンド • frame-relay interface-queue **コマンドの使用方法** `Router(config)# interface type number`Router(config-if)#**no ip address** Router(config-if)# frame-relay interface-queue priority 10 20 30 40	**コマンドの使用方法** Router(config)# **policy-map** policy-map-name Router(config-pmap)# **class class-default** Router(config-pmap-c)# **priority** Router(config-pmap)# **class class-default** Router(config-pmap-c)# **priority**
フレーム リレー トラフィック シェーピングの設定	
コマンド • frame-relay bc • frame-relay be • frame-relay cir （注） Cisco IOS リリース 15.1(3)T では、これらのコマンドは隠されていませんが、（PVC ではなく）SVC に対してのみ有効です。 **コマンドの使用方法** Router(config)# **map-class frame-relay** map-class-name Router(config-map-class)# frame-relay bc {in | out} committed-burst-size-in-bits Router(config-map-class)# frame-relay be {in | out} excess-burst-size-in-bits Router(config-map-class)# frame-relay cir {in | out} bits-per-second	**コマンドの使用方法** Router(config)# **policy-map** policy-map-name Router(config-pmap)# **class class-default** Router(config-pmap-c)# shape average *rate*
VC でのフレーム リレー トラフィック シェーピングの設定	
コマンド • frame-relay traffic-rate **コマンドの使用方法** Router(config)# **map-class frame-relay**map-class-name Router(config-map-class)# traffic-rate average [peak]	**コマンドの使用方法** Router(config)# **policy-map** policy-map-name Router(config-pmap)# **class class-default** Router(config-pmap-c)# shape average *rate* Router(config-pmap-c)# **service-policy output traffic-rate** service-policy output traffic-rate

隠しコマンド、削除済みのコマンド、またはサポート対象外のコマンド	代替 MQC コマンド シーケンス
インターフェイスまたは VC のキュー内部のパケットのコンテンツを表示します。	
コマンド • show queue **コマンドの使用方法** `Router# show queue interface`	**コマンドの使用方法** `Router# show policy-map interface`
キューイング戦略の表示	
コマンド • show queueing **コマンドの使用方法** `Router# show queueing`	**コマンドの使用方法** `Router# show policy-map interface`
重み付けランダム早期検出（WRED）情報の表示	
コマンド • show interfaces random-detect **コマンドの使用方法** `Router# show interfaces [type number] random-detect`	**コマンドの使用方法** `Router# show policy-map interface`
WRED パラメータ グループの表示	
コマンド • show random-detect-group **コマンドの使用方法** `Router# show random-detect-group`	**コマンドの使用方法** `Router# show policy-map interface`
トラフィック シェーピングの設定、キューイング、および統計情報の表示	

隠しコマンド、削除済みのコマンド、またはサポート対象外のコマンド	代替 MQC コマンド シーケンス

コマンド
• show traffic-shape
• show traffic-shape queue
• show traffic-shape statistics
コマンドの使用方法
`Router# show traffic-shape [interface-type interface-number]`
`Router# show traffic-shape queue [interface-number [dlci dlci-number]]`
`Router# show traffic-shape statistics [interface-type interface-number]`
	コマンドの使用方法
	`Router# show policy-map interface`

重み付け均等化キューイング情報の表示	
コマンド • show interfaces fair-queue コマンドの使用方法 `Router# show interfaces [interface-type interface-number] fair-queue`	コマンドの使用方法 `Router# show policy-map interface`

その他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『Cisco IOS Master Commands List, All Releases』
トラフィック クラスの定義、インターフェイスへのトラフィック ポリシーの対応付け	『*QualityofServiceSolutionsConfigurationGuide*』の「Applying QoS Features Using the MQC」モジュール
QoS コマンドに関する参照ページ	『*Cisco IOS Quality of Service Solutions Command Reference*』
ワイドエリア ネットワーク コマンドに関する参照ページ	『*Cisco IOS Wide-Area Networking Command Reference*』

シスコのテクニカル サポート

説明	リンク
★枠で囲まれた Technical Assistance の場合★右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

レガシー QoS コマンドの廃止予定に関する機能情報

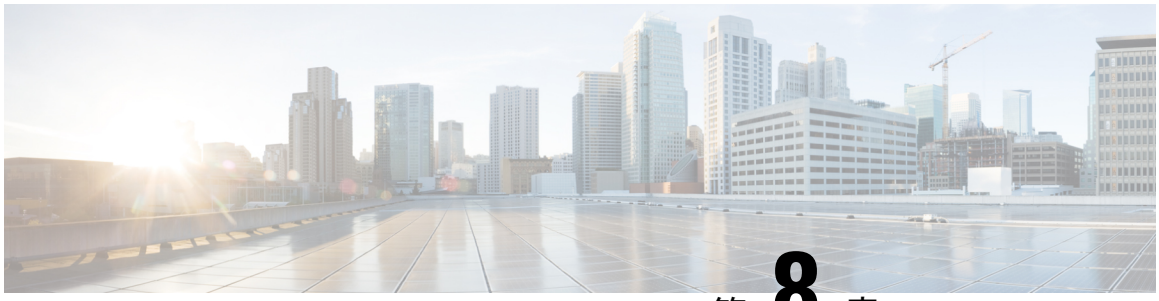
次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *6 :* レガシー *QoS* コマンドの廃止予定に関する機能情報

機能名	リリース	機能情報
レガシー QoS コマンドの廃止予定：隠しコマンド	15.0(1)S 15.1(3)T	Cisco IOS QoS を効率化するため、特定のコマンドを隠しコマンドにしました。つまり、コマンド ラインに疑問符（**?**）を入力して隠しコマンドを表示しようとしても、そのコマンドは表示されません。ただし、コマンド シンタックスをわかっていれば、入力できます。これらのコマンドは将来のリリースで削除される予定です。 これらの隠しコマンドによって提供される機能は、モジュラ QoS CLI（MQC）からの、QoS を設定するための、プラットフォームに依存しない一連のコマンドである同様の機能に置き換えられています。 次のコマンドが変更されました：**custom-queue-list**、**fair-queue (WFQ)**、**frame-relay adaptive-shaping**（**becn**キーワード）、**frame-relay adaptive-shaping**（**foresight**キーワード）、**frame-relay bc**、**frame-relay be**、**frame-relay cir**、**frame-relay congestion threshold de**、**frame-relay congestion threshold ecn**、**frame-relay custom-queue-list**、**frame-relay fair-queue**、**frame-relay fecn-adapt**、**frame-relay ip rtp priority**、**frame-relay priority-group**、**frame-relay qos-autosense**、**ip rtp priority**, **max-reserved-bandwidth**、**priority-group**、**random-detect**、**random-detect dscp** 、**random-detect**（**dscp-based**キーワード）**random-detect**、**exponential-weighting-constant random-detect flow** 、**random-detect flow average-depth-factor** 、 **random-detect flow count random-detect random-detect precedence**、**random-detect-group**、**show interfaces fair-queue prec-based show interfaces random-detect**、**show queue**、**show queueing**、**show random-detect-group**、**show traffic-shape**、**show traffic-shape queue**、 **show traffic-shape statistics**。

機能名	リリース	機能情報
レガシー QoS コマンドの廃止予定：隠しコマンド	Cisco IOS XE リリース 2.6	Cisco IOS XE QoS を効率化するため、特定のコマンドを隠しコマンドにしました。つまり、コマンドラインに疑問符（**?**）を入力して隠しコマンドを表示しようとしても、そのコマンドは表示されません。ただし、コマンド シンタックスをわかっていれば、入力できます。これらのコマンドは将来のリリースで削除される予定です。 これらの隠しコマンドによって提供される機能は、モジュラ QoS CLI（MQC）からの、QoS を設定するための、プラットフォームに依存しない一連のコマンドである同様の機能に置き換えられています。 次のコマンドが変更されました：**custom-queue-list**、**fair-queue (WFQ)**、**frame-relay adaptive-shaping** （**becn** キーワード）、**frame-relay adaptive-shaping**（**foresight** キーワード）、**frame-relay bc**キーワード**frame-relay**)、**be**、**frame-relay cir**、 **frame-relay congestion threshold de**、**frame-relay congestion threshold ecn**、**frame-relay custom-queue-list**、**frame-relay fair-queue**、**frame-relay fecn-adapt**、**frame-relay ip rtp priority**, **frame-relay priority-group**、**frame-relay qos-autosense**、**ip rtp priority**、**max-reserved-bandwidth**、**show interfaces fair-queue**、**show interfaces random-detect**、 **show queue**、**show queueing**、**show traffic-shape**、 **show traffic-shape queue**、**show traffic-shape statistics**。
レガシー QoS コマンドの廃止予定：削除済みコマンド	Cisco IOS XE リリース 3.2S	レガシー QoS コマンドは削除されました。そのため、適切な代替 MQC コマンドを使用する必要があります。 次のコマンドが削除されました： **custom-queue-list**、**fair-queue (WFQ)**、**frame-relay adaptive-shaping**（**becn** キーワード)、**frame-relay adaptive-shaping**（**foresight** キーワード）、**frame-relay bc**、**frame-relay be**、**frame-relay cir**、**frame-relay congestion threshold de**、**frame-relay congestion threshold ecn**、**frame-relay custom-queue-list**、 **frame-relay fair-queue**、**frame-relay fecn-adapt**、**frame-relay ip rtp priority**、**frame-relay priority-group**、**frame-relay qos-autosense**、**ip rtp priority**、**max-reserved-bandwidth**、**show interfaces fair-queue**、**show interfaces random-detect**、**show queue**、**show queueing**、**show traffic-shape**、**show traffic-shape queue**、**show traffic-shape statistics**。



第 8 章

QoS パケット マーキング

QoS パケット マーキングとは、レイヤ 2（802.1Q/p CoS、MPLS EXP）またはレイヤ 3（IP Precedence、DSCP、IP ECN）のいずれかでのパケット内のフィールドの次の変更のことです。また、以前に到達した分類の決定を保存することも意味します。

概要

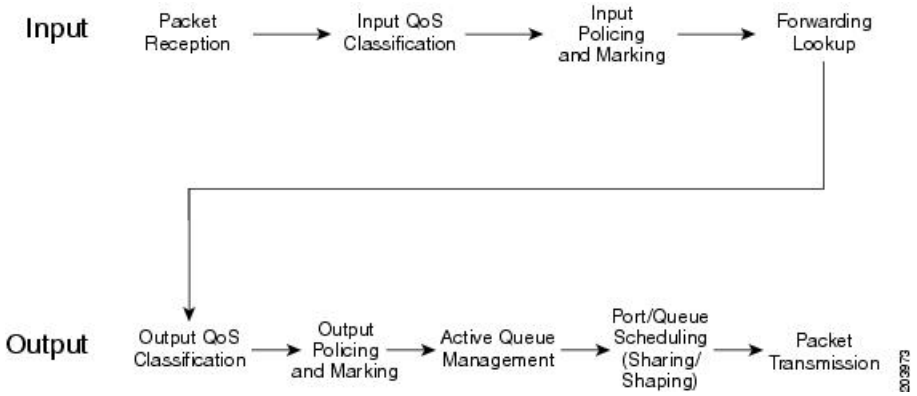
マーキングの定義

マーキングは航空券のファースト、ビジネス、エコノミーといった「サービス クラス」の定義と概念上は似ています。この値は、得られるサービスのレベル（品質）を反映します。同様に、パケットの値をマークして、ネットワークを通過するそのパケットのサービス クラス（以降、サービスクラスと呼びます）を示します。マークされた値を確認し、ネットワーク要素により、パケットの処理方法を指定することができます。

ビジネスクラスの乗客はその指定を得るために、さまざまな手段を使用したかもしれません。余分に料金を支払ったり、マイルを使用したり、または、幸運なことに他の座席が満席であったために通常の料金で予約できた可能性があります。どこかで誰かが特定のサービスクラスの利用資格を決定する複雑な分類タスクを実行し、その後でファーストクラス、ビジネスクラス、またはエコノミークラスの指定のみをチケットにマークしています。フライトアテンダントは、利用資格がどのように決定されたかについては関心がありません。チケットにマークされたクラスを見て、そのレベルのサービスを提供するだけです。

これがネットワーキングの世界で行われます。あるデバイスがフロー内のデータで複雑な分類を実行し、適切なサービスクラスを決定します。他のネットワーク要素は、受信するパケット内にマークされた値を「信頼」して、その指定に適したサービスを提供します。

図 *55 : QoS* パケット処理



QoS パケット処理というコンテキストでは、分類の後およびキューイングの前にマーキングが実行され、入力または出力に適用できます。

通常は、信頼境界を ネットワークのエッジで作成した後に、エッジ デバイスのパケットを分類してマークします。その後で、ネットワーク全体にわたってマークしたそのフィールドをホップ単位の処理についての分類と決定に使用することができます。

（注） 信頼境界を使用すると、パケットがネットワークに着信したときにそれらすべてのパケットにネットワーク制御のマーキングを適用し、適用しなかったデフォルト以外のマーキングを削除したり、変更したりすることができます。

たとえば、VoIP デバイスが接続されているルータ ポートをシステムが認識していることを想像してください。音声パケットの DiffServ コードポイント（DSCP）値を（ネットワークのエッジで）EF とマークし、ネットワークを通じて DSCP ベースの分類を使用して、低遅延処理を保証するパケットを決定します。

パケットをマーキングする理由

パケットをマーキングする理由は次のとおりです。

- ネットワークを通過する際にパケットをどのように処理するかを指定する。
- 複雑な分類を一度実行する。サービス クラスをマーキングすることによって、よりシンプルで、CPU への負荷が低い分類をネットワーク内の他の場所で使用できます。
- フローの可視性がさらに高いネットワークのポイントで分類を実行します。たとえば、データが暗号化されている場合は、そのフロー内で伝送されるアプリケーションを決定するなどの複雑な分類は実行できません。代わりに、暗号化前に分類して、パス上のネットワーク要素に表示される非暗号化ヘッダーの値をマークすることができます。

  パケットは異なる自律エンティティ（2 つの企業オフィス間のサービス プロバイダー ネットワークなど）が管理するネットワークを通過するため、それらのネットワーク上では

サービスレベルの指定に対するマーキングが整合しない場合は再度マーキングを行う必要があります。

パケットは異なるネットワーキングテクノロジーを通過するため、サービスクラスを示すために使用可能なフィールドが異なる場合があります。たとえば、IP パケットの DSCP フィールドでサービスクラスの指定を伝送していても、このパケットがマルチプロトコル ラベル スイッチング（MPLS）ネットワークを通過する場合は、ネットワーク要素がサービスクラスの判断に使用できるのはMPLS EXP フィールドのみの場合があります。ネットワークのその部分では、MPLS EXP ビットの適切なマーキングを判断する必要があります。

ネットワーク オペレータとして、ユーザから特定のレートでデータを受け取るように契約している場合があります。そのレートを超えたパケットをドロップするのではなく、低いサービスクラスとしてマークできます。

マーキング パケットに対するアプローチ

パケットのマーキングには、**set** コマンドおよびポリサー マーキング アクションの 2 つの主要なアプローチがあります。

（注） ここでは、「ポリシング」アクションについて簡単に説明します。

set コマンド

ルータ上でのマーキング パケットに対する最もシンプルなアプローチは、**set** コマンドをポリシーマップ定義に使用することです。（ポリシーマップで、定義した各クラスのトラフィックに対して QoS アクションを指定します）。

すべての RTP ポートを 1 つのトラフィック クラスに分類し、各パケットに AF41 とマークするように決定できます。ポリシーマップは、次のようになります。

```
policy-map mark-rtp
  class rtp-traffic
    set dscp af41
```

ポリサー マーキング アクション

ポリサーを使用して、トラフィック クラス内の定義したレートを超えるパケットをドロップすることができます。または、そのレートを超えるパケットをマークし、そのレート未満のパケットでなく、それらのパケットが異なるホップ単位の処理を受けられるようにすることができます。

たとえば、AF41 とマークされたビデオ トラフィックがルータに着信したとします。最大 2 Mbps までのユーザ トラフィックは 最上位の相対的優先転送動作と見なし、2 Mbps を超えるトラフィックを AF42 に降格することができます（契約外、不適合と見なす）。

ポリシーマップは次のように表示されます。

```
class-map video-traffic
  match dscp af41
!
policy-map enforce-contract
  class video-traffic
    police cir 2m conform-action transmit exceed-action set-dscp-transmit AF42
```

マーキング アクションの範囲

分類と同様に、マーキングはデータ パケット内のすべてのフィールドにはアクセスできません。たとえば、IP パケットがマルチプロトコル ラベル スイッチング（MPLS）でカプセル化されている場合は、IP ヘッダー内の DSCP をマークできません。MPLS から最初にカプセル化を解除する必要があります。ただし、MPLS EXP ビットはマークできます。

（注） マーキングに使用できるのは、レイヤ 2 のヘッダーと外部レイヤ 3 のヘッダーのみです。

複数の set ステートメント

複数のマーキング ルールを 1 つのクラス（またはポリサー アクション）内に設定できます。これにより、同じパケット内のレイヤ 2 とレイヤ 3 の両方のフィールドをマークすることができます。または、複数のトラフィック タイプが同じクラス内に存在する場合は、各タイプにマーキング値を定義します。

たとえば、イーサネット サブインターフェイスに適用された次の出力ポリシーがあるとします。

```
policy-map mark-rtp
  class rtp-traffic
    set cos 4
    set mpls exp topmost 4
    set dscp af41
```

MPLS パケットがこのサブインターフェイス経由で転送されると、レイヤ 2 の COS フィールドと MPLS ヘッダー内の EXP ビットがマークされます。IP データグラムがそのパケットにカプセル化されていた場合、その DSCP 値は変更されずにそのまま残ります。ただし、IP パケットがサブインターフェイス経由で転送された場合は、そのレイヤ 2 の COS 値とレイヤ 3 の DSCP 値がマークされます。

コマンドの詳細については、set cos （186 ページ）、set mpls experimental topmost （191 ページ）、および set dscp （188 ページ） コマンドのページを参照してください。

内部指定子のマーキング

シスコ ルータでは、2 つの内部値（qos-group と discard-class）をマークできます。これらの値はルータ内をパケットとともに移動しますが、パケットのコンテンツは変更しません。

通常、入力ポリシー内のこれらの指定子をマークし、それらを使用して、出力ポリシーのトラフィック クラスや WRED ドロップ プロファイルを分類します。たとえば、ユーザの IP アドレスに基づいて出力を分類したくても、暗号化が設定されているために出力インターフェイスでユーザの IP アドレスがわからない場合があります。それらのトラフィックを（暗号化前に）入力で分類して、適切な qos-group 値を設定することができます。これで、qos-group に基づいて出力で分類できるようになり、それに応じたアクションを選択できるようになりました。

入力マーキング アクションと出力マーキング アクション

特定のマーキング値は、入力ポリシーのみか、または出力ポリシーのみに関連しています。たとえば、入力ポリシーで ATM CLP ビットまたはフレーム リレー DE ビットをマークしても パケットのカプセル化解除時に破棄されるため、無意味です。同様に、出力ポリシーで qos-group または discard-class をマーキングしても、これらはパケットを変更せずにそのままにされ、次のホップへの転送時にパケットをエンキューした時点で廃棄されるため、効果がありません。

インポジション マーキング

特殊な状況下では、パケットにまだ追加されていないヘッダー フィールドをマークすることができます（シスコでは、この動作をインポジション マーキングと呼びます）。

最も一般的なインポジション マーキングの例は **set mpls experimental imposition** コマンドの適用です。 これは、IP データグラムを含み、マルチプロトコル ラベル スイッチング（MPLS）ヘッダーがないパケットが到着する入力インターフェイスで使用できます。ルータが MPLS ヘッダーを使用してデータグラムをカプセル化する場合、このコマンドによって指定されたとおりに、EXP ビットがマークされます。

set dscp tunnel コマンドと **set precedence tunnel** コマンド（IPv4 の場合のみ）の適用は、もう 1 つのインポジション マーキングの例です。出力ポリシーをトンネル インターフェイスに適用した場合、そのポリシーが実行されるときにはトンネル ヘッダーは存在しません。つまり、どのようなマーキングでも元の（最終的には内側の）IP ヘッダーに適用されます。どちらのコマンドを使用しても、トンネル（外側の）IP ヘッダーにはマークし、元のヘッダーは変更せずにそのまま残すことができます。

次の表に、これらのコマンドをサポートしているトンネルのタイプとさまざまなカプセル化を示します。

表 *7:* サポートされる *DSCP* とプレシデンス トンネル マーキングの設定

名前	外部ヘッダー（カプセル化）	内部ヘッダー（ペイロード）	注
GRE (4 over 4)	IPv4/GRE	IPv4	サポートあり
GRE (6 over 4)	IPv4/GRE	IPv6	カプセル化はサポート対象外
GREv6 (4 over 6)	IPv6/GRE	IPv4	カプセル化はサポート対象外

GREv6 (6 over 6)	IPv6/GRE	IPv6	カプセル化はサポート対象外
IP-IP	IPv4	IPv4	サポートあり
IPv6-IP	IPv4	IPv6	サポートあり
IPv6 (4 over 6)	IPv6	IPv4	カプセル化はサポート対象外
IPv6 (6 over 6)	IPv6	IPv6	サポート対象外
PSEC (4 over 4)	IPv4/IPSEC	IPv4	サポート対象外
PSEC (6 over 4)	IPv4/IPSEC	IPv6	サポート対象外
IPSECv6 (4 over 6)	IPv6/IPSEC	IPv4	カプセル化はサポート対象外
IPSECv6 (6 over 6)	IPv6/IPSEC	IPv6	サポート対象外
mVPN（マルチキャスト VPN）	IPv4/GRE	IPv4	サポートあり
DMVPN（ダイナミック マルチポイント VPN）			サポートあり
mGRE（マルチポイント GRE）			サポートあり
MPLSoGREv4	IPv4/GRE	MPLS	サポート対象外
MPLSoGREv6	IPv6/GRE	MPLS	サポート対象外
L2TP	IPv4/L2TP	PPPoX	サポート対象外

新しいヘッダー（encapsulated）を追加すると、内部ヘッダー内の QoS マーキングが外部ヘッダーにコピーされます。たとえば、IP データグラムは MPLS ヘッダーでカプセル化される場合、デフォルトでは、新たにインポーズされたヘッダー内の MPLS EXP ビットに IP プレシデンス ビットがIP ヘッダーからコピーされます。

ヘッダーディスポジションに関しては、通常、外部のマーキングを内部ヘッダーにはコピーしません。たとえば、外部および内部の IP ヘッダーに異なる DSCP 値を持つパケットを GRE トンネルのエンドポイントで受信しているとします。外部ヘッダーを削除した場合は内部ヘッダーに DSCP 値をコピーしません。

インポジション マーキングの設定例については、例 4：トンネル インポジション マーキングの設定 （170 ページ）と例 5：トンネル インポジション マーキングを使用した SP ネットワークに対する再マーキング （184 ページ）を参照してください。

コマンドの詳細については、set mpls experimental imposition （190 ページ）、set dscp tunnel （188 ページ）、およびset precedence tunnel （192 ページ）を参照してください。

設定例

例 1：入力マーキングの設定

一部のトラフィックにサービスクラスを指定し、その他すべてのトラフィックをブリーチするためにマーキングを使用する場合、ネットワークのエッジに信頼境界を設定できます（この後の *** を参照）。信頼境界をネットワークに対してすべての入力で適用すると、ネットワーク内の各サービスクラスにマップするアプリケーションの制御を維持できます。

```
policy-map ingress-marking
  class voice
    set dscp ef
  class video
    set dscp af41
  class scavenger
    set dscp cs1
  class class-default                                    ***
    set dscp 0                                           ***
!
interface gigabitethernet1/0/0
  Service-policy in ingress-marking
```

詳しくは set dscp （188 ページ） ページを参照してください。

例 2：出力マーキングの設定

別の管理者がネットワーク パスの一部を制御しており、サービスクラス マッピングに別の DSCP を使用している場合、出力マーキングが必要です（たとえば、エンタープライズ内で、RFC4594 で説明されているとおりに 12 の異なるクラスのトラフィックを分類します）。ただし、サービス プロバイダーが提供しているのは 3 クラス モデルのみです。

また、レイヤ 2 の特定のクラス（イーサネット、フレームリレー、または ATM スイッチド ネットワークなど）に対する処理を示すために出力マーキングが必要です。

```
policy-map egress-marking
  class scavenger
    set atm-clp
```

コマンドの詳細については、set atm-clp （186 ページ） ページを参照してください。

例 3：MPLS EXP インポジションの設定

MPLS では、プロバイダー エッジ（PE）ルータが MPLS ヘッダーを使用してデータグラムやフレームをカプセル化します。コア内でのスイッチングの決定は MPLS ヘッダーに基づいており、カプセル化されたデータは把握していません。

MPLS ヘッダーで IPv4 データグラムがカプセル化されているレイヤ 3 MPLS ネットワークについて検討します。カスタマー エッジ（CE）側のインターフェイスでは、パケットの IPv4 ヘッ

ダーを把握しています。コア側のインターフェイスでは、MPLS ヘッダーを使用してデータグラムをカプセル化しているため、それらのヘッダー以外のことはわかりません。

デフォルトでは、MPLS EXP ビットに IP プレシデンスをコピーします。この動作を上書きすることにします。コア側のインターフェイス上では、IPv4 タイプのサービス バイトは解析できません。しかし、入力時に IP ヘッダーを解析して EXP 値を格納できます。 MPLS ヘッダーが追加されたときにこの値を設定するようにします。コマンド実行時には MPLS ヘッダーは存在しないため、ルータは命令を取得して、出力インターフェイスで EXP ビットをマークします。

```
policy-map mpls-exp-remark
  class voice
    set mpls experimental imposition 5
  class video
    set mpls experimental imposition 4
  class scavenger
    set mpls experimental imposition 0
!
interface gigabitethernet1/0/0
  policy-map input mpls-exp-remark
```

コマンドの詳細については、set mpls experimental imposition （190 ページ） ページを参照してください。

例 4：トンネル インポジション マーキングの設定

トンネルと MPLS EXP インポジション マーキングは概念的に似ています。パケットにまだ追加されていないヘッダー内の値をマークできます。また、GRE や IPinIP のようなレイヤ 3 トンネル テクノロジーでは、外部 IP ヘッダーを使用してレイヤ 3 データグラムをカプセル化できます。（インポジション マーキング （167 ページ）を参照）。

DMVPN ネットワークがあり、 そのネットワークではブランチ ロケーションでデータを暗号化し、そのデータを GRE ヘッダーでカプセル化してからパブリック IP ネットワークに送信しているとします。管理者は、ポリシー マップをトンネル インターフェイスに適用してそのトンネル内のアプリケーションの優先順位付けを行うことができます。また、外部 IP ヘッダーの DSCP をマークしてプロバイダーのネットワーク内でのサービスクラスを示す必要もあります。ポリシーが実行される時点では、外部ヘッダーはまだ追加されておらず、**set dscp** や **set precedence** のようなコマンドが内部 IP ヘッダーをマークします。

この問題を解決するには、 **set dscp tunnel** および **set precedence tunnel** コマンドを使用します。これらのコマンドでは、まだ追加されていない外部ヘッダーに値を設定できます。

次に、音声とビデオのトラフィックが分類されてエンタープライズ ネットワーク内の個別のキューに投入される例を示します。サービス プロバイダーには僅かなサービスクラスしかないため、音声とビデオの両方をプロバイダーのネットワーク内のプライオリティ クラスに投入することにしました。

外部トンネル ヘッダーの DSCP をマーキングすることにより、 内部ヘッダーの元のマーキングを維持することができます。

```
policy-map mark-outer-gre-header
  class voice
```

```
    priority level1 percent 20
    set dscp tunnel ef
  class video
    priority level 2 percent 20
    set dscp tunnel ef
!
interface tunnel100
  service-policy out mark-outer-gre-header
```

コマンドの詳細については、set dscp tunnel （188 ページ） ページを参照してください。

例 5：QoS グループ マーキングの設定

場合によっては、出力キューイングを入力分類に基づかせることができます。たとえば、MPLS 対応のインターフェイスに 9 つ以上の出力キューを必要としているとします。出力分類を使用して MPLS EXP ビットを制限するため、クラスは 8 つになります。解決策として、入力インターフェイスで分類を実行し、その分類に一致するパケットに QoS グループを設定します。QoS グループには、現在のルータ内以外は関連性がありません。つまり、パケット ヘッダー内にあるものを変更しません。代わりに、これは、ルータを通過する際にパケットに関連付けられた値です。

次の例では、Network-Based Application Recognition（NBAR）分類を使用します 入力で使用し、TelePresence ビデオと Jabber ビデオの両方を qos-group 4 でマークします。出力ポリシーでは、入力でマークした qos-group に基づいて分類します（「***」を参照）。

```
class-map telepresence-video
  match protocol telepresence-media
class-map jabber-video
  match protocol cisco-jabber-video
class-map egress-video-traffic                          ***
  match qos-group 4                                     ***
!
policy-map mark-qos-group
  class telepresence-video
    set qos-group 4
  class jabber-video
    set qos-group 4
!
policy-map egress-queuing
  class egress-video-traffic
    bandwidth remaining percent 50
!
interface gig 1/0/0
  service-policy in mark-qos-group
!
interface serial1/1/0
  service-policy out egress-queuing
```

コマンドの詳細については、set qos-group （192 ページ） ページを参照してください。

例 6：discard-class マーキングの設定

例 5：QoS グループ マーキングの設定 （171 ページ）では、TelePresence ビデオと Jabber ビデオの両方を qos-group 4 でマークし、これらの両方のアプリケーションを同じ出力キューに配置しています。

この出力キューで、重み付けランダム早期検出（WRED）を実行し、輻輳時には最初に Jabber ビデオをドロップするとします。通常、WRED はプレシデンス値または DSCP 値を確認してフローのドロップしきい値を決定します。ただし、例 3：MPLS EXP インポジションの設定 （169 ページ）に示されているように、IP ヘッダーは把握できません。これを解決するため、discard-class という 2 つめの内部値をマークします。その後、qos-group を使用して出力クラス（およびキュー）を選択し、discard-class を使用してそのクラス内の WRED ドロップ プロファイルを選択することができます。

```
class-map telepresence-video
  match protocol telepresence-media
class-map jabber-video
  match protocol cisco-jabber-video
class-map egress-video-traffic
  match qos-group 4
!
policy-map mark-qos-group
  class telepresence-video
    set qos-group 4
    set discard-class 1
  class jabber-video
    set qos-group 4
    set discard-class 2
!
policy-map egress-queuing
  class egress-video-traffic
    bandwidth remaining percent 50
    random-detect discard-class-based
    random-detect discard-class 1 24 40
    random-detect discard-class 2 22 30
!
interface gig 1/0/0
  service-policy in mark-qos-group
!
interface serial1/1/0
  service-policy out egress-queuing
```

コマンドの詳細については、set discard-class （187 ページ） ページを参照してください。

QoS パケット マーキングの確認

show policy-map interface コマンドは、IOS XE プラットフォーム上での QoS の動作を確認する主要手段です。パケット転送パス（データプレーン）は IOS インスタンス（コントロール プレーン）から分離されていますが、この一般的な IOS コマンドによって統計情報が報告されます。この機能はデフォルトでイネーブルになっています。

次の表に、以降の項で使用するフィールドを示します。

表 *8 : show policy-map interface* フィールドの説明（マーキングの確認に有効）

フィールド	説明
Service-policy input	指定されたインターフェイスまたはVCに適用されている入力サービス ポリシーの名前を示します。
Class-map	表示するトラフィックのクラスを指定します。ポリシーに設定されている各クラスに対して出力が表示されます。クラス一致の実装の選択（match-all または match-any）もトラフィック クラスの横に表示できます。
packets、bytes	表示するトラフィックの属していると識別されたパケットの数を指定します（バイト単位で表示）。
offered rate	クラスに着信するパケットのレートを 1 秒あたりのビット数で指定します。
Match	トラフィック クラスの一致基準を指定します。
QoS Set	特定のクラスに対して設定された QoS マーキング アクションの詳細。
Packets marked	イネーブルにした場合は、特定のクラスにマークされたパケットの合計数を表示します。 イネーブルにしなかった場合は、「Marker statistics: Disabled」が表示されます。

show policy-map interface コマンドでの確認

show policy-map interface コマンドは、 IOS XE プラットフォーム上での QoS の動作を確認する主要手段です。通常、特定のクラスに一致するパケット（デフォルトでイネーブルになる「クラス一致統計情報」）の数と、設定されているマーキングアクション（設定されている場合）を認識することで、そのアクションによってマークされたパケット数を十分に把握できます。

（注） 「クラス一致統計情報」（デフォルトでイネーブル）と「マーキング統計情報」（デフォルトではディセーブル）の違いを理解する必要があります。通常は、クラス一致統計情報を理解しておくだけで十分です。パケットがクラスに「ヒット」すると、マークされたと想定できます。ただし、複数の相互に排他的なマーキング値を設定し、**set** コマンドごとにマークされたパッケージの数を知る必要がある場合、あらゆる注意を払って「マーキング統計情報」をイネーブルにすることができます。

次に、物理インターフェイスに適用するポリシーがある入力マーキングの例を示します。この例では、jabber-video をポート 2000 ～ 3000 上に設定するとします。

```
class-map match-all jabber-video
  match ip rtp 2000 3000
```

```
!
policy-map mark-traffic
  class jabber-video
    set dscp af41

show policy-map int g1/0/0
GigabitEthernet1/0/0

  Service-policy input: mark-traffic

    Class-map: jabber-video (match-all)
      850 packets, 51000 bytes                                    note 1
      5 minute offered rate 2000 bps, drop rate 0000 bps
      Match: ip rtp 2000 3000
      QoS Set                                                     note 2
        dscp af41
          Marker statistics: Disabled

    Class-map: class-default (match-any)                          note 3
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

脚注：

note 1	クラス一致の統計情報
note 2	パケット一致セクション
note 3	クラスデフォルトの統計情報セクション

入力マーキングの出力の「`Marker statistics: Disabled`」に注意してください。複数の統計情報を呼び出す場合、以前の出力で提供された情報で十分でなかったときは、「パケット マーカー統計情報」をイネーブルにすることができます。

QoS パケット マーキング統計情報での確認

始める前に

両方

- すべてのポリシー マップを削除し、コマンドを発行し、すべてのポリシー マップを再適用します。
- コマンドを発行し、設定を保存し、ルータをリロードします。

（注） QoS のパケット マーキング統計情報をイネーブルにすると 拡張された設定では CPU 使用率が増加する可能性があります。統計情報を表示する利点とシステムの CPU 使用率の増加とを比較して検討する必要があります。

QoS パケット マーキング統計情報のイネーブル化

パケット マーキング統計情報をイネーブルにするには、 **platform qos marker-statistics** コマンドを発行します。コンフィギュレーション モードで発行してください。

QoS パケット マーキング統計情報の表示

指定したインターフェイス（またはサブインターフェイス）、あるいはインターフェイスの特定の相手先固定接続（PVC）のいずれかで、すべてのサービス ポリシーに対して設定されたすべてのクラスのパケット統計情報を表示するには、 **show policy-map interface** コマンドを使用します。

ポリシー マップに単独でマーキングを設定した場合、ASR 1000 シリーズ アグリゲーション サービス ルータからの出力は次のようになります。

```
policy-map remark-af41
  class af41-traffic
    set dscp tunnel ef
```

ここで、ユーザの IP ヘッダー内に af41 とマークされたトラフィックと GRE IP ヘッダーに EF とマークされた DSCP があるトンネル インターフェイスにこのマップを配置した場合、.**show policy-map interface** の出力は次のようになります。

```
show policy-map interface tunnel1

  Service-policy output: remark-af41

    Class-map: af41-traffic (match-all)
      978 packets, 68460 bytes                                  note 1
      5 minute offered rate 2000 bps, drop rate 0000 bps
      Match:  dscp af41 (34)
      QoS Set                                                   note 2
        dscp tunnel ef
          Marker statistics: Disabled                           note 3

    Class-map: class-default (match-any)
      365 packets, 25550 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

脚注：

note 1	クラス一致統計情報を表示します（「確認された」パケットはすべて AF41 とマークされると想定しています）。
note 2	マーキングは設定された唯一のアクションです。
note 3	デフォルトでは、セット単位のアクション統計情報はディセーブルになっています。

ここで、マーキング統計情報をイネーブルにした場合、 **show policy-map interface** コマンドの出力は次のようになります。

```
show policy-map interface tunnel1

  Service-policy output: remark-af41

    Class-map: af41-traffic (match-all)
      575 packets, 40250 bytes
      5 minute offered rate 1000 bps, drop rate 0000 bps
      Match:  dscp af41 (34)
      QoS Set
        dscp tunnel ef
          Packets marked 575                    note

    Class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

脚注

note	マーキング統計情報はイネーブルになっていますが、この例では情報が冗長です。

コマンドの詳細については、set dscp tunnel （188 ページ） ページを参照してください。

データ プレーン設定の検証

データプレーン設定に IOS コントロール プレーン設定が反映されているかを確認するには、**show platform hardware qfp active feature qos interface [input|output]** コマンドを使用します。このコマンドは、ポリシーマップをインターフェイスに適用する前に発行した場合にのみ有効です。したがって、次のいずれかを実行する必要があります。

- すべてのポリシー マップを削除し、コマンドを発行し、すべてのポリシー マップを再適用する。
- コマンドを発行し、設定を保存し、ルータをリロードする。

次の出力では、データプレーンにアクションを設定し、値を設定していることに注意してください。

```
show platform hardware qfp active feature qos interface g1/0/0 input

Interface: GigabitEthernet1/0/0, QFP interface: 12
  Direction: Input
    Hierarchy level: 0
    Policy name: mark-traffic
      Class name: jabber-video, Policy name: mark-traffic
        QOS Set:
            dscp 34                                                    note
      Class name: class-default, Policy name: mark-traffic
```

脚注

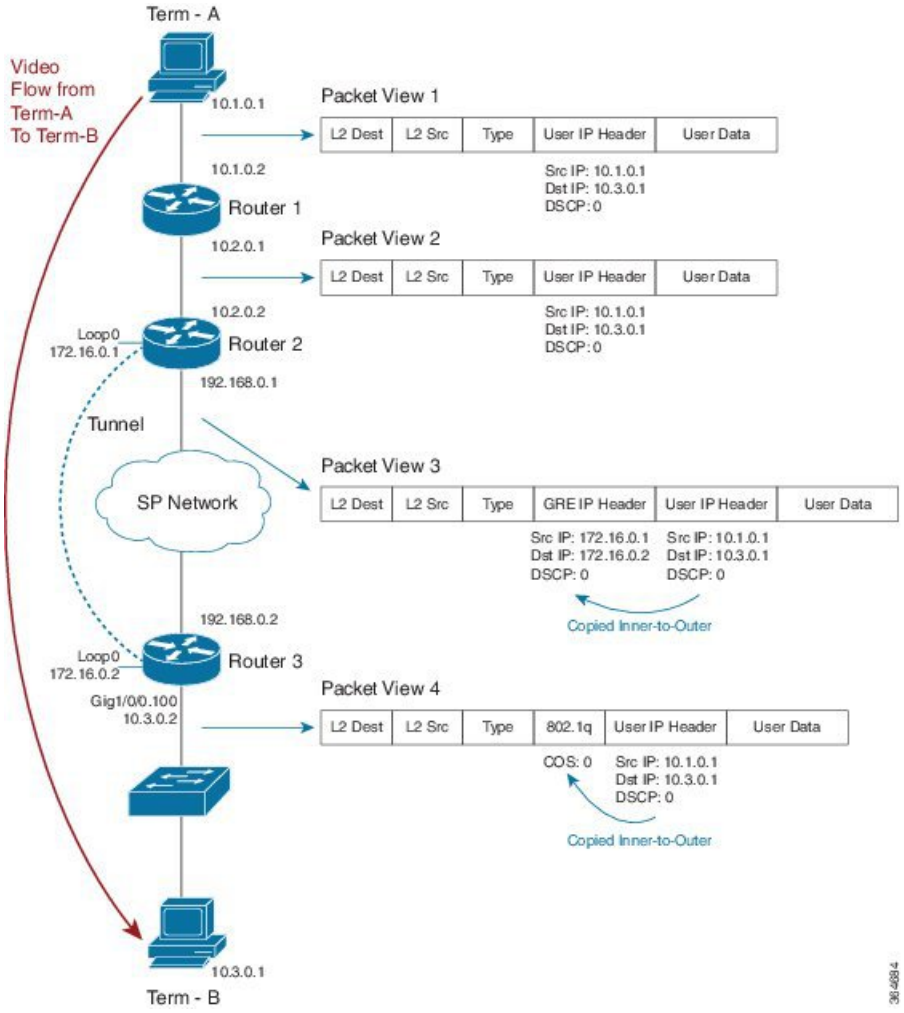
note	データプレーンはマーキングするようにプログラムされます。

ネットワーク レベルの設定例

次のシナリオでは、端末 A から端末 B にビデオ フローが移動します。

例 1：ネットワーク全体にわたるサービス クラス情報の伝達

図 *56 :* ネットワーク全体にわたるサービス クラス情報の伝達

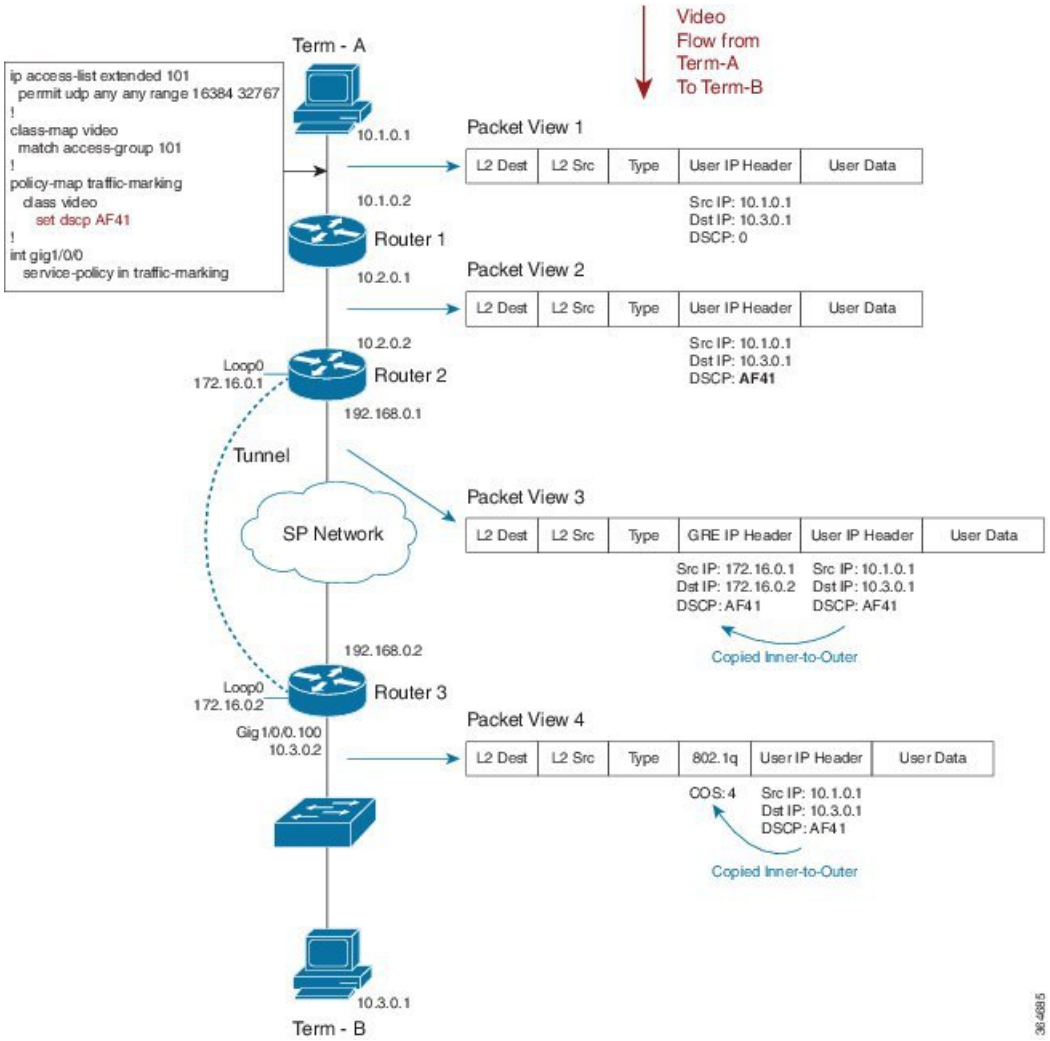


あるアプリケーションが DSCP コードポイント 0 を使用してビデオ ストリームをマークしていると想定します（パケット ビュー 1 を参照）。プロバイダーのネットワークを通過するために、GRE トンネルを通じて（可能な場合は暗号化して）ストリームを送信します。パケット ビュー 3 には、ユーザの IP データグラムを GRE パケット内にカプセル化していることが示されています。DSCP コードポイントがデフォルトでインポーズされた GRE ヘッダーにどのようにコピーされるかに注目してください。

最終の宛先の最後のホップで、ルータ 3 が VLAN のタグ付きパケットをスイッチに送信します（パケット ビュー 4 を参照）。VLAN の設定により、GRE ヘッダーが削除され、Dot1Q ヘッダーが追加されたことを確認してください。ユーザの DSCP 0（000 000）の先行部分が VLAN ヘッダーの COS ビットにデフォルトでコピーされます。CoS 値セットは 0（000）です。

例 2：ネットワークのエッジでのマーキングによるサービス クラスの指定

図 *57:* ネットワークのエッジでのマーキングによるサービス クラスの指定



この例では、ルータ 1 に入るときに入力ポリシーでユーザのトラフィックの DSCP を再度マーキングすることによって、デフォルト動作を変更します。次に、これを実行するコードを示します。

```
ip access-list extended 101
 permit udp any any range 16384 32767
!
class-map video
 match access-group 101
!
policy-map traffic-marking
  class video
    set dscp AF41
!
int gig1/0/0
  service-policy in traffic-marking
```

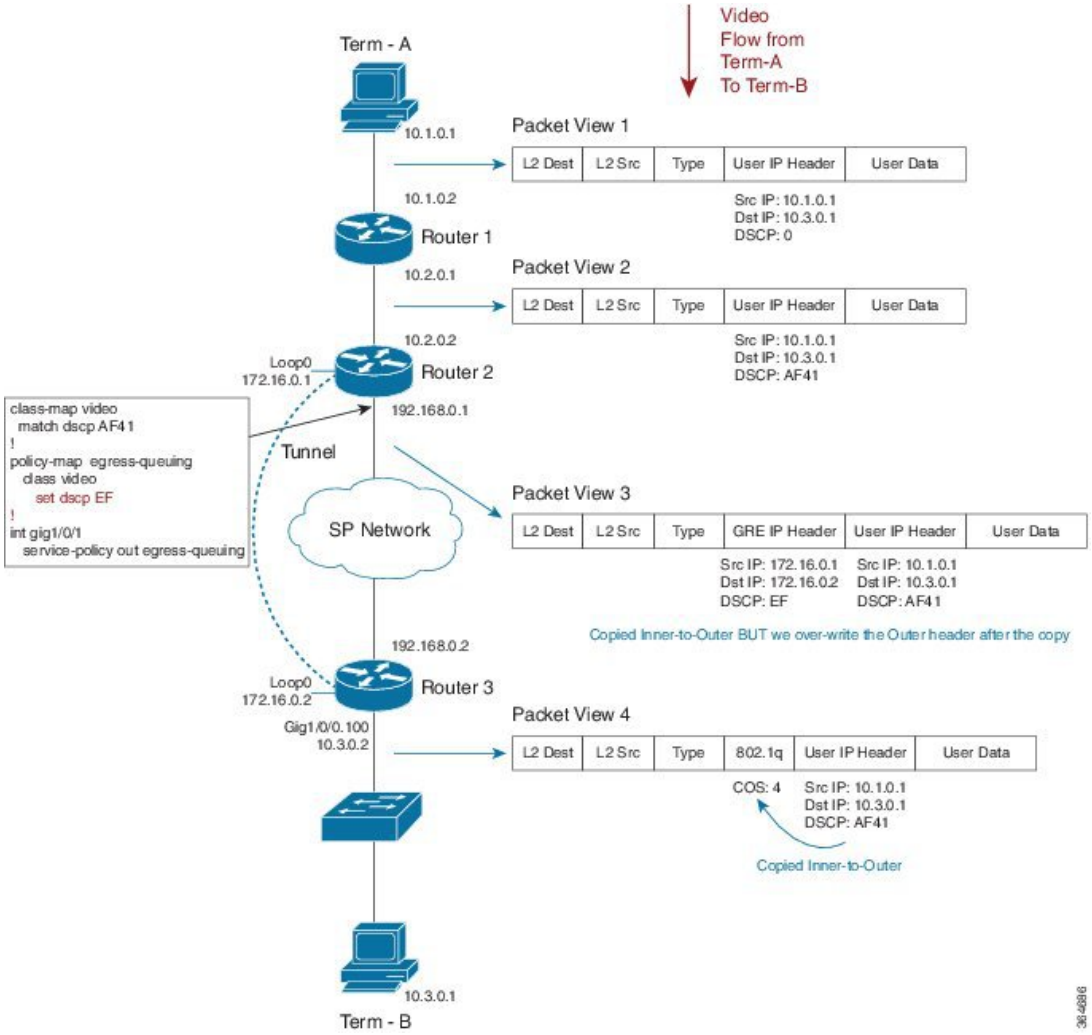
ビデオ トラフィックをネットワーク全体を通じて DSCP AF41 と指定するものとします。パケットが出力時に GRE インターフェイスに到達したとき、その DSCP 値はすでに AF41 に変更されており、その動作は例 1 と同じです。プロバイダー ネットワークを通過する際に、GRE トンネルを通じて（可能な場合は暗号化して）ストームを送信します。新しくマークされた DSCP コードポイント（AF41）がインポーズされた GRE ヘッダーにデフォルトでコピーされることに注目してください。

宛先に到着すると、ルータが VLAN のタグ付きパケットを最後のホップ（スイッチ）に送信します。ユーザの DSCP 値の先行部分が、VLAN ヘッダーの COS ビットにデフォルトでコピーされます。DSCP が FA41（100 010）になっているため、COS 値は 4（100）になります。

コマンドの詳細については、set dscp （188 ページ） コマンドのページを参照してください。

例 3：サービス プロバイダーの要件に一致させるためのトラフィックの再マーキング

図 *58 :* サービス プロバイダーの要件に一致させるためのトラフィックの再マーキング



ここでは、ネットワーク内で DSCP 値をマークしますが、サービス プロバイダーは別のマーキングを想定している例を示します。次に、これに対処するコードを示します。

```
class-map video
  match dscp AF41
!
policy-map  egress-queuing
  class video
    set dscp EF
!
int gig1/0/1
  service-policy out egress-queuing
```

ネットワーク内のビデオに対して DSCP を AF41 とマークします。一方、サービス プロバイダーはビデオ パケットが EF とマークされると想定しています。ルータ 2 の出力 Gig インターフェイスで、キューイング コマンドを含んだポリシーを追加します（この例では、設定のマーキング部分のみに重点を置いていることを思い出してください）。

パケットが出力物理インターフェイスに到達した時点で、GRE ヘッダーがすでにインポーズされているため、内部のカプセル化されたデータグラムから DSCP 値の AF 41 をコピーします。物理インターフェイスのポリシーは、外部 GRE ヘッダー内の DSCP 値のみを変更します。

（注） 内部ユーザ データグラムの IP ヘッダーがどのように変更されないままかに注目してください。

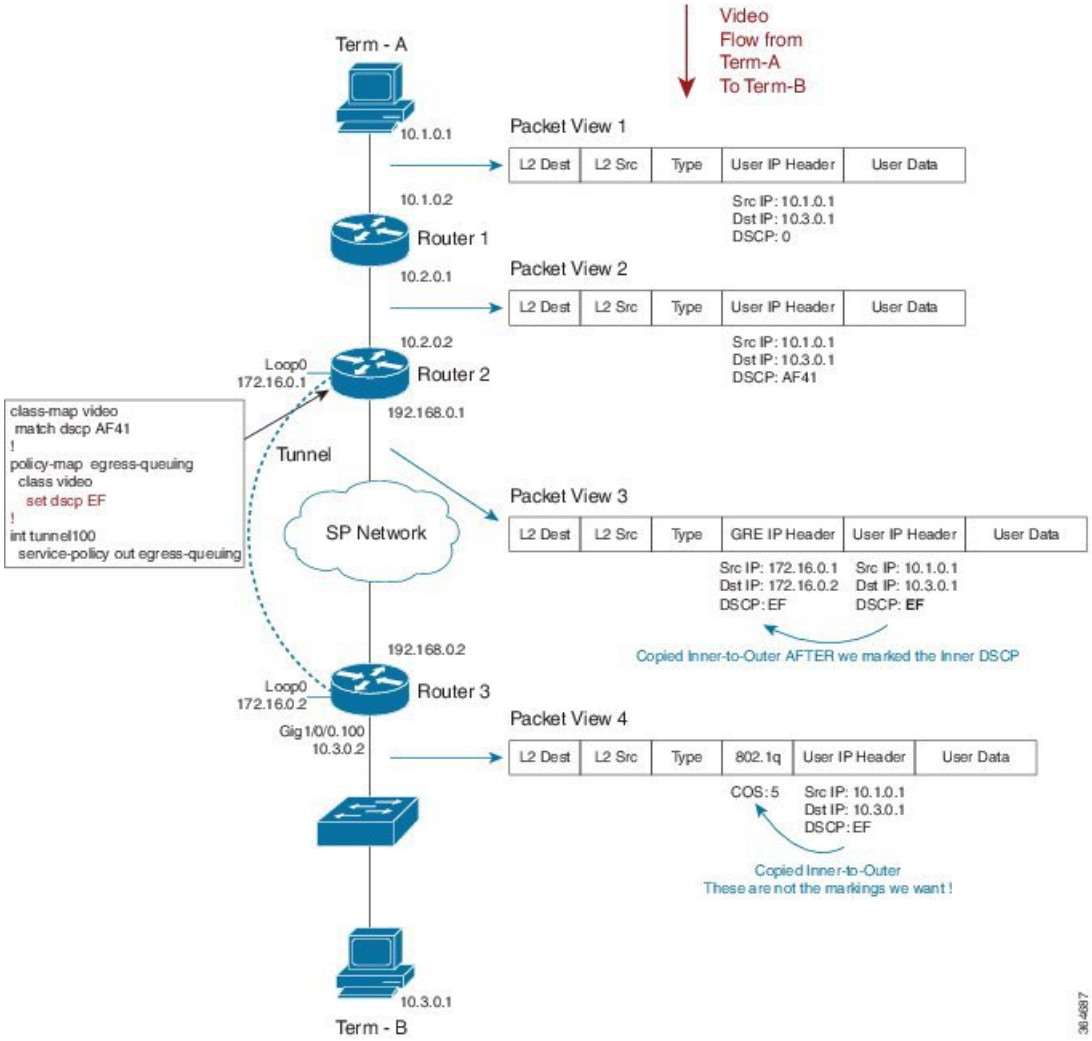
ルータ 3 に到達し、トンネルを終了すると、トンネル GRE ヘッダーが削除されます。その後、ユーザ データグラム の IP ヘッダーのみが表示されますが、ネットワークへの入力時にマークした値である AF41 が維持されています。

前の例のように、ルータが VLAN のタグ付きパケットを最後のホップ（スイッチ）に送信します。デフォルトでは、ユーザ IP ヘッダーの DSCP 値の先行部分が VLAN ヘッダー（802.1q）の CoS ビットにコピーされます。この時点で、DSCP 値は af41（100 010）であるため、COS 値は 4（100）になります。

コマンドの詳細については、set dscp （188 ページ） ページを参照してください。

例 4：SP ネットワークに対するトンネル インターフェイスでの再マーキング - Gotcha の可能性

図 *59 : SP* ネットワークに対するトンネル インターフェイスでの再マーキング - *Gotcha* の可能性



この例では、物理インターフェイスではなく、ルータ 1 のトンネル インターフェイスに QoS ポリシーを配置します（トンネルごとにキューイングを設定することで、物理インターフェイス上の集約ポリシーよりも多くの利点が得られます）。次に、これを実行するコードを示します。

```
class-map video
  match dscp AF41
!
policy-map  egress-queuing
  class video
    set dscp EF
!
int tunnel100
```

```
service-policy out egress-queuing
```

ポリシーのマーキング部分にのみ重きを置いています。トンネル インターフェイスでのマーキングは、トンネル ヘッダーが追加される前に実行されることが重要なポイントです。

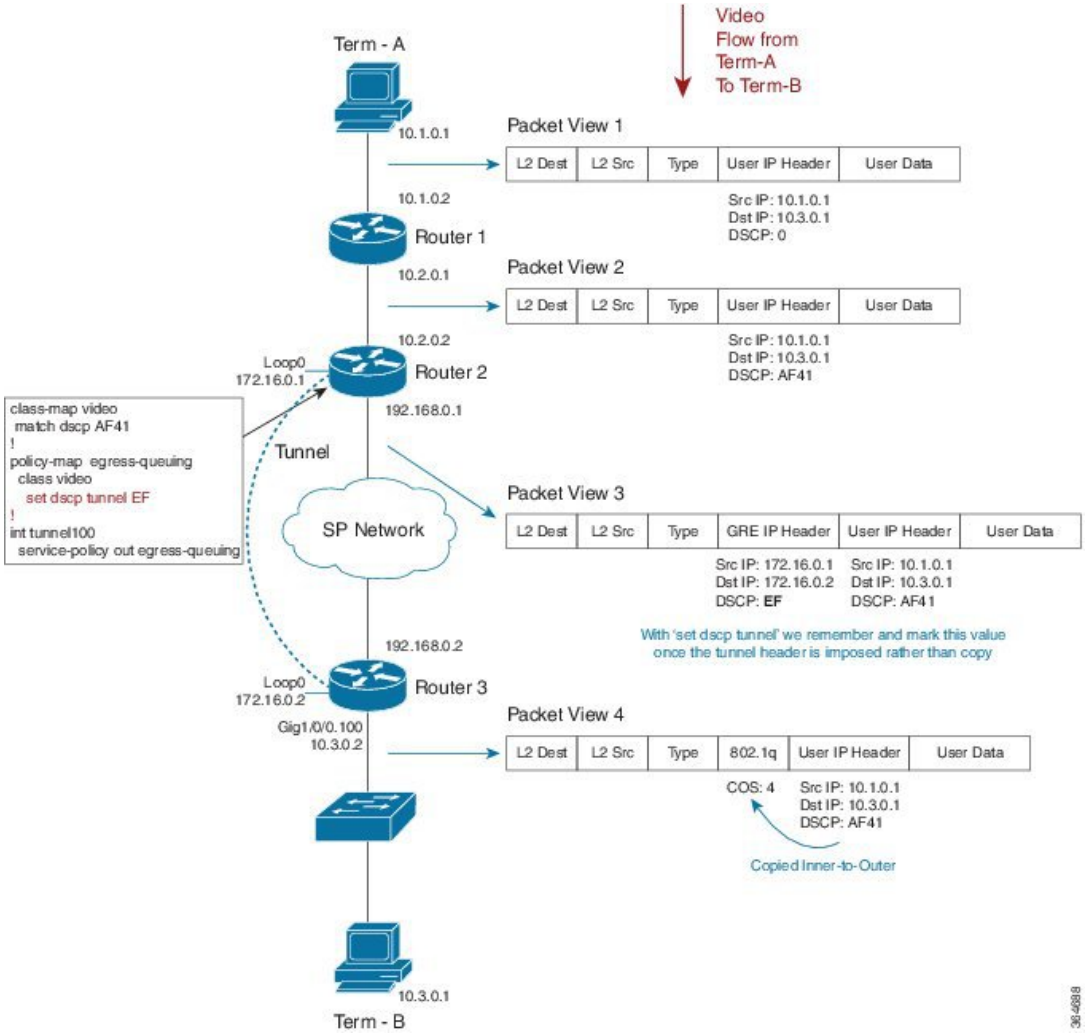
ユーザ データグラムの IP ヘッダーでどのようにポリシーが DSCP を上書きするかに注目してください。これは、GRE カプセル化の前に行われるため、新たにマークされた値が外部ヘッダーにコピーされます。

ルータ 3 に到達し、トンネルを終了すると、トンネル GRE ヘッダーが削除されます。ユーザ データグラム ヘッダーをマークしたため、新しい値が残りのネットワークを通じて伝達されます。これは、意図した動作ではありません。

コマンドの詳細については、set dscp （188 ページ） のページを参照してください。

例 5：トンネル インポジション マーキングを使用した SP ネットワークに対する再マーキング

図 *60 :* トンネル インポジション マーキングを使用した *SP* ネットワークに対する再マーキング



この例では、**set dscp tunnel** *dscp-value* コマンドを使用して、トンネル IP ヘッダーのみを変更します。

```
class-map video
  match dscp AF41
!
policy-map  egress-queuing
  class video
    set dscp tunnel EF
!
int tunnel100
  service-policy out egress-queuing
```

QoS ポリシーはルータ 2 のトンネル インターフェイスにあり、**set dscp** コマンドではなく **set dscp tunnel** コマンドを使用しました。

GRE ヘッダーはまだインポーズされていません。**set dscp tunnel** コマンドによって、DSCP 値が記憶され、カプセル化時に「内部から外部へ」コピーする代わりにこの値が使用されます。ユーザ IP データグラム ヘッダーの DSCP 値が変更されないことに注目してください。**set dscp tunnel** コマンドはトンネル IP ヘッダーのみを変更します。

コマンドの詳細については、set dscp tunnel （188 ページ） のページを参照してください。

コマンド リファレンス

platform qos marker-statistics

ルータに設定されたすべてのポリシーの各マーキング アクションに対して個別の統計情報の収集をイネーブルにするには、**platform qos marker-statistics** コマンドを グローバル コンフィギュレーション モードで使用します。パケット マーキング統計情報をディセーブルにするには、このコマンドの **no** 形式を使用します。

[no] platform qos marker-statistics

構文の説明

このコマンドには引数またはキーワードはありません。

コマンド デフォルト

無効になっています（パケット マーキング統計情報は表示されません）。ネットワーク オペレータは、クラス一致統計情報に依存します。

コマンド モード

policy-map（config-pmap）

使用上のガイドライン

このコマンドは、インターフェイスに対してポリシーマップが適用される前に発行された場合にのみ実行されます。したがって、次のいずれかを実行する必要があります。

- すべてのポリシー マップを削除し、コマンドを発行し、すべてのポリシー マップを再適用する。
- コマンドを発行し、設定を保存し、ルータをリロードする。

（注） パケット マーキング統計情報をイネーブルにすると、拡張設定では CPU 使用率が増加する可能性があります。そのため、統計情報の利点とシステムの CPU 使用率の増加とを比較して検討する必要があります。

set atm-clp

ATM セル損失率優先度（CLP）ビットを設定するには、**set atm-clp** コマンドを使用します ポリシー マップ クラス コンフィギュレーション モードで使用します。この設定をディセーブルにするには、このコマンドの **no** 形式を使用します。

[no] set atm-clp

構文の説明

このコマンドには引数またはキーワードはありません。

コマンド デフォルト

ATM CLP ビットは設定されません。

コマンド モード

policy-map（config-pmap）

使用上のガイドライン

ATM インターフェイスでは、アウトバウンド ポリシー内に **set atm-clp** コマンドを使用して、ATM セル ヘッダーを 1 に設定することができます。

このコマンドは、ATM、PPPoA、PPPoEoA、および L2TPv3 のカプセル化に対してサポートされています。ポリシーが VC に直接ではなく、トンネルに適用されている場合は、サポートされません。

セル損失優先度（CLP）ビット QoS が設定された ATM を含むポリシー マップは PPP over X（PPPoX）セッションに適用できません。マップは、**set atm-clp** コマンドを指定していない場合にのみ、受け入れられます。

set atm-clp コマンドを使用して出力マーキングを設定する例については、例 2：出力マーキングの設定 （169 ページ）を参照してください。

set cos

発信パケットのレイヤ 2 サービス クラス（CoS）値を設定するには、**set cos** コマンドを使用します。 ポリシー マップ クラス コンフィギュレーション モードで使用します。この設定をディセーブルにするには、このコマンドの **no** 形式を使用します。

[no] set cos *cos-value*

構文の説明

cos-value	発信パケットの IEEE 802.1Q CoS 値を 0 ～ 7 の範囲で指定します。

コマンド デフォルト

IP プレシデンス ビットまたは MPLS EXP ビットのいずれかがカプセル化されたデータグラムからコピーされます。

コマンド モード

policy-map（config-pmap）

使用上のガイドライン

set cos コマンドを使用して、サービスクラス情報を レイヤ 2 スイッチド ネットワークに伝達します。レイヤ 2 スイッチは組み込まれたレイヤ 3 情報（DSCP など）を解析できない場合がありますが、CoS 値に基づいて差別化サービスを提供することがあります。スイッチは、CoS 値のマークを含めて、レイヤ 2 ヘッダー情報を利用できます。

従来、**set cos** コマンドは、受信したフレームからレイヤ 2 情報をルータが廃棄するため、インターフェイスの出力方向に適用されたサービス ポリシー内でのみ意味を持ちます。EoMPLS や EVC のような機能の導入により、入力での CoS の設定では、ルーティングされたネットワーク全体を通じてレイヤ 2 情報を維持できます。

set cos-inner

QinQ パケットの内部 VLAN タグ内にレイヤ 2 CoS 値を設定するには、**set cos-inner** コマンドを使用します。 ポリシー マップ クラス コンフィギュレーション モードで使用します。この設定をディセーブルにするには、このコマンドの **no** 形式を使用します。

[no] set cos-inner *cos-value*

構文の説明

cos-value	IEEE 802.1q CoS 値を 0 ～ 7 の範囲で指定します。

コマンド デフォルト

IP プレシデンス ビットまたは MPLS EXP ビットのいずれかがカプセル化されたデータグラムからコピーされます。

コマンド モード

policy-map（config-pmap）

使用上のガイドライン

従来、ルータは受信したフレームからレイヤ 2 情報を破棄するため、**set cos-inner** コマンドはインターフェイスの出力方向に適用されたサービス ポリシー内でのみ意味がありました。EoMPLS や EVC のような機能の導入により、入力での CoS の設定は、ルーティングされたネットワーク全体を通じてレイヤ 2 情報を維持するため重要です。

set discard-class

パケットに対して QoS 廃棄クラスを設定するには、**set discard-class** コマンドをポリシー マップ コンフィギュレーション モードで使用します。この設定をディセーブルにするには、このコマンドの **no** 形式を使用します。

[no] set discard-class *discard-class-value*

構文の説明

discard-class-value	廃棄クラス値を 0 ～ 7 の範囲で指定します。

コマンド デフォルト

パケットに関連付けられた廃棄クラス値は 0 に設定されます。

コマンド モード

policy-map（config-pmap）

使用上のガイドライン

set discard-class コマンドでは、ルータによる処理中に廃棄クラス値をパケットに関連付けることができます。この値を設定すると、パケットは変更されません。

廃棄クラスと廃棄クラス ベースの WRED を出力ポリシー内に使用して、輻輳時にドロップするパケットを制御できます。

set dscp

IP ヘッダーに DSCP 値を設定するには、**set dscp** コマンドをポリシー マップ クラス コンフィギュレーション モードで使用します。この設定をディセーブルにするには、このコマンドの **no** 形式を使用します。

[no] set dscp *dscp-value*

構文の説明

dscp-value	IP ヘッダー内の DSCP 値を 0 ～ 63 の範囲で設定します。この値は数字か、または既知の DiffServe 名（つまり、EF）を使用して指定できます。

コマンド デフォルト 受信したパケットの既存の DSCP 値を保持します。

コマンド モード policy-map（config-pmap）

使用上のガイドライン このコマンドは、入力ポリシーまたは出力ポリシーに使用されることがあります。

パケットがネットワークを通過する際にパケットに対する QoS の処理を示すために DSCP 値を使用することができます。

（注） DSCP を使用した差別化サービス アーキテクチャはプレシデンスの使用よりも優先されます。

このコマンドは、一番外側のレイヤ 3 ヘッダーが IPv4 か IPv6 のいずれかの場合にパケットをマークします。

出力ポリシーマップ内で発行した場合、このコマンドはクラスやキューの選択は変更しませんが、WRED ドロップ プロファイルの選択に影響することがあります。

set dscp コマンドと **set ip dscp** コマンドは同じように動作し、IPv4 パケットと IPv6 パケットの両方をマークします。

（注） **match ip dscp** コマンドは IPv4 パケットのみを分類するのに対し、**match dscp** コマンドは IPv4 パケットと IPv6 パケットの両方を分類する分類プロセスとは異なります。

set dscp tunnel

パケットにまだ追加されていないトンネル ヘッダー内に DSCP 値を設定するには、**set dscp tunnel** コマンドをポリシー マップ クラス コンフィギュレーション モードで使用します。この設定をディセーブルにするには、このコマンドの **no** 形式を使用します。

[no] set dscp tunnel *dscp-value*

構文の説明

dscp-value	トンネル ヘッダー内の DSCP 値を 0 ～ 63 の範囲で指定します。この値は数字か、または既知の DiffServe 名（つまり、EF）のいずれかを使用して指定できます。

コマンド デフォルト

カプセル化されたデータグラムからの DSCP 値が新たにインポーズされたトンネル ヘッダーにコピーされます。

コマンド モード

policy-map（config-pmap）

使用上のガイドライン

このコマンドは、トンネル ヘッダーの追加前でないと意味がありません。

（注） トンネル インターフェイスに適用された入力ポリシーまたは出力ポリシーのいずれかにこのコマンドを使用できます。ただし、出力ポリシーが適用されている場合、ポリシーの評価時にすべてのヘッダーが追加されるため、このコマンドは意味がありません。

Cisco ASR シリーズ アグリゲーション サービス ルータでは、**set dscp tunnel** コマンドは IPv4 に対してのみサポートされています。サポートされている DSCP トンネルのマーキング設定をリストした表については、インポジション マーキング （167 ページ）を参照してください。

このコマンドを使用し、レイヤ 3 データグラムを外部 IP ヘッダーでカプセル化する例については、例 4：トンネル インポジション マーキングの設定 （170 ページ）を参照してください。

set fr-de

frame-relay（FR）discard eligible（DE）ビットを設定するには、**set fr-de** コマンドをポリシー マップ クラス コンフィギュレーション モードで使用します。設定をディセーブルにするには、コマンドの **no** 形式を入力します。

[no] set fr-de

構文の説明

このコマンドには引数またはキーワードはありません。

コマンド デフォルト

データグラムがフレーム リレーでカプセル化されていると、DE ビットは設定されません。

使用上のガイドライン

フレーム リレー カプセル化を使用して設定されたシリアル インターフェイスでは、**set fr-de** コマンドをアウトバウンド ポリシーに使用して、フレーム リレー ヘッダーの廃棄適性ビットを 1 に設定することができます。

set ip dscp

下位互換性を維持するために、同等の機能を実行する 2 つのコマンド バリアント、**set ip dscp** と **set dscp** をサポートしています。いずれかを使用して、IP ヘッダー内の DSCP 値をマークできます。詳細については、**set dscp** コマンド ページ（set dscp （188 ページ））を参照してください。

set ip dscp tunnel

下位互換性を維持するために、同等の機能を実行する 2 つのコマンド バリアント、**set ip dscp tunnel** と **set dscp tunnel** をサポートしています。詳細については、**set dscp tunnel** コマンド ページ（set dscp tunnel （188 ページ））を参照してください。

set ip precedence

下位互換性を維持するために、同等の機能を実行する 2 つのコマンド バリアント、**set ip precedence** と **set precedence** をサポートしています。いずれかを使用して、IP ヘッダー内のプレシデンス値をマークできます。詳細については、**set precedence** コマンド ページ（set precedence （191 ページ））を参照してください。

set ip precedence tunnel

下位互換性を維持するために、同等の機能を実行する 2 つのコマンド バリアント、**set ip precedence tunnel** と **set precedence tunnel** をサポートしています。詳細については、**set precedence tunnel** コマンド ページ（set precedence tunnel （192 ページ））を参照してください。

set mpls experimental imposition

インポーズされたすべてのラベル エントリの MPLS EXP フィールドの値を設定するには、**set mpls experimental imposition** コマンドを ポリシー マップ クラス コンフィギュレーション モードで使用します。この設定をディセーブルにするには、このコマンドの **no** 形式を使用します。

[no] set mpls experimental imposition *mpls-exp-value*

構文の説明

mpls-exp-value	MPLS EXP 値を 0 ～ 7 の範囲で指定します。

コマンド デフォルト

MPLS 値は、カプセル化されたパケット内の該当するフィールド（通常は precedence）からコピーされます。

コマンド モード

policy-map（config-pmap）

使用上のガイドライン

set mpls experimental imposition コマンドは、入力インターフェイス上でのみサポートされます。ラベル インポジション時にこのコマンドを使用し、インポーズされたすべてのラベル エントリの MPLS EXP フィールドを設定します。

このコマンドを使用して、データグラムまたはフレームをカプセル化するために使用する MPLS ヘッダー内の EXP ビットを設定する例については、例 3：MPLS EXP インポジションの設定（169 ページ）を参照してください。

set mpls experimental topmost

最上位ラベルの MPLS EXP フィールド値を設定するには、**set mpls experimental topmost** コマンドを使用しますポリシー マップ クラス コンフィギュレーション モードで使用します。この設定をディセーブルにするには、このコマンドの **no** 形式を使用します。

[no]set mpls experimental topmost *mpls-exp-value*

構文の説明

mpls-exp-value	MPLS EXP 値を 0 ～ 7 の範囲で指定します。

コマンド デフォルト

MPLS EXP 値は、カプセル化時に最も内側のヘッダーからコピーされるか、または変更されないままになります。

コマンド モード

policy-map（config-pmap）

使用上のガイドライン

このコマンドは、コマンドの評価時に最も外側のレイヤ 3 ヘッダーが MPLS ラベルである場合にパケットをマークします。

このコマンドは、最上位ラベル内の MPLS EXP 値のみを設定します。スタック内に複数のラベルが存在する場合、最上位以外のラベル内の MPLS EXP は変更されないままになります。

set precedence

パケット ヘッダー内の IP プレシデンス値を設定するには、**set precedence** コマンドをポリシー マップ クラス コンフィギュレーション モードで使用します。この設定をディセーブルにするには、このコマンドの **no** 形式を使用します。

[no] set precedence *precedence-value*

構文の説明

precedence-value	パケット ヘッダーに precedence ビットを 0 ~ 7 の範囲で指定します。

コマンド デフォルト

受信したパケットのプレシデンス値を保持します。

コマンド モード

policy-map（config-pmap）

使用上のガイドライン

このコマンドは、入力ポリシーまたは出力ポリシーに使用されることがあります。ただし、出力ポリシーマップ内でこのコマンドを発行した場合、このコマンドはクラスやキューの選択は変更しませんが、WRED ドロップ プロファイルの選択に影響することがあります。

プレシデンス値を設定することによって、パケットがネットワークを通過する際にパケットに対する QoS 処理を示します。

（注） DSCP を使用した差別化サービス アーキテクチャはプレシデンスの使用よりも大幅に優先されます。

set precedence コマンドと **set ip precedence** コマンドは同じように動作し、最も外側のレイヤ 3 ヘッダーが IPv4 または IPv6 のパケットをマークします。これに対して、**match ip precedence** コマンドは IPv4 パケットのみを分類し、**match precedence** コマンドは IPv4 と IPv6 の両方を分類します。

set precedence tunnel

まだパケットに追加されていないトンネル ヘッダー内に IP プレシデンス値を設定するには、**set precedence tunnel** コマンドをポリシー マップ クラス コンフィギュレーション モードで使用します。この設定をディセーブルにするには、このコマンドの **no** 形式を使用します。

[no] set precedence tunnel *precedence-value*

構文の説明

precedence-value	トンネル ヘッダーにプレシデンス ビットを 0~7 の範囲で指定します。

コマンド デフォルト

DSCP（および precedence 部分）がカプセル化されたヘッダーから新たにインポーズされたヘッダーへコピーされます。

コマンド モード

policy-map（config-pmap）

使用上のガイドライン

Cisco ASR シリーズ アグリゲーション サービス ルータでは、**set precedence tunnel** コマンドは IPv4 に対してのみサポートされています。サポートされている DSCP トンネルのマーキング設定をリストした表については、インポジション マーキング （167 ページ）を参照してください。

set qos-group

QoS グループ識別子（ID）をパケットに設定するには、ポリシー マップ クラス コンフィギュレーション モードで **set qos-group** コマンドを使用します。この設定をディセーブルにするには、このコマンドの **no** 形式を使用します。

[no] set qos-group *group-id*

構文の説明

group-id	QoS グループ ID を 0 ～ 99 の範囲で指定します。

コマンド デフォルト

QoS グループ ID はデフォルトで 0 に設定されます。

コマンド モード

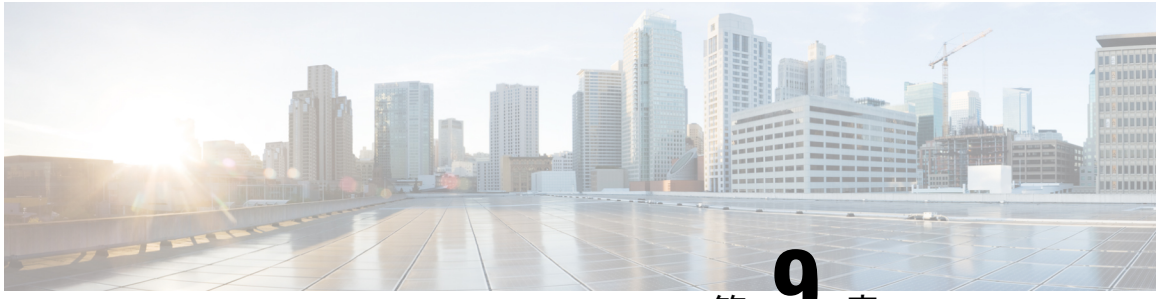
policy-map（config-pmap）

使用上のガイドライン

set qos-group コマンドでは、ルータによってパケットが処理されるため、グループ ID をパケットに関連付けることができます。

出力ポリシーでグループ ID を使用してパケットをサービスクラスに分類できます。従来、このアクションには意味がありませんでした。出力マーキングが行われる前にサービスクラスが

選択されるからです。ただし、カラー対応ポリシングを使用すると、出力ポリシーでの QoS グループ ID の設定には意味を持たせることができます。



第 9 章

QoS パケット一致統計情報の設定

QoS パケット一致統計情報機能は、次のサブ機能で構成されます。

- QoS パケット一致統計情報のフィルタ単位の機能では、ユーザが QoS クラスマップ内の個々のフィルタと一致する（match ステートメント）パケットおよびバイトの数をカウントし、表示することができます。

- QoS パケット一致統計情報の ACE 単位の機能では、ユーザがフィルタ内の個々のアクセス制御エントリ（ACE）に一致するパケットとバイトの数をカウントし、表示することができます。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「Bug Search Tool」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

QoS パケット一致統計情報機能の前提条件

ポリシーマップがシステム上のインターフェイスと関連付けられている場合は、フィルタ単位のQoSパケット一致統計情報機能をイネーブルまたはディセーブルにすることはできません。

ACE 単位の QoS パケット一致統計情報機能は、QoS パケット一致統計情報機能によって異なります。したがって、次の前提条件が適用されます。

- フィルタ単位の QoS パケット一致統計情報がイネーブルになっていない場合に、ユーザが ACE 単位の QoS パケット一致統計情報をイネーブルにしようとすると、この機能をイネーブルにするコマンドがCLIで拒否されます。情報メッセージが表示され、コマンドが拒否された理由がユーザに通知されます。

- ACE 単位の QoS パケット一致統計情報機能がイネーブルになっている場合に、ユーザがこの機能をディセーブルにしようとすると、この機能をディセーブルにするコマンドがCLIで拒否されます。情報メッセージが表示され、コマンドが拒否された理由がユーザに通知されます。

QoS パケット一致統計情報機能の制約事項

QoS のパケット一致統計情報機能をイネーブルにすると、拡張設定では CPU 使用率が増加する可能性があります。QoSのパケット一致統計情報機能をイネーブルにする前に、統計情報の利点とシステムの CPU 使用率の増加とを比較して検討する必要があります。

ここでは、QoSパケット一致統計情報のフィルタ単位の機能と、QoSパケット一致統計情報のACE 単位の機能に関する制約事項について説明します。

以降に、QoS パケット一致統計情報機能の制約事項を示します。

- QoSパケット一致統計情報のフィルタ単位の機能をイネーブルにすると、拡張された設定では CPU 使用率が増加する可能性があります。QoS パケット一致統計情報のフィルタ単位の機能をイネーブルにする前に、統計情報の利点とシステムの CPU 使用率の増加を比較して検討します。

- match-all クラスマップに対するQoSパケット一致統計情報のフィルタ単位の機能はサポートされていません。ただし、match-all クラス マップに対する QoS パケット一致統計情報の ACE 単位の機能はサポートされています。

次の表に、QoS パケット一致統計情報の ACE 単位のスケーリングの制約事項に関する情報を示します。

表 *9 : QoS* パケット一致統計情報：*ACE* 単位のスケーリングの制約事項

プラットフォーム	ACE（IPv4 または IPv6）
ASR1000-ESP5、ASR1001、ASR1002-F、ASR1002-X	25,000
ASR1000-ESP10	30,000
ASR1000-ESP20/ESP40/ESP100	30,000
ISR4400	20,000
CSR1000V	1,000

QoS パケット一致統計情報に関する情報

ここでは、フィルタ機能単位の QoS パケット一致統計情報とACE 機能単位の QoS パケット一致統計情報の概要を示します。

QoS パケット一致統計情報：フィルタ単位の機能の概要

QoS のパケット一致統計情報のフィルタ単位の機能では、フィルタに一致するパケット数およびバイト数をカウントし、表示することができます。

フィルタを定義するには、 **class-map** コマンドと **match-any** キーワードを使用します。次に例を示します。

```
class-map match-any my_class
   match ip precedence 4 <-------- User-defined filter
   match qos-group 10 <-------- User-defined filter
```

この情報を使用して、次のタスクを実行できます。

- ネットワークのセグメント上の音声トラフィックの量とデータ トラフィックの量の比較
- 帯域幅の可用性の調整
- 課金情報の正確な決定
- サービスに関する問題のトラブルシューティング

システムは、10秒のサイクルでパケット一致統計情報を収集します。インターフェイスやセッションが多数存在する場合、システムは各サイクルで約 8,000 のインターフェイスまたはセッションに関する統計情報を収集します。拡張された設定では、すべての統計情報を収集するのに 10 秒サイクルが数回必要になる場合があります。

QoS パケット一致統計情報：ACE 単位の機能の概要

QoS のパケット一致統計情報のACE 単位の機能では、ユーザが QoS ポリシー内に使用されている個々の ACE（クラス マップ内に使用されているアクセス グループ）に一致するパケットやバイトの数を追跡し、表示することができます。

この機能は、QoS ポリシーに使用する ACE 用のヒット カウンタを提供します。この機能を有効にすると、QoS ポリシーに使用されている ACE 用の QoS ヒット カウンタを、その特定の ACE 用の既存のセキュリティ アクセス リスト カウンタに追加します。アクセス リストのカウンタは、次のコマンド出力で確認できます。

```
Router# show ip access-lists

Extended IP access list A1
    10 permit ip 32.1.1.0 0.0.0.255 any (129580275 matches)
Extended IP access list A6and7
    10 permit ip 32.1.6.0 0.0.0.255 any (341426749 matches)
    20 permit ip 32.1.7.0 0.0.0.255 any (398245767 matches)
Extended IP access list source
    10 permit ip any host 16.1.1.5 (16147976 matches)
```

QoS ヒット カウンタ（QoS ポリシーで使用されている ACE の数）が、アクセス リスト カウンタに追加されます。この機能をイネーブルにするときは、次の点に注意することを推奨します。

- アクセス リストのカウントは、**show ip access-lists** コマンドの出力で確認できるように（インターフェイスに関する記述なし）、インターフェイス固有ではありません。これらは、ACE を使用し、そのカウントをサポートしているすべての機能のすべてのインターフェイスとすべての方向にわたる、すべてのヒットの総数です。

- インターフェイス固有のカウントは、QoS パケット一致統計情報のフィルタ単位の機能がイネーブルになっている場合は、既存の QoS コマンド（**show policy-map interface**）で提供されます。ただし、前に指定したコマンドでは、次の出力例に示すように、ACE 単位ではなく、フィルタ単位（ACL またはアクセス グループ）のカウントのみが示されます。

```
Router# show access-lists

Extended IP access list A1
    10 permit ip 32.1.1.0 0.0.0.255 any (2000 matches)

Router# show policy-map interface GigabitEthernet0/0/2

  Service-policy input: simple

    Class-map: A1-class (match-all)
      1000 packets, 124000 bytes
      5 minute offered rate 4000 bps
      Match: access-group name A1

    Class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 261000 bps, drop rate 0 bps
      Match: any
```

- QoS フィルタ（クラス マップ内の match ステートメント）に ACE があっても、パケットがそのステートメントに一致しない場合、ACE カウンタはそのパケットについては増加しません。これは次の場合に発生します。
  - ACE が deny ステートメントで使用されている。
  - match-all クラス マップの定義の他の一致基準（match ip prec 1 など）がパケットをクラスに一致させないようにしている。
  - match-any クラス マップの定義の他の一致基準（match ip prec 1 など）がパケットに一致しており、ACE 一致基準と一致させないようにしている。（このフィルタは ACE フィルタよりも優先され、パケットは両方のステートメントに一致しています）。
- アクセス リスト カウントは、ACE を使用し、ACE 単位のカウントをサポートするすべての機能のヒット カウントの（特定の ACE についての）総計です。（Cisco IOS XE3.10 では、セキュリティと QoS の ACL のみが ACE 単位のカウントをサポートしていますが、将来のリリースで変更される可能性があります）。したがって、1 つのパケットが同じ ACE を使用している複数の機能にヒットする（カウントされる）ため、同じパケットに対して（パケットが機能それぞれを通過するたびに）複数回カウントされることになります。次に、この例を示します。

```
ip access-list extended A1
  permit ip 32.1.1.0 0.0.0.255 any
class-map match-all A1-class
  match access-group name A1

interface GigabitEthernet0/0/2
 ip address 32.0.0.1 240.0.0.0
 ip access-group A1 in
 duplex auto
 speed auto
 media-type rj45
 no negotiation auto
 service-policy input simple


Router# show access-lists

Extended IP access list A1
    10 permit ip 32.1.1.0 0.0.0.255 any (2000 matches)

Router# show policy-map interface GigabitEthernet0/0/2

  Service-policy input: simple

    Class-map: A1-class (match-all)
      1000 packets, 124000 bytes
      5 minute offered rate 4000 bps
      Match: access-group name A1

    Class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 261000 bps, drop rate 0 bps
      Match: any
```

QoS パケット一致統計情報の設定方法

ここでは、QoS パケット一致統計情報の設定方法について説明します。

QoS パケット一致統計情報の設定：フィルタ単位

始める前に

- QoS パケット一致統計情報のフィルタ単位の機能をイネーブルにする前に、システム上のインターフェイスに関連付けられているポリシーマップがないことを確認します。ある場合は、システムによって次のメッセージが返されます。

```
Either a) A system RELOAD or
       b) Remove all service-policies, re-apply the change
            to the statistics, re-apply all service-policies
   is required before this command will be activated.
```

- QoS パケット一致統計情報のフィルタ単位の機能をイネーブルにする前に、**class-map** コマンドと **match-any** キーワードを使用するフィルタを定義していることを確認します。

（注） QoS パケット一致統計情報のフィルタ単位の機能をイネーブルにすると、拡張された設定では CPU 使用率が増加する可能性があります。QoS パケット一致統計情報のフィルタ単位の機能をイネーブルにする前に、統計情報の利点とシステムの CPU 使用率の増加を比較して検討します。

QoS パケット一致統計情報のフィルタ単位の機能を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **platform qos match-statistics per-filter**
4. **interface** *interface -name*
5. **service-policy** {**input** | **output**} *policy-map-name*
6. **end**
7. **show policy-map interface** *interface-name*
8. **configure terminal**
9. **interface** *interface-name*
10. **no service-policy** {**input** | **output**} *policy-map-name*
11. **exit**
12. **no platform qos match-statistics per-filter**
13. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**platform qos match-statistics per-filter** 例： `Router(config)# platform qos match-statistics per-filter`	QoS パケット一致統計情報のフィルタ単位の機能をイネーブルにします。
ステップ 4	**interface** *interface -name* 例： `Router(config)# interface GigabitEthernet0/0/0`	ポリシー マップを適用するためのインターフェイスを指定します。
ステップ 5	**service-policy** {**input** | **output**} *policy-map-name* 例： `Router(config-if)# service-policy input pol1`	インターフェイスに QoS ポリシー マップを適用します。QoS パケット一致統計情報機能は、QoS ポリシーを適用する前にイネーブルにする必要があります。
ステップ 6	**end** 例： `Router# end`	コンフィギュレーション モードを終了します。
ステップ 7	**show policy-map interface** *interface-name* 例： `Router# show policy-map interface serial4/0/0`	指定したインターフェイス、サブインターフェイス、またはインターフェイス上の特定の相手先固定接続（PVC）に存在するすべてのサービス ポリシーに対して設定されているすべてのクラスのパケット統計情報を表示します。
ステップ 8	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 9	**interface** *interface-name* 例： `Router(config)# interface GigabitEthernet0/0/0`	ポリシー マップを削除するためのインターフェイスを指定します。

	コマンドまたはアクション	目的
ステップ 10	**no service-policy** {**input** | **output**} *policy-map-name* 例： `Router(config-if)# no service-policy input pol1`	インターフェイスから QoS ポリシー マップを削除します。QoS パケット一致統計情報機能をディセーブルにする前に、すべての QoS ポリシーをインターフェイスから削除する必要があります。
ステップ 11	**exit** 例： `Router(config-if)# exit`	インターフェイス コンフィギュレーション モードを終了します。
ステップ 12	**no platform qos match-statistics per-filter** 例： `Router(config)# no platform qos match-statistics per-filter`	QoS パケット一致統計情報のフィルタ単位の機能をディセーブルにします。
ステップ 13	**end** 例： `Router# end`	コンフィギュレーション モードを終了します。

例

show policy-map interface コマンドを使用して、指定したインターフェイス、サブインターフェイス、またはインターフェイス上の PVC に存在するすべてのサービス ポリシーに対して設定されているすべてのクラスのパケット統計情報を表示します。

```
Router# show policy-map interface gig1/1/0

 GigabitEthernet1/1/0
  Service-policy input: pol1     ! target = gig1/1/0,input
    Class-map: class1 (match-any)
      1000 packets, 40000 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: ip precedence 1 <-------- User-defined filter
        800 packets, 32000 bytes <-------- Filter matching results
      Match: ip precedence 2 <-------- User-defined filter
        200 packets, 8000 bytes <-------- Filter matching results
      QoS Set
        ip precedence 7
          No packet marking statistics available
    Class-map: class-default (match-any)
      500 packets, 20000 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any <-------- User-defined filter
        500 packets, 20000 bytes <-------- Filter matching results
```

QoS パケット一致統計情報の設定：ACE 単位

始める前に

QoS パケット一致統計情報の ACE 単位の機能をイネーブルにする前に、QoS パケット一致統計情報のフィルタ単位の機能をイネーブルにしたことを確認します。

次に、**show platform hardware qfp active feature qos configuration global** コマンドを使用して機能の統計情報を確認する例を示します。

```
Router# show platform hardware qfp active feature qos configuration global
Marker statistics are: disabled
Match per-filter statistics are: enabled <<<<<<<
Match per-ace statistics are: enabled <<<<<<
Performance-Monitor statistics are: disabled
```

QoS パケット一致統計情報の ACE 単位の機能を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **platform qos match-statistics per-filter**
4. **platform qos match-statistics per-ace**
5. **interface** *interface-name*
6. **service-policy** {**input**|**output**} *policy-map-name*
7. **end**
8. **show policy-map interface** *interface-name*
9. **show access-lists**
10. **configure terminal**
11. **interface** *interface-name*
12. **no service-policy** {**input**|**output**} *policy-map-name*
13. **exit**
14. **no platform qos match-stat per-ace**
15. **no platform qos match-statistics per-filter**
16. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例：	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	`Router# configure terminal`	
ステップ 3	**platform qos match-statistics per-filter** 例： `Router(config)# platform qos match-statistics per-filter`	QoS パケット一致統計情報のフィルタ単位の機能をイネーブルにします。
ステップ 4	**platform qos match-statistics per-ace** 例： `Router(config)# platform qos match-statistics per-ace`	QoS パケット一致統計情報の ACE 単位の機能をイネーブルにします。
ステップ 5	**interface** *interface-name* 例： `Router(config)# interface GigabitEthernet0/0/0`	ポリシー マップを適用するためのインターフェイスを指定します。
ステップ 6	**service-policy**{**input**|**output**}*policy-map-name* 例： `Router(config-if)# service-policy input pol1`	インターフェイスに QoS ポリシー マップを適用します。QoS 一致統計情報機能は、QoS ポリシーを適用する前にイネーブルにする必要があります。
ステップ 7	**end** 例： `Router# end`	コンフィギュレーション モードを終了します。
ステップ 8	**show policy-map interface** *interface-name* 例： `Router# show policy-map interface serial4/0/0`	指定したインターフェイス、サブインターフェイス、またはインターフェイス上の特定の PVC のいずれかで、すべてのサービス ポリシーに対して設定されているすべてのクラスに関するパケット統計情報を表示します。
ステップ 9	**show access-lists** 例： `Router# show access-lists`	ACE 単位の QoS パケット一致統計情報を含めて、現在のアクセス リストのコンテンツを表示します。
ステップ 10	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 11	**interface** *interface-name* 例： `Router(config)# interface GigabitEthernet0/0/0`	ポリシー マップを削除するためのインターフェイスを指定します。

	コマンドまたはアクション	目的
ステップ 12	**no service-policy**{**input**|**output**}*policy-map-name* 例： `Router(config-if)# no service-policy input pol1`	インターフェイスから QoS ポリシー マップを削除します。QoS 一致統計情報機能をディセーブルにする前に、すべての QoS ポリシーをインターフェイスから削除する必要があります。
ステップ 13	**exit** 例： `Router(config-if)# exit`	インターフェイス コンフィギュレーション モードを終了します。
ステップ 14	**no platform qos match-stat per-ace** 例： `Router(config)# no platform qos match-stat per-ace`	QoS パケット一致統計情報の ACE 単位の機能をディセーブルにします。
ステップ 15	**no platform qos match-statistics per-filter** 例： `Router(config)# no platform qos match-statistics per-filter`	QoS パケット一致統計情報のフィルタ単位の機能をディセーブルにします。
ステップ 16	**end** 例： `Router# end`	コンフィギュレーション モードを終了します。

例

show policy-map interface コマンドを使用して、指定したインターフェイス、サブインターフェイス、またはインターフェイス上の PVC に存在するすべてのサービス ポリシーに対して設定されているすべてのクラスのフィルタ単位の統計情報を表示します。

```
Router# show policy-map interface GigabitEthernet0/0/2


  Service-policy input: test-match-types

    Class-map: A1orA2-class (match-any)
      482103366 packets, 59780817384 bytes
      5 minute offered rate 6702000 bps
      Match: access-group name A1
        62125633 packets, 7703578368 bytes
        5 minute rate 837000 bps
      Match: access-group name A2
        419977732 packets, 52077238892 bytes
        5 minute rate 5865000 bps

    Class-map: A3andprec1-class (match-all)
      5673520 packets, 703516480 bytes
      5 minute offered rate 837000 bps
```

```
    Match: access-group name A3
    Match: ip precedence 1

  Class-map: A5-class (match-all)
    227101820 packets, 28160625680 bytes
    5 minute offered rate 3351000 bps
    Match: access-group name A5

  Class-map: A6and7-class (match-all)
    627615840 packets, 77824340228 bytes
    5 minute offered rate 9215000 bps
    Match: access-group name A6and7

  Class-map: A3-class (match-all)
    111548288 packets, 13831987712 bytes
    5 minute offered rate 1675000 bps
    Match: access-group name A3

  Class-map: A4andsource (match-all)
    16115590 packets, 1998333160 bytes
    5 minute offered rate 2513000 bps
    Match: access-group name A4
    Match: access-group name source

  Class-map: class-default (match-any)
    164881212 packets, 20445270288 bytes
    5 minute offered rate 0 bps, drop rate 0 bps
    Match: any
```

show ip access-lists コマンドを使用して、現在のアクセス リストのコンテンツ（ACE 単位の QoS パケット一致統計情報を含む）を表示します。

```
Router# show ip access-lists

Extended IP access list A1
    10 permit ip 32.1.1.0 0.0.0.255 any (129580275 matches)
Extended IP access list A2
    10 permit ip 32.1.2.0 0.0.0.255 any (486342300 matches)
Extended IP access list A3
    10 permit ip 32.1.3.0 0.0.0.255 any (306738457 matches)
Extended IP access list A4
    10 permit ip 32.1.4.0 0.0.0.255 any (16147975 matches)
Extended IP access list A5
    10 permit ip 32.1.5.0 0.0.0.255 any (294357455 matches)
Extended IP access list A6and7
    10 permit ip 32.1.6.0 0.0.0.255 any (341426749 matches)
    20 permit ip 32.1.7.0 0.0.0.255 any (398245767 matches)
Extended IP access list source
    10 permit ip any host 16.1.1.5 (16147976 matches)
```

トラブルシューティングのヒント

QoS のパケット一致統計情報機能がイネーブルかどうかを確認するには、**show platform hardware qfp active feature qos config global** コマンドを使用します。この機能がディセーブルの場合は、次のようなメッセージが表示されます。

Router# **show platform hardware qfp active feature qos config global**

```
Marker statistics are: enabled
Match per filter statistics are: enabled
```

例：QoS パケット一致統計情報の設定：フィルタ単位

次に、QoS パケット一致統計情報を設定する例を示します。フィルタ単位で次のタスクを実行します。

- QoS パケット一致フィルタを定義する
- **show policy-map interface** コマンドの出力を表示します。

```
Router# show policy-map interface Tunnel1

  Service-policy output: DATA-OUT-PARENT
    Class-map: class-default (match-any)
      4469 packets, 4495814 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any <-------- User-defined filter
      Queueing
      queue limit 416 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 4469/4558380
      shape (average) cir 100000000, bc 400000, be 400000
      target shape rate 100000000
      Service-policy : DATA-OUT
        queue stats for all priority classes:
          Queueing
          queue limit 200 packets
          (queue depth/total drops/no-buffer drops) 0/0/0
          (pkts output/bytes output) 4469/4558380
        Class-map: ATM-VTI-RIP-SPK1-DATA (match-any)
          4469 packets, 4495814 bytes <-------- Filter matching results
          5 minute offered rate 0000 bps, drop rate 0000 bps
          Match: access-group 121 <-------- User-defined filter
            4469 packets, 4495814 bytes <-------- Filter matching results
            5 minute rate 0 bps
          QoS Set
            ip precedence 3
              Packets marked 4469
          Priority: 100 kbps, burst bytes 2500, b/w exceed drops: 0
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『Cisco IOS Master Commands List, All Releases』
Quality of Service コマンド	『*Cisco IOS Quality of Service Command Reference*』

標準

標準	タイトル
新しい規格または変更された規格はサポートされていません。また、既存の規格に対するサポートに変更はありません。	--

MIB

MIB	MIB のリンク
CISCO-CLASS-BASED-QOS-MIB	選択したプラットフォーム、Cisco ソフトウェア リリース、およびフィーチャ セットの MIB を検索してダウンロードする場合は、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

RFC

RFC	タイトル
新しい RFC または変更された RFC はサポートされていません。また、既存の RFC に対するサポートに変更はありません。	--

テクニカル サポート

説明	リンク
★枠で囲まれた Technical Assistance の場合★右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

QoS パケット一致統計情報の機能情報

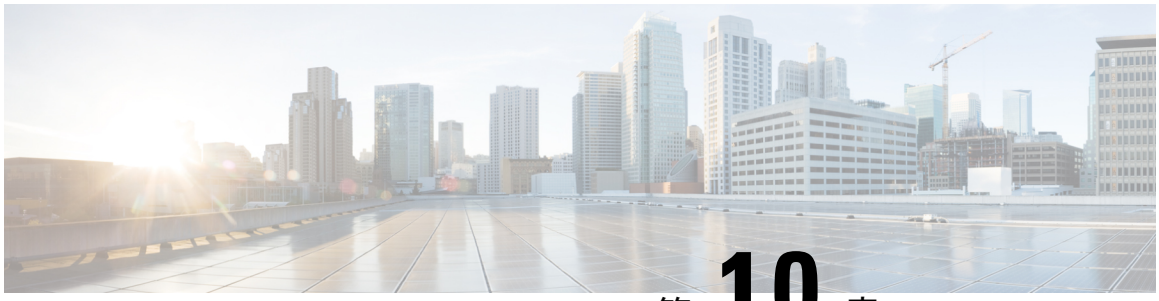
次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだ

けを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *10 : QoS* パケット一致統計情報の機能情報

機能名	リリース	機能情報
QoS パケット一致統計情報：フィルタ単位	Cisco IOS XE リリース 3.3S	QoS パケット一致統計情報のフィルタ単位の機能では、パケットの QoS サービス ポリシー内のクラスマップに使用されている個々のフィルタに一致する（match ステートメント）パケットの数をカウントし、表示することができます。 次のコマンドが導入または変更されました。 • **platform qos match-statistics per-filter** • **no platform qos match-statistics per-filter** • **show platform hardware qfp active feature qos config global**
QoS パケット一致統計情報：ACE 単位	Cisco IOS XE リリース 3.10S	QoS のパケット一致統計情報のACE 単位の機能では、ユーザが QoS ポリシー内に使用されている個々の ACE（クラスマップ内に使用されているアクセス グループ）に一致するパケットやバイトの数を追跡し、表示することができます。 次のコマンドが導入されました。 **platform qos match-statistics per-ace**



第 10 章

ポリサーを使用した ATM CLP 設定ビット

ポリサーを使用した ATM CLP 設定ビットの機能では、ポリシングして、アウトバウンドの PPP over ATM（PPPoA）トラフィックをマークすることができます。ATM セル損失率優先度（CLP）ビットは、次のいずれかの方法で設定できます。

- ポリシングされたしきい値
- クラスの一致

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「Bug Search Tool」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

ポリサーを使用した ATM CLP 設定ビットの前提条件

ポリシングされたしきい値を使用して ATM CLP ビットを設定する場合は、**set-clp-transmit** アクションがポリシーマップに含まれていることを確認します。新しいポリサーアクションは、

トラフィックが所定のレートを超えたときの ATM ネットワーク内の高い廃棄確率に一致するクラス内の PPPoA トラフィックを条件付きでマークします。

クラスの一致により ATM CLP ビットを厳格に設定する場合は、**set atm-clp** アクションがポリシー マップに含まれていることを確認します。set ディレクティブは、ATM ネットワーク内の高い廃棄確率に一致するクラス内のすべてのトラフィックをマークします。

set-clp-transmit アクションまたは **set atm-clp** アクションを使用して、ポリシー マップを仮想テンプレートに適用することができます。このテンプレートは、PPPoA セッションの作成時か、またはダイナミックに割り当てることで複製されます。

ポリサーを使用した ATM CLP 設定ビットに関する情報

ATM CLP ビット

ATM CLP ビットは、ATM セルのドロップ プライオリティを示します。ATM ネットワークの輻輳時に、ルータは CLP ビットが 1 に設定されている ATM セルを廃棄してから、CLP ビットが 0 に設定されているセルを廃棄します。

ポリサーを使用した ATM CLP ビットの設定機能を使用すると、セル ヘッダー内の ATM CLP ビットをイネーブルにするように **police** コマンドを設定できます。ATM CLP ビットは set ディレクティブで明示的にマークできます。

ポリサーを使用した ATM CLP ビットの設定機能は、次のタイプのポリシーで **set-clp-transmit** ポリシング アクションをサポートします。

- シングル レート ポリシング
- デュアルレート ポリシング
- 階層型

ポリサーを使用した ATM CLP ビットの設定方法

PPPoA ブロードバンドのトラフィック ポリシングの設定

始める前に

ポリシー マップを設定する前に、トラフィックを分類するために使用するクラス マップを定義していることを確認します。

手順の概要

1. **enable**
2. **configure terminal**

3. **policy-map** *policy-map-name*
4. **class** {*class-name*| **class-default**]
5. **police** [**cir** *cir*] [**conform-action** *action*] [**exceed-action** *action*]
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**policy-map** *policy-map-name* 例： `Device(config)# policy-map parent-policy`	ポリシー マップ コンフィギュレーション モードを開始し、ポリシー マップを作成します。
ステップ 4	**class** {*class-name*| **class-default**] 例： `Device(config-pmap)# class class-default`	ポリシー マップ クラス コンフィギュレーション モードを開始します。 作成または変更するポリシーのクラス名を指定するか、ポリシーを指定する前にデフォルト クラス（一般に class-default クラスといいます）を指定します。作成または変更する子クラスまたは親クラスを指定するのに必要な回数だけ、このコマンドを繰り返してください。 • *class name*：設定するクラス、またはポリシーを編集するクラスの名前。クラス名は、クラス マップに使用するとともに、ポリシー マップのクラスにポリシーを設定する場合にも使用します。 • **class-default** ポリシーを設定または変更できるようデフォルト クラスを指定します。
ステップ 5	**police** [**cir** *cir*] [**conform-action** *action*] [**exceed-action** *action*] 例： `Device(config-pmap-c)# police 1000000`	トラフィック ポリシングを設定し、指定のレートに準拠、超過、または違反としてマーク付けされたパケットに適用する複数のアクションを指定します。 • ポリシー マップ クラス ポリス コンフィギュレーション モードを開始します。1 つのアク

	コマンドまたはアクション	目的

	例：
	`Router(config-pmap-c-police)# conform-action`
	例：
	`transmit`
	例：
	`Device(config-pmap-c-police)# exceed-action`
	例：
	`set-clp-transmit`
		ションにつき 1 行を使用して、アクションを指定します。
		• **cir**：（任意）認定情報レート。CIR がトラフィック ポリシングに使用されることを示します。
		• **conform-action**：（任意）準拠バーストを下回るレートのパケットに対して実行するアクション。
		• **exceed-action**：（任意）準拠バースト以上で、準拠バーストと超過バーストの合計以下のレートのパケットに対して実行するアクション。

ステップ 6	**end** 例： `Device(config-pmap-c)# end`	（任意）特権 EXEC モードに戻ります。

例

次に、ポリサーを使用して ATM CLP を設定する例を示します。

```
policy-map egress_atm_clp_policer
 class prec0
  police cir 5000000
 class prec1
  police cir 3000000 conform-action transmit exceed-action set-clp-transmit
 class class-default
  police cir 1000000 conform-action transmit exceed-action set-clp-transmit
```

ATM CLP ビットのマーキング

始める前に

ポリシー マップを設定する前に、トラフィックを分類するために使用するクラス マップを定義していることを確認します。

手順の概要

1. **enable**
2. **configure terminal**
3. **policy-map policy-map-name**
4. **class** {*class-name*| **class-default**]
5. **set atm-clp**

6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**policy-map policy-map-name** 例： `Router(config)# policy-map parent-policy`	ポリシー マップ コンフィギュレーション モードを開始し、ポリシー マップを作成します。
ステップ 4	**class** {*class-name*| **class-default**] 例： `Router(config-pmap)# class class-default`	ポリシー マップ クラス コンフィギュレーション モードを開始します。 作成または変更するポリシーのクラス名を指定するか、ポリシーを指定する前にデフォルト クラス（一般に class-default クラスといいます）を指定します。作成または変更する子クラスまたは親クラスを指定するのに必要な回数だけ、このコマンドを繰り返してください。 • *class name*：設定するクラス、またはポリシーを編集するクラスの名前。クラス名は、クラス マップに使用するとともに、ポリシー マップのクラスにポリシーを設定する場合にも使用します。 • **class-default** ポリシーを設定または変更できるようデフォルト クラスを指定します。
ステップ 5	**set atm-clp** 例： `Router(config-pmap-c)# set atm-clp`	このクラスに一致するすべてのトラフィックに対して ATM CLP ビットのマーキングを設定します。
ステップ 6	**end** 例： `Router(config-pmap-c)# end`	（任意）特権 EXEC モードに戻ります。

例

次に、明示的なマーキングを使用して ATM CLP を設定する例を示します。

```
policy-map egress_atm_clp_policer
 class prec0
  police cir 5000000
 class class-default
  set atm-clp
```

ポリサーを使用した ATM CLP 設定ビットの設定例

例：クラスに一致するポリサーアクションによる ATM CLP のマーキング

この例では、次のタスクの実行方法を説明します。

- トラフィック クラスを定義する。
- 2 レイヤ ポリシー マップを設定する。
- PPPoA セッションにポリシー マップを適用する。

このポリシーは、クラスのトラフィックが所定のレートを超えたときに、一致する low_interest クラスのトラフィックに ATM CLP ビットを条件付きでマークします。

```
class-map voice
  match precedence 4
!
class-map web
  match precedence 3
!
class low_interest
  match precedence 1 0
!
policy-map child
 child class voice
  police cir 256000
  priority level 1
 class web
  bandwidth remaining ratio 10
 class low_interest
  police cir 1000000 conform-action transmit exceed-action set-clp-transmit
 class class-default
  bandwidth remaining ratio 1
!
policy-map parent
 class class-default
  shape average 15000000
  service-policy child
```

仮想テンプレートに適用されたポリシー マップが複製され、各 PPPoA セッションに対して仮想アクセス インターフェイスを作成するために使用されます。

```
interface Virtual-Template1
 ip unnumbered Loopback1
 load-interval 30
 peer default ip address pool POOL1
 ppp authentication chap ppp
 ipcp address required
 service-policy output parent
```

例：ポリサー アクションのポリシングされたしきい値による ATM CLP のマーキング

この例では、次のタスクの実行方法を説明します。

- トラフィック クラスを定義する。
- 2 レイヤ ポリシー マップを設定する。
- PPPoA セッションにポリシー マップを適用する。

このポリシーは、ATM ネットワークに輻輳が発生したときにドロップされるように、必須以外のトラフィックを ATM CLP ビッドを使用してマークします。

```
class-map video
  match precedence 5
!
class-map voice
  match precedence 4
!
class-map web
  match precedence 3
!
policy-map child
 child class voice
  police cir 256000
  priority level 1
 class video
  police cir 4000000
  priority level 2
 class web
  set atm-clp
  bandwidth remaining ratio 10
 class class-default
  bandwidth remaining ratio 1
  set atm-clp
!
interface Virtual-Template1
 ip unnumbered Loopback1
 load-interval 30
 peer default ip address pool POOL1
 ppp authentication chap ppp
 ipcp address required
 service-policy output parent
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『Cisco IOS Master Commands List, All Releases』
Quality of Service コマンド	『*Cisco IOS Quality of Service Command Reference*』

標準

標準	タイトル
この機能でサポートされる新規の規格または変更された規格はありません。また、既存の規格のサポートは変更されていません。	--

MIB

MIB	MIB のリンク
この機能によってサポートされる新しい MIB または変更された MIB はありません。またこの機能による既存 MIB のサポートに変更はありません。	選択したプラットフォーム、Cisco ソフトウェア リリース、およびフィーチャ セットの MIB を検索してダウンロードする場合は、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	--

テクニカル サポート

説明	リンク
★枠で囲まれた Technical Assistance の場合★右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

ポリサーを使用した ATM CLP 設定ビットの機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *11* : ポリサーを使用した *ATM CLP* 設定ビットの機能情報

機能名	リリース	機能情報
ポリサーを使用した ATM CLP 設定ビット	Cisco IOS リリース XE 3.3S Cisco IOS リリース XE 3.14S	ポリサーを使用した ATM CLP 設定ビットの機能では、ポリシングが可能であり、アウトバウンド PPPoA トラフィックをマークします。 Cisco IOS リリース 3.14S では、この機能は Cisco 4451-X サービス統合型ルータに追加されました。 次のコマンドが導入または変更されました：**set atm-clp**、**police**。

第 11 章

EVC Quality of Service

このマニュアルでは、イーサネット仮想回線（EVC）で使用する Quality of Service（QoS）機能（トラフィックの分類やポリシングなど）をイネーブルにする方法について説明します。

Metro Ethernet Forum で定義された EVC は、ポートレベルのポイントツーポイントまたはマルチポイントツーマルチポイント回線です。プロバイダーから顧客に提供されているサービスの 1 つのインスタンスをエンドツーエンドで表します。さまざまなパラメータが統合されて、サービスが提供されます。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「Bug Search Tool」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

EVC での Quality of Service に関する情報

EVC Quality of Service と MQC

QoS 機能は、通常、トラフィック クラス、クラス マップ、ポリシー マップを使用して適用します。たとえば、特定のクラスに属するトラフィックを特定のカテゴリに分類し、特定の QoS 処理（分類やポリシングなど）を実行するように指定できます。トラフィックに対して実行される特定の QoS 処理はポリシー マップで指定し、ポリシー マップはインターフェイスに適用します。このように QoS を適用するために使用されるメカニズムはモジュラ QoS CLI（MQC）です。

ポリシー マップは、**service-policy** コマンドを使用して、着信（入力）または発信（出力）方向でインターフェイスに適用できます。

MQC 構造では、トラフィック クラスの定義、トラフィック ポリシーの作成、インターフェイス（この場合、EVC）へのトラフィック ポリシーの適用を行えます。

MQC 構造は、大きく次の 3 つの手順からなります。

1. **class-map** コマンドを使用して、トラフィック クラスを定義します。トラフィック クラスは、トラフィックの分類に使用します。
2. **policy-map** コマンドを使用して、トラフィック ポリシーを作成します（トラフィック ポリシーという用語とポリシーマップという用語は、多くの場合同じ意味で使用されます）。トラフィック ポリシー（ポリシー マップ）には、1 つのトラフィック クラスと、トラフィック クラスに適用する 1 つ以上の QoS 機能を含めます。トラフィック ポリシー内の QoS 機能によって、分類されたトラフィックの処理方法が決まります。
3. **service-policy** コマンドを使用して、インターフェイスにトラフィック ポリシー（ポリシー マップ）を適用します。

（注） 階層型ポリシー マップやクラス マップを含む MQC の詳細については、「MQC を使用した QoS 機能の適用」モジュールを参照してください。

QoS 対応イーサネット フロー ポイント（EFP）

EVC Quality of Service と MQC （222 ページ）で説明したように、MQC はネットワーク トラフィックに 1 つまたは複数の QoS 機能を適用するために使用されます。MQC を使用する最後のステップでは、**service-policy** コマンドを使用して、トラフィック ポリシー（ポリシー マップ）をインターフェイス（この場合は EVC）に適用します。

EVC Quality of Service 機能では、**service-policy** コマンドを使用して、EVC の着信（入力）方向または発信（出力）方向のイーサネット フロー ポイント（EFP）にポリシー マップを適用できます。このようにして、EFP は「QoS 対応」と認識されます。

QoS 機能と EVC

QoS の特定の機能として、次が含まれています。

- パケットの分類（Diffserv コード ポイント（DSCP）値や QoS グループ ID などに基づく）
- パケット マーキング（サービス クラス（CoS）値）に基づくなど）
- トラフィック ポリシング（2 色、3 色および複数アクション）
- 帯域幅の共有
- プライオリティ キューイング（EVC のアウトバウンド方向のみ）
- 重み付けランダム早期検出（WRED）

QoS 機能は、次の項の一覧に示す適切なコマンドを使用してイネーブルにします。

EVC QoS がサポートするトラフィック分類の match コマンド

次の表に、EVC でのトラフィックの分類時にで使用できる **match** コマンドの一部を示します。使用可能な **match** コマンドは、Cisco IOS XE リリースによって異なります。コマンドおよびコマンド シンタックスの詳細については、『Cisco IOS Quality of Service Solutions Command Reference』を参照してください。

表 *12 : MQC* で使用可能な *match* コマンド

Command	目的
match access-group	指定したアクセス コントロール リスト（ACL）に基づいて、クラス マップの一致基準を設定します。
match any	すべてのパケットの一致基準を設定します。
match cos	レイヤ 2 CoS マーキングに基づいてパケットを照合します。
match cos inner	レイヤ 2 CoS マーキングに基づいて QinQ パケットの内部 CoS を照合します。
match [ip] dscp	特定の IP DSCP 値を一致基準として識別します。1 つの match 文に最大 8 つの DSCP 値を含めることができます。

Command	目的
match not	成功しない一致基準として使用する、1 つの一致基準値を指定します。 （注） **match not** コマンドは、一致基準として使用する特定の match パラメータを指定する代わりに、パケットがクラスのメンバとして分類されるのを防ぐ一致基準を指定するために使用します。たとえば、トラフィック クラスの設定中に **match not qos-group 6** コマンドを発行すると、QoS グループ 6 だけが、成功する一致基準として考慮されない QoS グループ値となります。他の QoS グループ値は成功する一致基準となります。
match [ip] precedence	IP プレシデンス値を一致基準として識別します。
match qos-group	特定の QoS グループ値を一致基準として識別します。
match source-address mac	送信元 MAC アドレスを一致基準として使用します。 （注） **match source-address mac** コマンドを使用したトラフィックの分類は、入力方向でのみサポートされます。
match vlan (QoS)	VLAN ID 番号に基づいてトラフィックを照合し、分類します。
match vlan inner	802.1q のタグ付きフレームの最も内部にある VLAN ID を照合するクラス マップを設定します。

1 つのトラフィック クラスでの複数の match コマンド

トラフィック クラスに複数の **match** コマンドが含まれている場合、**match** コマンドの評価方法を指定する必要があります。これは、**class-map** コマンドの **match-any** キーワードまたは **match-all** キーワードを使用して指定します。**match-any** キーワードと **match-all** キーワードについては、次の点に注意してください。

- **match-any** キーワードを指定した場合、トラフィック クラスによって評価されるトラフィックは、指定した基準の *1* つに一致する必要があります。
- **match-all** キーワードを指定した場合、トラフィック クラスによって評価されるトラフィックは、指定した基準の すべてに一致する必要があります。
- どちらのキーワードも指定しなかった場合、トラフィック クラスによって評価されるトラフィックは、指定した基準のすべてに一致する必要があります（つまり、**match-all** キーワードの動作が使用されます）。

EVC で QoS 機能をイネーブルにするために使用するコマンド

QoS 機能をイネーブルにするために使用するコマンドは、Cisco IOS XE リリースごとに異なります。以下の表に、使用可能なコマンドとそれによってイネーブルになる QoS 機能の一部を示します。コマンドシンタックスについては、『Cisco IOS Quality of Service Solutions Command Reference』を参照してください。

イネーブルにする特定の QoS 機能の詳細については、『Cisco IOS Quality of Service Solutions Configuration Guide』の適切なモジュールを参照してください。

表 *13 : QoS* 機能をイネーブルにするために使用するコマンド

コマンド	目的
bandwidth	クラスの最小帯域幅保証を設定します。
bandwidth remaining	クラスの過剰重量を設定します。
drop	指定したトラフィック クラスのパケットを廃棄します。
fair-queue	トラフィック クラス内のフローベースのキューイング機能をイネーブルにします。
police	トラフィック ポリシングを設定します。複数のポリシング アクションを指定できるようにします。
police (percent)	インターフェイスで利用可能な帯域幅の割合に基づいてトラフィック ポリシングを設定します。
police (two rates)	認定情報レート（CIR）と最大情報レート（PIR）の2つのレートを使用したトラフィック ポリシングを設定します。
priority	ポリシー マップに属するトラフィックのクラスにプライオリティを与えます。
queue-limit	ポリシーマップで設定されているクラスに対してキューが保持できるパケットの最大数を指定または変更します。
random-detect	重み付けランダム早期検出（WRED）をイネーブルにします。
random-detect cos-based	パケットのサービス クラス（CoS）値に基づいて、Weighted Random Early Detection（WRED）をイネーブルにします。
random-detect dscp-based	パケットの廃棄確率を計算する場合、Weighted Random Early Detection（WRED）で DiffServ コード ポイント（DSCP）の値を使用するよう、指定します。
random-detect discard-class	ポリシー マップ内のクラスの discard-class 値に対し、WRED パラメータを設定します。

コマンド	目的
random-detect discard-class-based	パケットの廃棄クラス値に基づく WRED を設定します。
random-detect exponential-weighting-constant	クラス用に予約されたキューの平均キューサイズ計算用の指数加重係数を設定します。
random-detect precedence	ポリシー マップ内のクラス ポリシーに対する、特定の IP プレシデンスの WRED パラメータを設定します。
service-policy	一致基準として使用するトラフィック ポリシーの名前を指定します（トラフィック ポリシーを互いにネストさせるため（階層型トラフィック ポリシー））。
set cos	発信パケットのレイヤ 2 CoS 値を設定します。
set cos-inner	ブリッジ フレームの内部サービス クラス フィールドをマークします。
set discard-class	discard-class 値でパケットをマークします。
set [ip] dscp	タイプ オブ サービス（ToS）バイト内の DSCP 値を設定することでパケットをマークします。
set mpls experimental	パケットが指定したポリシーマップに一致する場合にマルチプロトコル ラベル スイッチング（MPLS）ビットを設定する値を指定します。
set precedence	パケット ヘッダーにプレシデンス値を設定します。
set qos-group	後でパケットを分類するために使用できる QoS グループ ID を設定します。
shape	指定したアルゴリズムに従って、指示されたビット レートまでトラフィックをシェーピングします。

service-policy コマンドの input および output キーワード

一般的な規則として、トラフィック ポリシーで設定する QoS 機能は、インターフェイスで受信されるパケットか、インターフェイスで送信されるパケットに適用できます。そのため、**service-policy** コマンドを使用するとき、**input** または **output** キーワードを使用することで、トラフィック ポリシーの方向を指定する必要があります。

たとえば、**service-policy output policy-map1** コマンドは、トラフィック ポリシーの QoS 機能を出力方向のインターフェイスに適用します。インターフェイス（出力）から送信されるすべてのパケットが、policy-map1 という名前のトラフィック ポリシーで指定された基準に従って評価されます。

（注） Cisco のリリースでは、キューイング メカニズムは入力方向ではサポートされていません。非キューイング メカニズム（トラフィック ポリシングやトラフィック マーキングなど）は、入力方向でサポートされています。また、送信元 MAC アドレスに基づくトラフィックの分類（**match source-address mac** コマンドを使用）は、入力方向でのみサポートされています。

EVC での Quality of Service 機能の設定方法

EVC で使用するトラフィック クラスの作成

トラフィック クラスを作成するには、**class-map** コマンドを使用してトラフィック クラス名を指定します。次に、1 つ以上の **match** コマンドを使用して、適切な一致基準を指定します。指定した基準に一致するパケットがトラフィック クラスに分類されます。

EVC で使用するトラフィック クラスを作成するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **class-map** [**match-all** | **match-any**] *class-name*
4. **match cos** *cos-number*
5. 必要に応じて追加の **match** コマンドを入力します。追加のコマンドが不要な場合は次のステップに進みます。
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**class-map** [**match-all**	**match-any**] *class-name* 例：	クラスマップを作成し、クラスマップ コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	`Router(config)# class-map match-any class1`	• クラスマップは、パケットを指定したクラスに照合するために使用します。 （注） **match-all** キーワードは、すべての一致基準が満たされることが必要であることを指定します。**match-any** キーワードは、いずれかの一致基準が満たされることが必要であることを指定します。これらのキーワードは、複数の **match** コマンドを指定する場合にだけ使用します。
ステップ 4	**match cos** *cos-number* 例： `Router(config-cmap)# match cos 2`	レイヤ 2 CoS 番号に基づいて、パケットを照合します。 （注） **match cos** コマンドは、使用できる **match** コマンドの一例です。
ステップ 5	必要に応じて追加の **match** コマンドを入力します。追加のコマンドが不要な場合は次のステップに進みます。	--
ステップ 6	**end** 例： `Router(config-cmap)# end`	（任意）クラスマップ コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

EVC で使用するポリシー マップの作成

EVC で使用するトラフィック ポリシー（ポリシー マップ）を作成するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-map-name*
4. **class** {*class-name*| **class-default**}
5. **police** *bps* [*burst-normal*] [*burst-max*] [**conform-action** *action*] [**exceed-action** *action*] [**violate-action** *action*]
6. イネーブルにする追加の QoS 機能に対するコマンドを入力します。他に QoS 機能が必要ない場合は、次のステップに進みます。
7. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。

ステップ 3	**policy-map** *policy-map-name*
	例：
	`Router(config)#`
	`policy-map policy1`
		トラフィック ポリシーの名前を作成または指定し、QoS ポリシー マップ コンフィギュレーション モードを開始します。

ステップ 4	**class** {*class-name*| **class-default**}
	例：
	`Router(config-pmap)#`
	`class class1`
		クラスの名前を指定し、QoS ポリシー マップ クラス コンフィギュレーション モードを開始します。
		（注） この手順により、トラフィック クラスがトラフィック ポリシーに関連付けられます。

ステップ 5	**police** *bps* [*burst-normal*] [*burst-max*] [**conform-action** *action*] [**exceed-action** *action*] [**violate-action** *action*] 例： `Router(config-pmap-c)# police 3000`	（任意）トラフィック ポリシングを設定します。 （注） **police** コマンドは、QoS 機能をイネーブルにするためにポリシー マップで使用できるコマンドの一例です。
ステップ 6	イネーブルにする追加の QoS 機能に対するコマンドを入力します。他に QoS 機能が必要ない場合は、次のステップに進みます。	--
ステップ 7	**end** 例： `Router(config-pmap-c)# end`	（任意）QoS ポリシー マップ クラス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

EVC の設定および EVC へのトラフィック ポリシーの適用

ポリシーマップを EVC に適用すると、トラフィック ポリシー（ポリシーマップ）は、イネーブルにした QoS 機能をトラフィック クラスに適用します。

EVC を設定し、EVC にトラフィック ポリシーを適用するには、次の手順を実行します。

（注） EVC にトラフィック ポリシーを適用するために使用するコマンドの 1 つに、**service-policy** コマンドがあります。このコマンドを使用する場合は、ポリシー マップ名とともに **input** キーワードまたは **output** キーワードを指定する必要があります。ポリシー マップには、使用する QoS 機能が含まれます。特定の QoS 機能は、入力方向または出力方向のいずれかでのみ使用できます。これらのキーワードとサポートされる QoS 機能の詳細については、service-policy コマンドの input および output キーワード（226 ページ）を参照してください。また、複数の EVC を含むインターフェイスにトラフィック ポリシーを適用する場合、トラフィック ポリシーはインターフェイス上のすべての EVC に適用されます。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface** *interface-type interface-number*
4. **service instance** *id* **ethernet** [*evc-name*]
5. **encapsulation dot1q** *vlan-id* [,*vlan-id*[-*vlan-id*]] [**native**]
6. **rewrite ingress tag translate 1-to-1 dot1q** *vlan-id* **symmetric**
7. **bridge domain** *domain-number*
8. **service-policy** {**input** | **output**} *policy-map-name*
9. **end**
10. **show policy-map interface** *type number* **service instance** *service-instance-number*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface** *interface-type interface-number* 例： `Router(config)# interface gigabitethernet 0/0/1`	インターフェイス タイプを設定し、インターフェイス コンフィギュレーション モードを開始します。 • インターフェイス タイプとインターフェイス番号を入力します。

	コマンドまたはアクション	目的
ステップ 4	**service instance** *id* **ethernet** [*evc-name*] 例： `Router(config-if)# service instance 333 ethernet evc1`	インターフェイスでイーサネット サービス インスタンスを設定し、イーサネット サービス コンフィギュレーション モードを開始します。 • サービス インスタンス ID と、該当する場合は EVC 名（任意）を入力します。
ステップ 5	**encapsulation dot1q** *vlan-id* [,*vlan-id*[-*vlan-id*]] [**native**] 例： `Router(config-if-srv)# encapsulation dot1q 10`	インターフェイスの 802.1Q フレーム入力を適切なサービス インスタンスにマップするための一致基準を定義します。
ステップ 6	**rewrite ingress tag translate 1-to-1 dot1q** *vlan-id* **symmetric** 例： `Router(config-if-srv)# rewrite ingress tag translate 1-to-1 dot1q 300 symmetric`	サービス インスタンスに入るフレームで実行されるカプセル化調整を指定します。
ステップ 7	**bridge domain** *domain-number* 例： `Router(config-if-srv)# bridge domain 1`	ブリッジ ドメインを設定します。 • ブリッジ ドメイン番号を入力します。

ステップ 8	**service-policy** {**input** | **output**} *policy-map-name*
	例：
	`Router(config-if-srv)#`
	`service-policy input policy1`
		インターフェイスにポリシーマップを適用します。
		• **input** キーワードまたは **output** キーワードとポリシー マップ名を入力します。

ステップ 9	**end** 例： `Router(config-if-srv)# end`	（任意）特権 EXEC モードに戻ります。
ステップ 10	**show policy-map interface** *type number* **service instance** *service-instance-number* 例： `Router# show policy-map interface gigabitethernet 1/0/0 service instance 30`	（任意）インターフェイスに適用された入力ポリシーと出力ポリシーの統計情報と設定を表示します。 • インターフェイス タイプ、インターフェイス番号、サービス インスタンス番号を入力します。

EVC QoS の設定例

例：EVC で使用するトラフィック クラスの作成

この例では、CoS 値が 2 のトラフィックが、class1 と呼ばれるトラフィック クラスに置かれます。

```
Router> enable

Router# configure terminal

Router(config)# class-map match-any class1

Router(config-cmap)# match cos 2

Router(config-cmap)# end
```

例：EVC で使用するポリシー マップの作成

この例では、トラフィック ポリシングは、policy1 と呼ばれるポリシー マップで設定されています。トラフィック ポリシングは、class1 のトラフィックに適用される QoS 機能です。

```
Router> enable

Router# configure terminal

Router(config)#
 policy-map policy1

Router(config-pmap)#
 class class1

Router(config-pmap-c)# police 3000

Router(config-pmap-c)# end
```

例：EVC の設定と EVC へのトラフィック ポリシーの適用

この例では、EVC が設定され、policy1 と呼ばれるトラフィック ポリシーが EVC に適用されています。

```
Router> enable

Router# configure terminal

Router(config)# interface gigabitethernet 0/0/1

Router(config-if)# service instance 333 ethernet evc1

Router(config-if-srv)# encapsulation dot1q 10

Router(config-if-srv)# rewrite ingress tag translate 1-to-1 dot1q 300 symmetric

Router(config-if-srv)# bridge domain 1

Router(config-if-srv)# service-policy input policy1

Router(config-if-srv)# end
```

例：EVCのトラフィック クラスおよびトラフィック ポリシー情報の確認

次に、**show policy-map interface service instance** コマンドの出力例を示します。GigabitEthernet インターフェイス 1/1/7 の EFP 用に設定され、EFP に適用される QoS 機能が表示されます。

```
Router# show policy-map interface gigabitethernet 1/1/7 service instance 10
 GigabitEthernet1/1/7: EFP 10
  Service-policy input: multiaction
    Class-map: c1 (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: ip precedence 3
      police:
          cir 300000 bps, bc 2000 bytes
        conformed 0 packets, 0 bytes; actions:
          set-prec-transmit 7
          set-qos-transmit 10
        exceeded 0 packets, 0 bytes; actions:
          drop
        conformed 0000 bps, exceed 0000 bps
    Class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『Cisco IOS Master Commands List, All Releases』
QoS コマンド：コマンド構文の詳細、コマンドモード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『*Cisco IOS Quality of Service Solutions Command Reference*』
パケット分類	『Classifying Network Traffic』モジュール
選択的パケット廃棄	「IPv6 Selective Packet Discard」モジュール

標準

標準	タイトル
新しい規格または変更された規格はサポートされていません。また、既存の規格に対するサポートに変更はありません。	--

MIB

MIB	MIB のリンク
新しい MIB または変更された MIB はサポートされていません。また、既存の MIB に対するサポートに変更はありません。	選択したプラットフォーム、Cisco IOS リリース、およびフィーチャ セットに関する MIB を探してダウンロードするには、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

RFC

RFC	タイトル
新しい RFC または変更された RFC はサポートされていません。また、既存の RFC に対するサポートに変更はありません。	--

テクニカル サポート

説明	リンク
★枠で囲まれた Technical Assistance の場合★右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

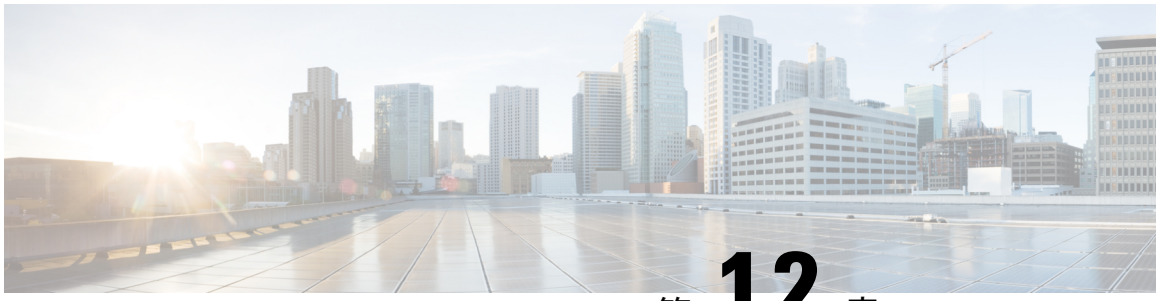
EVC QoS を設定するための機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *14 : EVC QoS* の機能情報

機能名	リリース	機能情報
EVC Quality of Service	Cisco IOS XE リリース 3.3 Cisco IOS リリース 15.5(2)T	このマニュアルでは、イーサネット仮想回線（EVC）で使用する QoS 機能（トラフィックの分類やポリシングなど）をイネーブルにする方法について説明します。 EVC QoS 機能は、Cisco ASR 1000 シリーズのアグリゲーション サービス ルータで導入されました。 次のコマンドが導入または変更されました：**service-policy**、**show policy-map interface service instance**。



第 12 章

EtherChannel インターフェイスの QoS

Cisco ASR 1000 シリーズ ルータのイーサネット チャネル（EtherChannel）インターフェイスで Quality of Service（QoS）がサポートされています。QoS 機能は、いくつかの Cisco IOS XE リリースを経て進化しており、ソフトウェア レベル、EtherChannel の設定、および設定したモジュラ QoS CLI（MQC）機能に基づいたさまざまな能力を持っています。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「Bug Search Tool」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

EtherChannel の QoS に関する情報

QoS 機能を備えた EtherChannel の進化

EtherChannel は、複数の物理リンクのグループを作成し、スイッチ、ルータ、およびサーバ間に障害耐性と高速リンクを提供するための論理イーサネット リンクを 1 つ作成できるポート

チャネル アーキテクチャです。EtherChannel は、2 ～ 8 つのアクティブな高速のギガビットまたは 10 ギガビットのイーサネット ポート間に、他のアクティブなポートに障害が発生したときにアクティブになる 1 ～ 8 つの非アクティブな（フェールオーバー）ポートとともに作成されます。

EtherChannel インターフェイスの QoS は、Cisco IOS XE がリリースされるたびに進化してきました。現在のレベルの Cisco IOS XE ソフトウェアに得られるサポートのレベルと、基礎となる EtherChannel 設定を理解することが重要です。EtherChannel の設定方法に基づいて、QoS のさまざまな組み合わせがサポートされます。EtherChannel は、次の 3 つのモードで設定できます。

- ポートチャネル サブインターフェイスのカプセル化 CLI を使用した EtherChannel VLAN ベースのロード バランシング
- LACP による EtherChannel アクティブ/スタンバイ（EtherChannel のロード バランシングなし）
- LACP によるロード バランシングを備えた EtherChannel

これらの各モデルには、Cisco IOS XE ソフトウェアに含まれるサポートのレベルと、それぞれによって可能な QoS 設定に関する特定の制約があります。

次に、サポートされる EtherChannel と QoS 設定のさまざまな組み合わせをまとめます。設定例は、このドキュメントで後述します。特に併記のない限り、所定の EtherChannel 設定に対する別の論理インターフェイスと物理インターフェイスでのサービス ポリシーの組み合わせはサポートされていません。

ポートチャネル サブインターフェイスのカプセル化 CLI を使用した EtherChannel VLAN ベースのロード バランシング

Cisco IOS XE リリース 2.1 以降で次がサポートされています。

- ポートチャネル サブインターフェイスでの出力 MQC キューイングの設定
- ポートチャネルのメンバー リンクでの出力 MQC キューイングの設定
- QoS のポリシー集約：サブインターフェイスでの出力 MQC キューイング
- ポートチャネル サブインターフェイスでの入力ポリシングとマーキング
- ポートチャネル メンバー リンクでの出力ポリシングとマーキング

Cisco IOS XE リリース 2.6 以降で次がサポートされています。

- QoS ポリシー集約：メイン インターフェイスでの複数キュー集約に対する MQC サポート：メイン インターフェイスでの出力 MQC キューイング

LACP による EtherChannel アクティブ/スタンバイ（EtherChannel のロード バランシングなし）

Cisco IOS XE 2.4 以降で次がサポートされています。

- ポートチャネル メンバー リンクでの出力 MQC キューイング：EtherChannel ロード バランシングなし

LACP による EtherChannel とロード バランシング

Cisco IOS XE 2.5 以降で次がサポートされています。

- ポートチャネル メンバー リンクでの出力 MQC キューイング設定：EtherChannel ロード バランシング

Cisco IOS XE 3.12 以降で次がサポートされています。

- ポートチャネル メインインターフェイスでの一般的な MQC QoS のサポート

  QoS に関するベスト プラクティスとして、ポートチャネル集約を使用することをお勧めします（「集約 EtherChannel QoS 」の章を参照）。

Cisco IOS XE 3.16.3 以降および Cisco IOS XE Fuji 16.3 以降で次がサポートされています。

- ポートチャネル サブインターフェイスでの一般的な MQC QoS のサポート

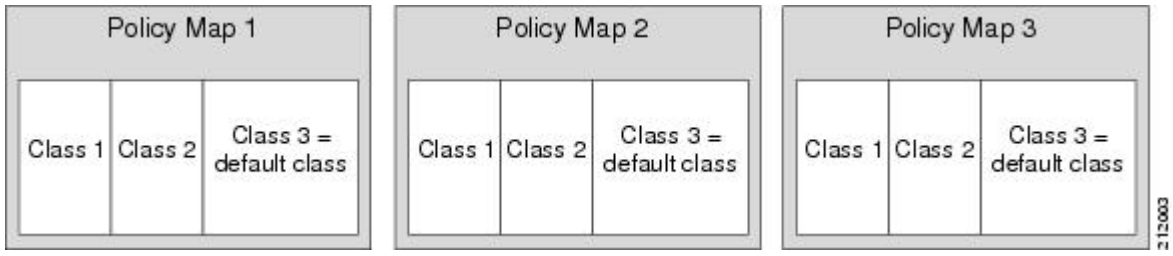
  QoS に関するベスト プラクティスとして、ポートチャネル集約を使用することをお勧めします（「集約 EtherChannel QoS 」の章を参照）。

クラス定義文内のフラグメントについて

QoS ポリシー集約機能は、クラス定義文にフラグメントという発想を取り入れています。ポリシー マップ中で、デフォルト トラフィック クラス定義文をフラグメントとしてマークできます。同じインターフェイス上の他のポリシー マップでも、必要に応じてそのデフォルト トラフィック クラス文をフラグメントとして定義できます。その後、サービス フラグメント クラス定義文を使用して個別のポリシーマップを作成し、すべてのフラグメントに 1 つのグループとして QoS を割り当てるために使用できます。

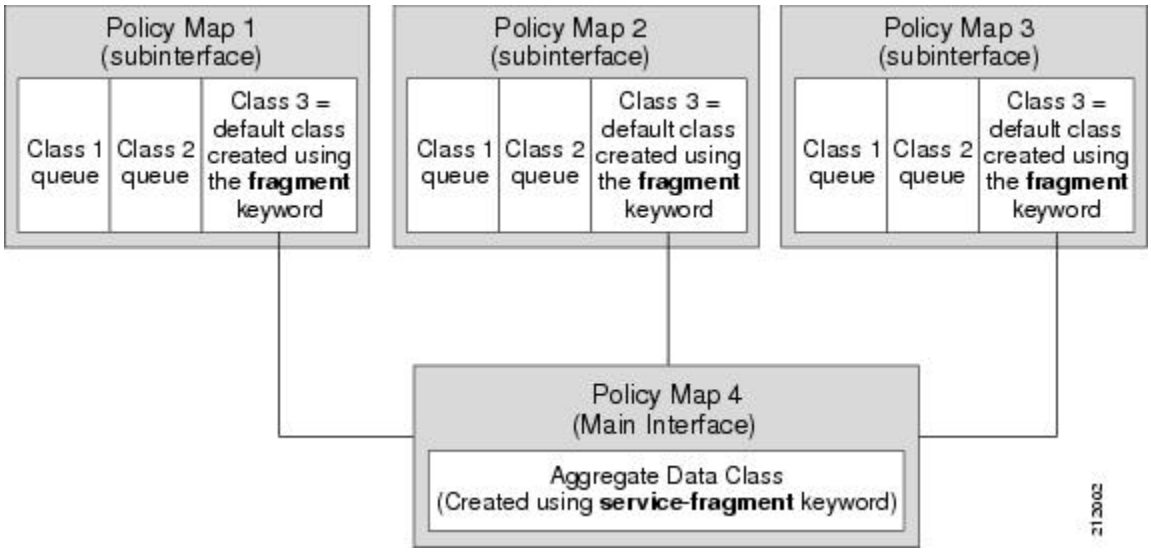
次の図に、フラグメントを使用していない 3 つのポリシーマップが適用された 1 つの物理インターフェイスの例を示します。各ポリシー マップには、独自のポリシー マップ内でデフォルト トラフィックのトラフィックのみを分類できるデフォルトのトラフィック クラスがあることに注意してください。

図 *61* : ポリシー マップを持つ物理インターフェイス：フラグメントの使用なし



次の図にフラグメントを使用して設定した同じ設定を示し、フラグメントをまとめて分類するクラス定義文を使用して 4 つ目のポリシー マップを追加します。デフォルト トラフィック クラスは、個々のポリシー マップ内の 3 つの個別のデフォルト トラフィック クラスとしてではなく、1 つのサービス フラグメント グループとして分類されます。

図 *62 :* ポリシー マップを持つ物理インターフェイス：フラグメントを使用



Gigabit EtherChannel バンドルのフラグメント

Gigabit Etherchannel バンドルに対してフラグメントを設定すると、**fragment** キーワードを使用して設定したデフォルトのトラフィック クラスを持つポリシーマップがメンバー サブインターフェイスのリンクに適用され、フラグメントをまとめて分類するように **service-fragment** キーワードを使用して設定したトラフィック クラスを持つポリシーマップが物理インターフェイスに適用されます。

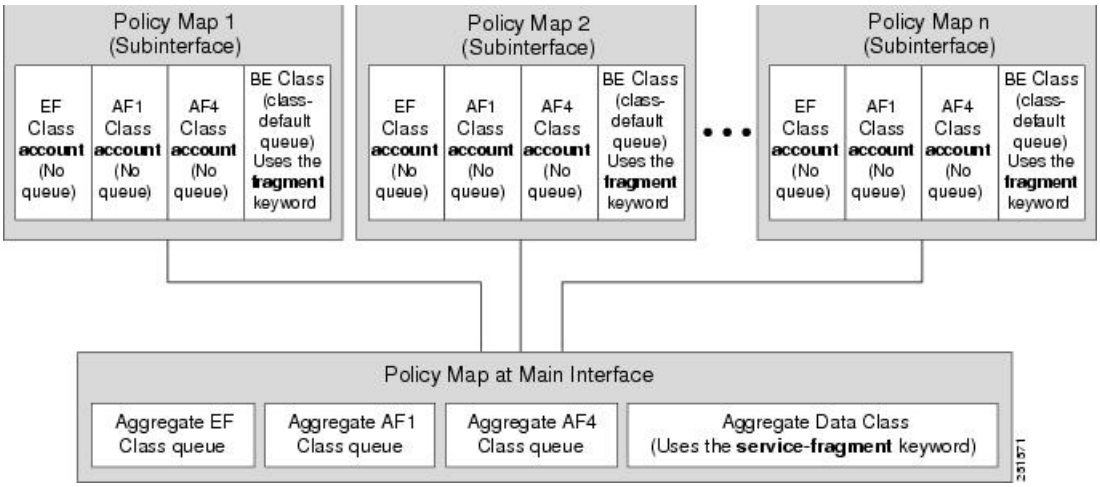
特定のポートチャネルのメンバー リンクで現在アクティブなフラグメントを使用して設定したすべてのポートチャネル サブインターフェイスは、そのメンバー リンクに集約サービス フラグメント クラスを使用します。メンバー リンクがダウンすると、セカンダリ メンバー リンクに切り替わる必要があるポートチャネルのサブインターフェイスは、新しいインターフェイスに集約サービス フラグメントを使用します。

QoS：ポリシー集約 MQC

メイン インターフェイスでの複数キュー集約に対する QoS のポリシー集約 MQC サポート機能は、**fragment** の設定と **service-fragment** の設定を使用した class-default トラフィックの以前の集約のサポートを、次の図に示すように、メインインターフェイスのポリシーマップで集約された DSCP ベースのトラフィック クラスなどのサブインターフェイスのポリシーマップ内のユーザ定義の他のトラフィック クラスに拡張します。

キューイングがサブインターフェイスのポリシーマップのトラフィック クラスに設定されていない場合は、**account** コマンドを使用して、これらのクラスの集約レベルで行われるキューイングのドロップを追跡でき、**show policy-map interface** コマンドを使用して表示できます。

図 *63 :* メイン インターフェイスでの複数のキュー集約に対する *MQC* サポート機能のポリシーマップの概要



元の機能と複数キューの集約に対するMQCサポートとの違いポリシー集約の違い：サブインターフェイスでの出力 MQC キューイングとメイン インターフェイスでの複数キューの集約に対する MQC サポート

「ポリシー集約：サブインターフェイスでの出力 MQC キューイング」のシナリオと、「メイン インターフェイスでの複数キューの集約に対する MQC サポート：メイン インターフェイスでの出力 MQC キューイング」シナリオ間での設定の一部は同じように見えますが、キューイング動作と内部のデータ処理に重要な違いがあります。「QoS の概要：ポリシーアグリゲーション MQC」の項の図を参照してください。

たとえば、どちらの設定も、サブスクライバ ポリシー マップの **class class-default** コマンドの **fragment** キーワードを共有し、これを使用する必要があるだけでなく、集約トラフィックに対して共通のポリシー処理を実現するために、メインインターフェイスのポリシーマップのユーザ定義クラスに **service-fragment** キーワードを設定する必要があります。ただし、この設定を使用すると、元の QoS ポリシー集約の実装と、強化された QoS ポリシー集約の実装とでは動作に違いが生じます。

- フラグメントとサービス フラグメント アーキテクチャを使用した元の実装では、すべてのデフォルトクラスのトラフィックと、サブインターフェイスで定義されたキューイング機能を備えていないクラスのすべてのトラフィックが class-default キューに移動し、メイン ポリシー マップで定義されている共通のユーザ定義のキューとポリシーに集約されます。サブインターフェイスの（同じ物理インターフェイス上の複数のサブスクライバからなどの）トラフィック集約は、最終的には、デフォルトクラスである 1 つのクラスに対してのみ発生します。

- フラグメントとサービス フラグメント機能を使用するメイン インターフェイスでの複数キューの集約に対する MQC サポート機能が強化された実装でも、デフォルトのすべてのクラスのトラフィックも class-default キューに移動し、メイン ポリシー マップで定義された共通のユーザ定義キューとポリシーに集約されます。ただし、集約ポリシーについて

は、DSCPベースのサブスクライバ トラフィック クラスなどの他のクラスもサポートされています。これらのトラフィック クラスは、サブスクライバ ポリシー マップの **account** 以外のキューやキューイング機能をサポートしていません。フラグメントとサービス フラグメント アーキテクチャを使用することによって、（同じ物理インターフェイス上の複数のサブスクライバからの）これらの他のサブスクライバ トラフィック クラスがメイン ポリシーマップでこれらの同じクラスに対して定義されている集約トラフィックに対して共通のポリシー処理が実現します。

EtherChannel 用の QoS の設定方法

ポートチャネルのサブインターフェイスでの出力 MQC キューイングの設定

始める前に

class-map コマンドを使用してトラフィック クラスが設定されている必要があります。以前に定義したクラス マップを使用して、1 レベルまたは 2 レベルの階層型ポリシーマップを設定する必要があります。EtherChannel 上の選択したプライマリおよびセカンダリの物理インターフェイスに一致するように、適切なカプセル化サブコマンドを使用して、ポートチャネルのサブインターフェイスが設定されている必要があります。Cisco IOS XE リリース 2.1 以降のソフトウェアが必要です。グローバル コンフィギュレーションに **port-channel load-balancing vlan-manual** コマンドが含まれているか、ポートチャネルのメインインターフェイス設定に **load-balancing vlan** コマンドが含まれている必要があります。これらのコマンドがすでに実行済みであると想定しています。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface port-channel** *port-channel-number* **.** *subinterface-number*
4. **service-policy output** *policy-map-name*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface port-channel** *port-channel-number . subinterface-number* 例： `Device(config)# interface port-channel 1.200`	サービス ポリシー設定を受け取るポートチャネルのサブインターフェイスを指定します。

ステップ 4	**service-policy output** *policy-map-name*
	例：
	`Device(config-subif)#`
	`service-policy output WAN-GEC-sub-Out`
		出力トラフィックに適用するサービス ポリシーの名前を指定します。

ステップ 5	**end** 例： `Device(config-subif)# end`	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

ポートチャネルのメンバー リンクでの出力 MQC キューイングの設定

始める前に

class-map コマンドを使用してトラフィック クラスが設定されている必要があります。以前に定義したクラス マップを使用して、キューイング機能を使用する 1 レベルまたは 2 レベルの階層型ポリシーマップを設定する必要があります。EtherChannel のメンバー リンク インターフェイスがチャネル グループ（EtherChannel グループ）の一部となるようにすでに設定されている必要があります。キューイング コマンドが含まれているポリシー マップを、ポートチャネルのサブインターフェイス上に設定する必要があります。Cisco IOS XE リリース 2.1 以降のソフトウェアが必要です。グローバル コンフィギュレーションに **port-channel load-balancing vlan-manual** コマンドが含まれているか、ポートチャネルのメインインターフェイス設定に **load-balancing vlan** コマンドが含まれている必要があります。これらのコマンドがすでに実行済みであると想定しています。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface GigabitEthernet** *card/bay/port*
4. **service-policy output** *policy-map-name*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface GigabitEthernet** *card/bay/port* 例： `Device(config)# interface GigabitEthernet 0/1/0`	サービス ポリシー設定を受け取るメンバー リンクの物理インターフェイスを指定します。

ステップ 4	**service-policy output** *policy-map-name*
	例：
	`Device(config-if)#`
	`service-policy output WAN-GEC-sub-Out`
		EtherChannel の一部であるこの物理インターフェイスの出力トラフィックに適用するサービス ポリシーの名前を指定します。

ステップ 5	**end** 例： `Device(config-if)# end`	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

QoS のポリシー集約の設定：サブインターフェイスでの出力 MQC キューイング

始める前に

fragment キーワードをサブインターフェイス **class class-default** の設定と、**service-fragment** 設定をメインインターフェイス クラスに使用するときに、複数のポートチャネルのサブインターフェイスからのデフォルト クラスのトラフィックをメイン インターフェイスの共通ポリシーマップに集約できます。キューイングは、サブインターフェイスのポリシーマップでキューイング機能を使用して定義した他のトラフィック クラスのサブインターフェイスで実行されます。

この機能は、モジュラ QoS CLI（MQC）を使用して設定されます。これは、同じ物理インターフェイスに適用された複数のポリシーマップを、複数のポートチャネルのサブインターフェイスからの複数のデフォルトのトラフィック クラスを集約して扱う QoS 設定で最も有効です。Cisco IOS XE リリース 2.1 以降のソフトウェアが必要です。グローバル コンフィギュレーショ

ンに **port-channel load-balancing vlan-manual** コマンドが含まれているか、ポートチャネルのメインインターフェイスに **load-balancing vlan** コマンドが存在している必要があります。これらのコマンドがすでに実行済みであると想定しています。

（注） この機能は、ポリシーマップが複数のポートチャネルのサブインターフェイスとポートチャネルのメンバーリンクのインターフェイスに適用されている場合にサポートされます。この機能を使用して、異なる物理インターフェイス上のポリシー マップのデフォルトのトラフィック クラスをまとめて分類することはできません。ポートチャネルのメンバー リンクがサブインターフェイスでの **encapsulation** コマンドで **primary** ディレクティブまたは **secondary** ディレクティブによって指定された場合に、そのリンクに向かうすべてのトラフィックをまとめて分類できます。すべてのサブインターフェイスのトラフィック クラスにキューが必要です。ただし、キューイング機能（**priority**、**shape**、**bandwidth**、**queue-limit**、**fair-queue**、**random-detect** などのコマンド）を使用してトラフィック クラスがサブインターフェイスのポリシーマップに設定されていない場合、トラフィックは class-default キューに割り当てられます。**fragment** と **service-fragment** の設定を使用しないサブインターフェイスのトラフィッククラスに対しては、メインインターフェイスのポリシーマップでの分類は行われず、サポートもされていません。

QoS ポリシー集約機能を完全に設定するには、多段階のプロセスが関わっています。以降の項ではこれらのステップについて詳しく説明します。

ポリシー マップの適用と削除については、次の点に注意してください。

- QoS ポリシー集約を設定するには、**service-fragment** キーワードを含むポリシー マップをメイン インターフェイスに最初に適用してから、**fragment** キーワードを含むポリシーマップをサブインターフェイスに適用する必要があります。
- QoS：ポリシー集約をディセーブルにするには、まず fragment キーワードを含んでいるポリシー マップをサブインターフェイスから削除し、次に service-fragment キーワードを含んでいるポリシー マップをメイン インターフェイスから削除する必要があります。

ポリシーマップでのフラグメント トラフィック クラスの設定

始める前に

この手順では、ポリシーマップ内にデフォルト トラフィック クラスをフラグメントとして設定する方法のみを示します。ポリシーマップ内の他のクラス、またはデバイス上の他のポリシーマップを設定する手順は含まれません。

例

（注） この例では、Cisco IOS XE リリース 2.6 よりも前のリリースでサポートされている設定例を示します。

次の例では、BestEffort というフラグメントがポリシーマップ subscriber1 とポリシーマップ subscriber 2 に作成されます。この例では、他のトラフィック クラスのキューイング 機能はサブインターフェイスのポリシーマップでサポートされます。

```
policy-map subscriber1
 class voice
  set cos 5
  priority level 1
 class video
  set cos 4
  priority level 2
 class class-default fragment BestEffort
  shape average 200000000
  bandwidth remaining ratio 10
policy-map subscriber 2
 class voice
  set cos 5
  priority level 1
 class video
  set cos 4
  priority level 2
 class class-default fragment BestEffort
  shape average 200000000
  bandwidth remaining ratio 10
```

（注） この例では、Cisco IOS XE リリース 2.6 以降のリリースでサポートされている設定を示します。

また、次に、メインインターフェイスの実装で、複数キューの集約に対する QoS ポリシー集約 MQC サポートを使用して、サブインターフェイスのポリシーマップにデフォルト クラスとして BestEffort というフラグメントを設定する例を示します。この例では、ポリシーマップの他のクラスに対してキューイング機能がサポートされていないことに注意してください。

```
policy-map subscriber1
 class voice
  set cos 5
  account
 class video
  set cos 4
  account
 class AF1
  account
 class class-default fragment BestEffort
  shape average 200000000
  bandwidth remaining ratio 10
```

default class ステートメントを複数のサブインターフェイス ポリシーマップにフラグメントとして設定した後、service-fragment キーワードを使用した class ステートメントを持つ個別のポリシーマップを設定し、フラグメントとして設定した class ステートメントに QoS を適用する必要があります。

次の作業

デフォルトの複数の class ステートメントをポリシーマップのフラグメントとして設定した後に、**service-fragment** キーワードを使用して class ステートメントの個別のポリシーマップを設定し、フラグメントとして設定した class ステートメントに QoS を適用する必要があります。

このプロセスについては、「サービス フラグメント トラフィック クラスの設定」の項を参照してください。

サービス フラグメントのトラフィック クラスの設定

始める前に

次に、ポリシーマップ内にサービス フラグメント トラフィックの class ステートメントを設定するタスクを示します。サービスフラグメントトラフィック クラスを使用して、他のポリシーマップを以前にフラグメントとして設定した default class ステートメントのコレクションに QoS を適用します。

この手順では、フラグメントのデフォルト トラフィック クラスがすでに作成されていると想定しています。フラグメントのデフォルト トラフィック クラスの作成手順については、「ポリシーマップでのフラグメント トラフィック クラスの設定」の項を参照してください。

すべてのポリシーマップと同様に、設定はインターフェイスに適用されるまで、ネットワークトラフィックを管理しません。この手順では、インターフェイスへのポリシーマップの適用プロセスは扱いません。

（注） サービスフラグメントを使用すると、同じ物理インターフェイスからのフラグメントのみをまとめて分類できます。同じサービスフラグメントを使用して、別のインターフェイスからのフラグメントを分類することはできません。

service-fragment キーワードが入力されているクラスではキューイング機能だけが許可され、**service-fragment** キーワードが使用されているクラスでは 1 つ以上のキューイング機能を入力する必要があります。

service-fragment キーワードを使用したクラスがあるポリシーマップは、インターフェイスから出て行くトラフィックのみに適用できます（**service-policy output** コマンドを使用してインターフェイスに適用したポリシーマップ）。

service-fragment キーワードを使用して設定したクラスは、インターフェイス上にまだ設定されているフラグメントに QoS をまとめて適用するために使用中である場合は削除できません。**service-fragment** キーワードを使用して設定したクラスを削除するには、サービス フラグメントを削除する前にフラグメント トラフィックのクラスを削除します。

service-fragment キーワードは、子ポリシーマップには入力できません。

手順の概要

1. **enable**

2. **configure terminal**
3. **policy-map** *policy-map-name*
4. **class** *class-name* **service-fragment** *fragment-class-name*
5. **shape average percent** *percent*
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**policy-map** *policy-map-name* 例： `Device(config)# policy-map BestEffortFragments`	設定するトラフィック ポリシーの名前を指定し、ポリシーマップ コンフィギュレーション モードを開始します。
ステップ 4	**class** *class-name* **service-fragment** *fragment-class-name* 例： `Device(config-pmap)# class data service-fragment BestEffort`	*fragment-class-name* に一致するすべてのフラグメントを合わせたトラフィックのクラスを指定します。他のポリシーマップでフラグメントを定義する場合、サービス フラグメント クラスを正しく設定するには、*fragment-class-name* がこのコマンド ラインの *fragment-class-name* に一致する必要があります。
ステップ 5	**shape average percent** *percent* 例： `Device(config-pmap-c)# shape average percent 50`	QoS コンフィギュレーション コマンドを入力します。フラグメントとして設定されたデフォルトのトラフィック クラスでは、キューイング機能のみがサポートされます。 サポートされているキューイング機能は**bandwidth**、**shape**、**random-detect exponential-weighting-constant** です。 複数の QoS キューイング コマンドを入力できます。
ステップ 6	**end** 例： `Device(config-pmap-c)# end`	ポリシー マップ クラス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

例

（注） この例では、Cisco IOS XE リリース 2.6 よりも前のリリースでサポートされている設定例を示します。

次の例では、BestEffort という名前のすべてのフラグメントに QoS を適用するために、ポリシーマップが作成されます。

```
policy-map main-interface
 class data service-fragment BestEffort
  shape average 400000000
```

次の例では、2 つのフラグメントを作成し、サービス フラグメントを使用してまとめて分類します。

```
policy-map subscriber1
 class voice
  set cos 5
  priority level 1
 class video
  set cos 4
  priority level 2
 class class-default fragment BestEffort
  shape average 200000000
  bandwidth remaining ratio 10
policy-map subscriber 2
 class voice
  set cos 5
  priority level 1
 class video
  set cos 4
  priority level 2
 class class-default fragment BestEffort
  shape average 200000000
  bandwidth remaining ratio 10
```

（注） この例では、Cisco IOS XE リリース 2.6 以降のリリースでサポートされている設定を示します。

次に、サブインターフェイス ポリシーマップに BestEffort という 2 つのフラグメントを作成し、BestEffort という **service-fragment** を設定してメイン インターフェイスのポリシーマップでキューを集約する例を示します。

```
policy-map subscriber1
 class voice
  set cos 5
  account
 class video
  set cos 4
  account
```

```
 class AF1
  account
 class class-default fragment BestEffort
  shape average 200000000
  bandwidth remaining ratio 10
policy-map subscriber2
 class voice
  set cos 5
  account
 class video
  set cos 4
  account
 class AF1
  account
 class class-default fragment BestEffort
  shape average 200000000
  bandwidth remaining ratio 10
policy-map main-interface
 class voice
  priority level 1
 class video
  priority level 2
 class AF1
  bandwidth remaining ratio 90
 class data service-fragment BestEffort
  shape average 400000000
  bandwidth remaining ratio 1
```

トラブルシューティングのヒント

同じサービス フラグメントの一部であるすべての class ステートメントが同じフラグメント クラス名を共有していることを確認します。

次の作業

サービス フラグメント トラフィック クラスをメインの物理インターフェイスに適用します。

フラグメント トラフィック クラスをメンバーリンクのサブインターフェイスに適用します。

Gigabit EtherChannel バンドルをサポートする物理インターフェイスでのサービス フラグメントの設定

始める前に

この手順では、サービス フラグメント トラフィック クラスがすでに作成されている必要があります。フラグメント クラスを設定しないと、サービス フラグメントのトラフィック クラスは設定できません。フラグメントのクラスを作成する手順については、「ポリシーマップでのフラグメントのトラフィック クラスの設定」の項を参照してください。サービス フラグメントのクラスを作成する手順については、「サービス フラグメントのトラフィック クラスの設定」の項を参照してください。

これらの手順には、Gigabit EtherChannel のメンバー リンクのサブインターフェイスに設定可能なオプションに関する詳細は示されていません。これらの手順では、フラグメントのトラフィック クラスをすでに持っているポリシーマップをメンバー リンクのサブインターフェイスに適用する手順のみが説明されています。

（注） 正しく動作させるには、ポートチャネルのメンバーリンクがダウンしたときに、すべてのメンバー リンクに同じポリシーマップが適用されている必要があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface GigabitEthernet** *card/bay/port*
4. **service-policy output** *service-fragment-class-name*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface GigabitEthernet** *card/bay/port* 例： `Device(config)# interface GigabitEthernet 0/1/0`	サービス ポリシー設定を受け取るメンバー リンクの物理インターフェイスを指定します。
ステップ 4	**service-policy output** *service-fragment-class-name* 例： `Device(config-if)# service-policy output aggregate-member-link`	サービス フラグメントのデフォルトのトラフィック クラスを含むサービス ポリシーを物理ギガビット イーサネット インターフェイスに適用します。
ステップ 5	**end** 例： `Device(config-if)# end`	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

例

次に、ポリシーマップの aggregate-member-link を物理インターフェイスに適用する例を示します。

```
interface GigabitEthernet1/1/1
 service-policy output aggregate-member-link
!
interface GigabitEthernet1/1/2
 service-policy output aggregate-member-link
```

次のタスク

service-fragment の定義と fragment-class の定義で、フラグメント クラス名が整合していることを確認します。「Gigabit EtherChannel メンバー リンクのサブインターフェイスでのフラグメントの設定」の項に進みます。

Gigabit EtherChannel メンバー リンクのサブインターフェイスでのフラグメントの設定

始める前に

この手順では、サービス フラグメント トラフィック クラスがすでに作成されている必要があります。フラグメント クラスを設定しないと、サービス フラグメントのトラフィック クラスは設定できません。フラグメントのクラスを作成する手順については、「ポリシーマップでのフラグメントのトラフィック クラスの設定」の項を参照してください。サービス フラグメントのクラスを作成する手順については、「サービス フラグメントのトラフィック クラスの設定」の項を参照してください。

これらの手順には、Gigabit EtherChannel のメンバー リンクのサブインターフェイスに設定可能なオプションに関する詳細は示されていません。これらの手順では、フラグメントのトラフィック クラスをすでに持っているポリシー マップをメンバー リンクのサブインターフェイスに適用する手順のみが説明されています。

フラグメントは、2 つ以上の物理インターフェイスのトラフィックに使用できません。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface port-channel** *port-channel-interface-number* **.** *port-channel-subinterface-number*
4. **service-policy output** *fragment-class-name*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。

	コマンドまたはアクション	目的
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface port-channel** *port-channel-interface-number* **.** *port-channel-subinterface-number* 例： `Device(config)# interface port-channel 1.100`	サブインターフェイス コンフィギュレーション モードを開始し、EtherChannel メンバー リンクのサブインターフェイスを設定します。
ステップ 4	**service-policy output** *fragment-class-name* 例： `Device(config-subif)# service-policy output subscriber`	EtherChannel メンバー リンクのサブインターフェイスに対し、フラグメントのデフォルトのトラフィック クラスが含まれているサービス ポリシーを適用します。
ステップ 5	**end** 例： `Device(config-subif)# end`	サブインターフェイス コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。

例

次の例では、subscriber というサービス ポリシーにフラグメントのデフォルトのトラフィック クラスがあり、このサービス ポリシーが EtherChannel バンドルのポートチャネルのサブインターフェイスに適用されます。

```
interface port-channel 1.100
 service-policy output subscriber
```

ポートチャネルサブインターフェイスでの入力ポリシングとマーキングの設定

始める前に

class-map コマンドを使用してトラフィック クラスが設定されている必要があります。以前に定義したクラス マップを使用して、1 レベルまたは 2 レベルの階層型ポリシーマップを設定する必要があります。EtherChannel のメンバー リンク インターフェイスがチャネル グループ（EtherChannel グループ）の一部となるようにすでに設定されている必要があります。Cisco IOS XE リリース 2.1 以降のソフトウェアが必要です。グローバル コンフィギュレーションに **port-channel load-balancing vlan-manual** コマンドが含まれているか、ポートチャネルのメイン

インターフェイス設定に **load-balancing vlan** コマンドが含まれている必要があります。これらのコマンドがすでに実行済みであると想定しています。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface port-channel** *port-channel-number . port-channel-interface-number . sub-interface-number*
4. **service-policy input** *policy-map-name*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface port-channel** *port-channel-number . port-channel-interface-number . sub-interface-number* 例： `Device(config)# interface port-channel 1.100.100`	サブインターフェイス コンフィギュレーション モードを開始し、EtherChannel メンバー リンクのサブインターフェイスを設定します。
ステップ 4	**service-policy input** *policy-map-name* 例： `Device(config-subif)# service-policy input sub-intf-input`	以前に指定したポートチャネルのサブインターフェイスの入力トラフィックに適用するサービス ポリシーの名前を指定します。
ステップ 5	**end** 例： `Device(config-subif)# end`	サブインターフェイス コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。

例

次の例では、sub-intf-input という名前のサービス ポリシーを定義し、そのポリシーをポートチャネルのサブインターフェイスの入力方向に適用します。

```
policy-map sub-intf-input
  class voice
    set precedence 5
  class video
    set precedence 6
  class class-default
    set precedence 3
!
interface Port-channel 1.100
 service-policy input sub-intf-input
```

ポートチャネルのメンバーリンクでの出力ポリシングとマーキングの設定

始める前に

class-map コマンドを使用してトラフィック クラスが設定されている必要があります。以前に定義したクラス マップを使用して、1 レベルまたは 2 レベルの階層型ポリシーマップを設定する必要があります。EtherChannel のメンバー リンク インターフェイスがチャネル グループ（EtherChannel グループ）の一部となるようにすでに設定されている必要があります。Cisco IOS XE リリース 2.1 以降のソフトウェアが必要です。グローバル コンフィギュレーションに **port-channel load-balancing vlan-manual** コマンドが含まれているか、ポートチャネルのメインインターフェイス設定に **load-balancing vlan** コマンドが含まれている必要があります。これらのコマンドがすでに実行済みであると想定しています。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface port-channel** *port-channel-number . port-channel-interface-number . sub-interface-number*
4. **service-policy output** *policy-map-name*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**interface port-channel** *port-channel-number.port-channel-interface-number.sub-interface-number* 例： `Device(config)# interface port-channel 1.100.100`	サブインターフェイス コンフィギュレーション モードを開始し、EtherChannel メンバー リンクのサブインターフェイスを設定します。
ステップ 4	**service-policy output** *policy-map-name* 例： `Device(config-subif)# service-policy output WAN-GEC-member-Out-police`	前のステップで指定した EtherChannel メンバー リンクのサブインターフェイスの出力トラフィックに適用するサービス ポリシーの名前を指定します。
ステップ 5	**end** 例： `Device(config-subif)# end`	サブインターフェイス コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。

例

次の例では、WAN-GEC-member-Out-police という名前のサービス ポリシーを定義し、ポートチャネルのサブインターフェイスの出力方向に適用します。

```
policy-map WAN-GEC-member-Out-police
  class voice
    set precedence 5
  class video
    set precedence 6
  class class-default
    set precedence 3
!
interface port-channel 1.100
 service-policy output WAN-GEC-member-Out-police
```

ポリシー集約の設定：メインインターフェイスでの複数キュー集約に対する MQC サポート

始める前に

この機能は MQC を使用して設定されます。これは、同じ物理インターフェイスに適用された複数のポリシーマップを、複数のポートチャネルのサブインターフェイスからの複数のユーザ定義のトラフィック クラスを集約して扱う QoS 設定で最も有効です。Cisco IOS XE リリース 2.6 以降のソフトウェアが必要です。グローバル コンフィギュレーションには **port-channel load-balancing vlan-manual** コマンドが含まれている必要があります。または、設定するポート

チャネルのメイン インターフェイスに **port-channel load-balancing vlan** コマンドが存在している必要があります。これらのコマンドがすでに実行済みであると想定しています。

この機能は、ポリシーマップが複数のポートチャネルのサブインターフェイスとポートチャネルのメンバーリンクのインターフェイスに適用されている場合にサポートされます。この機能を使用して、異なる物理インターフェイス上のポリシー マップのデフォルトのトラフィック クラスをまとめて分類することはできません。ポートチャネルのメンバー リンクがサブインターフェイスでの **encapsulation** コマンドで **primary** ディレクティブまたは **secondary** ディレクティブによって指定された場合に、そのリンクに向かうすべてのトラフィックをまとめて分類できます。次の項で、EtherChannel によるこのタイプの QoS ポリシー集約を設定する際の動作と制約事項を説明します。

- 設定されたキューイング機能がないサブインターフェイスのトラフィック クラスには、サブスクライバ レベルのキューがありません。
- **fragment** キーワードをサブインターフェイス **class class-default** の設定と、**service-fragment** 設定をメイン インターフェイス クラスで使用するときに、複数のサブインターフェイスからのデフォルト クラスのトラフィックをメイン インターフェイスの共通ポリシー マップに集約できます。
- この設定ではさらに、メインインターフェイスの共通ポリシーマップに集約する他のサブインターフェイスのトラフィック クラス（DSCP ベースのクラスなど）に対するサポートをイネーブルにします。
- **fragment** キーワードをサブインターフェイスの **class-default** クラスに使用し、**service-fragment** 設定をメイン インターフェイス クラスに使用することで、この機能がイネーブルになります（これは、デフォルト クラスの集約もイネーブルにします）。
- キューイング機能は、他のトラフィック クラスのサブインターフェイスのポリシー マップでは設定されません。
- 他のサブインターフェイスのトラフィック クラスについては、メインインターフェイスのポリシー マップで集約としてキューイングが実行されます。
- 統計情報の任意のトラッキングをサポートするには、サブインターフェイスのポリシー マップに **account**account**account** コマンドを使用します。

メイン インターフェイでの QoS の複数キュー集約機能を完全に設定するには、次に示すように多段階のプロセスが関わっています。

1. 「ポリシー マップでのフラグメント トラフィック クラスの設定」の項で説明した複数のサブインターフェイスのポリシー マップ内にフラグメントとして default class ステートメントを設定します。
2. 「サービス フラグメントのトラフィック クラスの設定」の項で説明したように、フラグメントとして設定した class ステートメントに QoS を適用するために、**service-fragment** キーワードを使用して、class ステートメントで個別のポリシー マップを設定します。

3. 「Gigabit EtherChannel バンドルをサポートする物理インターフェイスでのサービス フラグメントの設定」の項で説明したように、サービス フラグメントのトラフィック クラスを設定し、それらをメインの物理インターフェイスに適用します。
4. 「Gigabit EtherChannel メンバー リンクのサブインターフェイスでのフラグメントの設定」の項で説明したように、フラグメント トラフィックのクラスを設定し、それらをメンバー リンクのサブインターフェイスに適用します。

ポートチャネルのメンバー リンクでの MQC キューイング設定：EtherChannel ロード バランシングなし

始める前に

class-map コマンドを使用してトラフィック クラスが設定されている必要があります。以前に定義したクラス マップを使用して、1 レベルまたは 2 レベルの階層型ポリシーマップを設定する必要があります。

Cisco IOS XE リリース 2.4 以降のソフトウェアが必要です。

また、ポートチャネル メイン インターフェイスにアクティブ/スタンバイ シナリオを作成する次のコマンドが含まれている必要があります。このような設定では、1 つのインターフェイスのみをアクティブにすることで、トラフィックをいつでも転送できます。

- **interface Port-channel1**
- **lacp fast-switchover**
- **lacp max-bundle 1**

手順の概要

1. **enable**
2. **configure terminal**
3. **interface GigabitEthernet** *card/bay/port*
4. **service-policy output** *policy-map-name*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例：	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
	`Device# configure terminal`	
ステップ 3	**interface GigabitEthernet** *card/bay/port* 例： `Device(config)# interface GigabitEthernet 0/1/0`	サービス ポリシー設定を受け取るメンバー リンクの物理インターフェイスを指定します。

ステップ 4	**service-policy output** *policy-map-name*
	例：
	`Device(config-if)#`
	`service-policy output WAN-GEC-member-Out`
		出力トラフィックに適用するサービス ポリシーの名前を指定します。

ステップ 5	**end** 例： `Device(config-if)# end`	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

例

次に、main-intf というサービス ポリシーを定義して、ポートチャネルのメンバー リンクの出力方向に適用する例を示します。

```
interface Port-channel 1
 lcap fast-switchover
 lacp max-bundle 1
!
policy-map main-intf
 class voice
  priority
  police cir 10000000
 class video
  bandwidth remaining ratio 10
 class class-default
  bandwidth remaining ratio 3
!
interface GigabitEthernet0/0/0
 channel-group 1 mode active
 service-policy output main-intf
!
interface GigabitEthernet0/0/1
 channel-group 1 mode active
 service-policy output main-intf
```

ポートチャネルのメンバーリンクでのMQCキューイング設定の設定：EtherChannel ロード バランシング

始める前に

class-map コマンドを使用してトラフィック クラスが設定されている必要があります。以前に定義したクラス マップを使用して、1 レベルまたは 2 レベルの階層型ポリシーマップを設定する必要があります。EtherChannel 上の選択したプライマリおよびセカンダリの物理インターフェイスに一致するように、適切なカプセル化サブコマンドを使用して、ポートチャネルのサブインターフェイスが設定されている必要があります。Cisco IOS XE リリース 2.5 以降のソフトウェアが必要です。

EtherChannel の設定には、フローベースのロードバランシングをイネーブルにした複数のアクティブなインターフェイスがある場合があります。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface GigabitEthernet** *card/bay/port*
4. **service-policy output** *policy-map-name*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface GigabitEthernet** *card/bay/port* 例： `Device(config)# interface GigabitEthernet 0/1/0`	サービス ポリシー設定を受け取るメンバー リンクの物理インターフェイスを指定します。

ステップ 4	**service-policy output** *policy-map-name*
	例：
	`Device(config-if)#`
	`service-policy output WAN-GEC-member-Out`
		出力トラフィックに適用するサービス ポリシーの名前を指定します。

	コマンドまたはアクション	目的
ステップ 5	**end** 例： `Device(config-if)# end`	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

例

次に、main-intf というサービス ポリシーを定義して、ポートチャネルのメンバー リンクの出力方向に適用する例を示します。

```
  class voice
   priority
   police cir 10000000
  class video
   bandwidth remaining ratio 10
  class class-default
   bandwidth remaining ratio 3
!
interface GigabitEthernet0/0/0
 channel-group 1 mode active
 service-policy output main-intf
!
interface GigabitEthernet0/0/1
 channel-group 1 mode active
 service-policy output main-intf
```

EtherChannels の QoS の設定例

例：QoS ポリシー集約の設定：サブインターフェイスでの出力 MQC キューイング

```
port-channel load-balancing vlan-manual
!
class-map match-all BestEffort
!
class-map match-all video
 match precedence 4
!
class-map match-all voice
 match precedence 5
!
policy-map subscriber
 class voice
  priority level 1
 class video
  priority level 2
 class class-default fragment BE
```

```
  shape average 100000000
  bandwidth remaining ratios 80

policy-map aggregate-member-link
 class BestEffort service-fragment BE
 shape average 100000000
!
interface Port-channel1
 ip address 209.165.200.225 255.255.0.0
!
interface Port-channel1.100
 encapsulation dot1Q 100
 ip address 209.165.200.226 255.255.255.0
 service-policy output subscriber
!
interface Port-channel1.200
 encapsulation dot1Q 200
 ip address 209.165.200.227 255.255.255.0
 service-policy output subscriber
!
interface Port-channel1.300
 encapsulation dot1Q 300
 ip address 209.165.200.228 255.255.255.0
 service-policy output subscriber
!
interface GigabitEthernet1/1/1
 no ip address
 channel-group 1 mode on
 service-policy output aggregate-member-link
!
interface GigabitEthernet1/1/2
 no ip address
 channel-group 1 mode on
 service-policy output aggregate-member-link
```

例：QoS ポリシー集約の設定：メインインターフェイスでの複数キュー集約に対する MQC サポート

```
port-channel load-balancing vlan-manual
!
policy-map subscriber1
 class voice
  set cos 5
  account
 class video
  set cos 4
  account
 class AF1
  account
 class class-default fragment BestEffort
  shape average 200000000
  bandwidth remaining ratio 10
!
policy-map subscriber2
 class voice
  set cos 2
  account
 class video
  set cos 3
```

```
  account
 class AF1
  account
 class class-default fragment BestEffort
  shape average 200000000
  bandwidth remaining ratio 10
!
policy-map main-interface-out
 class voice
  priority level 1
 class video
  priority level 2
 class AF1
  bandwidth remaining ratio 90
 class data service-fragment BestEffort
  shape average 400000000
  bandwidth remaining ratio 1
!
interface GigabitEthernet1/1/1
 no ip address
 channel-group 1 mode on
 service-policy output main-interface-out
!
interface GigabitEthernet1/1/2
 no ip address
 channel-group 1 mode on
 service-policy output main-interface-out
!
interface Port-channel1.100
 encapsulation dot1Q 100
 ip address 10.0.0.1 255.255.255.0
 service-policy output subscriber1
!
interface Port-channel1.200
 encapsulation dot1Q 200
 ip address 10.0.0.2 255.255.255.0
 service-policy output subscriber2
!
interface Port-channel1.300
 encapsulation dot1Q 300
 ip address 10.0.0.4 255.255.255.0
 service-policy output subscriber2
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『*Cisco IOS Master Command List, All Releases*』
QoS コマンド：コマンド構文の詳細、コマンド モード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『*Cisco IOS Quality of Service Solutions Command Reference*』

関連項目	マニュアル タイトル
モジュラ QoS コマンドライン インターフェイス	「Applying QoS Features Using the MQC」モジュール
RADIUS ベースのポリシングの設定	『*Intelligent Services Gateway Configuration Guide*』
CISCO ASR 1000 シリーズ ソフトウェアの設定	『*Cisco ASR 1000 Series Aggregation Services Routers Software Configuration Guide*』

シスコのテクニカル サポート

説明	リンク
★枠で囲まれた Technical Assistance の場合★右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

EtherChannel インターフェイスの QoS の機能情報

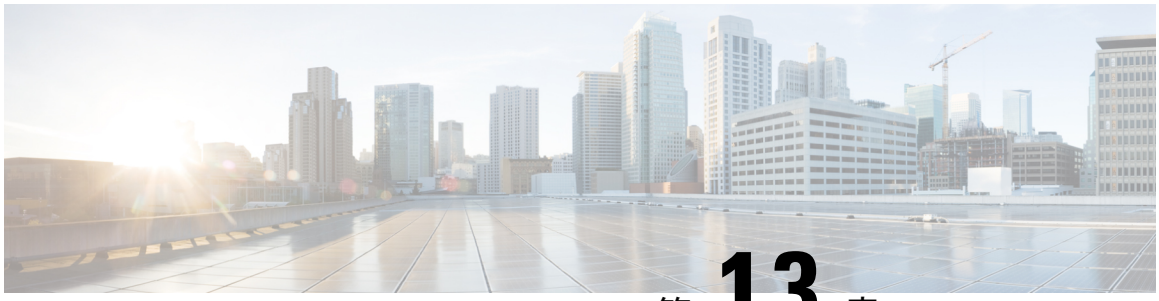
次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *15 : EtherChannel* インターフェイスの *QoS* の機能情報

機能名	リリース	機能情報
ポートチャネル サブインターフェイスでの出力MQCキューイングの設定	Cisco IOS XE リリース 2.1	この機能は、ポートチャネルのサブインターフェイスでの出力 MQC キューの設定をサポートします。 この機能は、Cisco ASR 1000 シリーズ ルータに追加されました。
ポートチャネルのメンバー リンクでの出力MQCキューイングの設定	Cisco IOS XE リリース 2.1	この機能は、ポートチャネルのメンバー リンクで出力MQCキューの設定をサポートします。 この機能は、Cisco ASR 1000 シリーズ ルータに追加されました。
QoS のポリシー集約：サブインターフェイスでの出力MQCキューイング	Cisco IOS XE リリース 2.1	この機能は、 QoS ポリシー集約のサブインターフェイスでの出力MQCの設定をサポートします。 この機能は、Cisco ASR 1000 シリーズ ルータに追加されました。
ポートチャネル サブインターフェイスでの入力ポリシングとマーキング	Cisco IOS XE リリース 2.1	この機能は、ポートチャネルのサブインターフェイスでの入力ポリシングとマーキングの設定をサポートします。 この機能は、Cisco ASR 1000 シリーズ ルータに追加されました。
ポートチャネル メンバー リンクでの出力ポリシングとマーキング	Cisco IOS XE リリース 2.1	この機能は、ポートチャネルのメンバー リンクでの出力MQCポリシングとマーキングの設定をサポートします。 この機能は、Cisco ASR 1000 シリーズ ルータに追加されました。
ポートチャネル メンバー リンクでの出力MQCキューイング設定：EtherChannel ロード バランシングなし	Cisco IOS XE リリース 2.4	この機能は、ポートチャネルのメンバー リンクでの（EtherChannel ロードバランシングなしの）出力MQCキューイングをサポートします。 この機能は、Cisco ASR 1000 シリーズ ルータに追加されました。

機能名	リリース	機能情報
ポートチャネル メンバー リンクでサポートされる出力MQC キューイング設定：EtherChannel ロード バランシング	Cisco IOS XE リリース 2.5	この機能は、ポートチャネルのメンバー リンクでの（EtherChannel ロードバランシングありの）出力MQC キューイングをサポートします。 この機能は、Cisco ASR 1000 シリーズ ルータに追加されました。
QoS ポリシー集約：メイン インターフェイスでの複数キュー集約に対するMQC サポート：メイン インターフェイスでの出力 MQC キューイング	Cisco IOS XE リリース 2.6	この機能は、QoS ポリシー集約、メイン インターフェイスでの複数キュー集約に対するMQC サポート、メインインターフェイスでの出力MQC キューイングの設定をサポートします。 この機能は、Cisco ASR 1000 シリーズ ルータに追加されました。



第 13 章

集約 EtherChannel QoS

集約 EtherChannel の QoS 機能を使用すると、ポートチャネルのメイン インターフェイスまたはサブインターフェイス上に集約出力キューイング ポリシー マップを適用できます。この機能によって、Cisco ASR 1000 シリーズ アグリゲーション サービス ルータに対する集約ポートチャネルのメイン インターフェイス上での QoS サポートが可能になります。

集約 EtherChannel QoS の制約事項

- 集約ポートチャネル インターフェイスを持つイーサネット仮想回線（EVC）上での QoS の設定はサポートされていません。
- 集約ポートチャネル インターフェイス上でのインテリジェント サービス ゲートウェイ（ISG）およびインテリジェント ワイヤレス アクセス ゲートウェイ（iWAG）のコンテキストでの Point-to-Point Protocol over Ethernet（PPPoE）セッションと IP over Ethernet（IPoE）セッションはサポートされていません。
- 集約ポートチャネル インターフェイス上での QoS を備えた仮想プライベート LAN サービス（VPLS）はサポートされていません。
- 集約ポートチャネル インターフェイス上での QoS を備えた Xconnect はサポートされていません。

- モジュラ QoS CLI（MQC）の fragment キーワードと service-fragment キーワードは、集約ポート チャネル インターフェイス タイプと組み合わせて使用できません。
- 集約型ポートチャネル インターフェイスには次の制限があります。
  - ポート チャネルのすべてのメンバー リンクの速度は同じである必要があります。これにより、パケットの再順序付けの可能性を回避します。ギガビットイーサネット、ファスト イーサネット、またはイーサネット インターフェイスを同じポート チャネルに組み合わせることはできません。
  - 10 ギガビット イーサネットは、Cisco IOS XE 3.16.3 以降でサポートされています（Cisco IOS XE 3.17 ではサポートされていません）。10 ギガビット イーサネットは、Cisco IOS XE Denali 16.3 以降でもサポートされています。
- 集約ポートチャネルのメインインターフェイスとポートチャネルのサブインターフェイスの両方に適用された MPOL ポリシーは、どの Cisco IOS XE 3S リリースでもサポートされておらず、Cisco IOS XE Everest 16.5.x 以前でもサポートされていません。
- 集約ポートチャネル サブインターフェイス上の QoS は、Cisco IOS XE 3.16.2 以前ではサポートされていません（また、Cisco IOS XE 3.17 でもサポートされていません）。

集約 EtherChannel QoS に関する情報

集約 EtherChannel QoS のサポート対象機能

集約 EtherChannel QoS 機能は、次をサポートします。

- フローベースのロード バランシング
- 最大 3 レベルの階層
- シェーピング、絶対帯域幅、および相対帯域幅の設定
- サブクラス（VLAN）の最小帯域幅量
- 集約ポートチャネルのメインインターフェイスおよびサブインターフェイス上で同時にイネーブルになる入力 QoS（ポリシングおよびマーキング）と出力 QoS（すべてのキューイング機能）

集約 EtherChannel QoS のサポート対象外機能の組み合わせ

QoS を備えた次のトンネル タイプ インターフェイスの次の組み合わせはサポートされていません。

- 集約キューイングを備えたポート チャネル経由で出力するキューイング ポリシー マップが適用された Generic Routing Encapsulation（GRE）トンネル

- 集約キューイングによってポート チャネル経由で出力するキューイング QoS が適用された静的仮想トンネル インターフェイス（SVTI）および動的仮想トンネル インターフェイス（DVTI）
- サービス グループに属しているサブインターフェイスと、サービスポリシーが適用されたサブインターフェイスは、同じ集約ポートチャネル上では同時に設定できない
- MPOL：集約ポートチャネルのメインインターフェイスとポートチャネルのサブインターフェイスの両方に適用されたポリシー

（注） キューイング QoS のないトンネル（前述）はサポートされていますが、IP アドレスに十分なダイバーシティがない所定の物理インターフェイスをハッシングアルゴリズムが過負荷にするため、推奨しません。

集約 EtherChannel QoS のスケーラビリティ

QoS ポリシーは、次のスケーラビリティの制限に従って、集約ポートチャネル インターフェイスに適用できます。

- 最大 8 個の ポート アドレス
- ポート チャネル内で最大 4 個のメンバー リンク
- メンバー リンクは複数の共有ポート アダプタ（SPA）と SPA インターフェイス プロセッサ（SIP）カードに分割可能

集約 EtherChannel QoS の設定方法

ここでは、Cisco ASR 1000 シリーズ アグリゲーション サービス ルータで集約 EtherChannel QoS を設定する方法について説明します。

手順の概要

1. **enable**
2. **configure terminal**
3. **platform qos port-channel-aggregate** *port-channel-number*
4. **interface port-channel** *port-channel-number*
5. **service-policy** {**output**} *policy-map*
6. **service-policy** {**input**} *policy-map*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**platform qos port-channel-aggregate** *port-channel-number* 例： `router(config)# platform qos port-channel-aggregate 1`	集約ポートチャネルインターフェイスをイネーブルにします。
ステップ 4	**interface port-channel** *port-channel-number* 例： `router(config)# interface port-channel 1`	特定のポートチャネルを設定するインターフェイス コンフィギュレーション モードを開始します。
ステップ 5	**service-policy** {**output**} *policy-map* 例： `router(config-if)# service-policy output egress_policy`	インターフェイスのサービスポリシーとして使用するポリシーマップを出力インターフェイスに付加します。
ステップ 6	**service-policy** {**input**} *policy-map* 例： `router(config-if)# service-policy input ingress_policy`	インターフェイスのサービスポリシーとして使用するポリシーマップを入力インターフェイスに付加します。

集約 EtherChannel QoS の設定解除方法

ここでは、Cisco ASR 1000 シリーズ アグリゲーション サービス ルータで集約 EtherChannel QoS の設定を解除する方法について説明します。

手順の概要

1. **enable**
2. **configure terminal**
3. **no interface port-channel** *port-channel-number*
4. **no platform qos port-channel-aggregate** *port-channel-number*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**no interface port-channel** *port-channel-number* 例： `router(config)# no interface port-channel 1`	特定のポート チャネルの設定を解除します。
ステップ 4	**no platform qos port-channel-aggregate** *port-channel-number* 例： `router(config)# no platform qos port-channel-aggregate 1`	集約ポートチャネルインターフェイスをディセーブルにし、そのインターフェイスにある必要なQoSポリシーを削除します。

集約 EtherChannel QoS の設定例

例：集約ポートチャネル インターフェイスの設定

```
Router# configure terminal
Router(config)# platform qos port-channel-aggregate 1
Router(config)# interface port-channel 1
Router(config-if)# interface GigabitEthernet1/0/1
Router(config-if)# channel-group 1
Router(config-if)# interface GigabitEthernet1/0/0
Router(config-if)# channel-group 1
Router(config-if)# interface port-channel 1.1
Router(config-subif)# encap
Router(config-subif)# encapsulation dot
Router(config-subif)# encapsulation dot1Q 2
Router(config-subif)# ip addr 14.0.1.2 255.255.255.0
Router(config-subif)# interface port-channel 1.2
Router(config-subif)# encapsulation dot1Q 3
Router(config-subif)# ip addr 14.0.2.2 255.255.255.0
Router(config-subif)# interface port-channel 1.3
Router(config-subif)# encapsulation dot1Q 4
Router(config-subif)# ip addr 14.0.3.2 255.255.255.0
Router(config-subif)# end
```

例：QoS に対するクラス マップの設定

```
Router# configure terminal
Router(config)# class-map vlan_2
Router(config-cmap)# match vlan  2
Router(config-cmap)# class-map vlan_3
Router(config-cmap)# match vlan  3
Router(config-cmap)# class-map vlan_4
Router(config-cmap)# match vlan  4
Router(config-cmap)# class-map prec1
Router(config-cmap)# match precedence 1
Router(config-cmap)# class-map prec2
Router(config-cmap)# match precedence 2
Router(config-cmap)# class-map prec3
Router(config-cmap)# match precedence 3
Router(config-cmap)# class-map prec4
Router(config-cmap)# match precedence 4
Router(config-cmap)# end
```

例：QoS に対するポリシー マップの設定

```
Router# configure terminal
Router(config)# policy-map child-vlan
Router(config-pmap)# class prec1
Router(config-pmap-c)# police cir percent 20
Router(config-pmap-c-police)# exit
Router(config-pmap-c)# priority level 1
Router(config-pmap-c)# class prec2
Router(config-pmap-c)# police cir percent 40
Router(config-pmap-c-police)# exit
Router(config-pmap-c)# priority level 2
Router(config-pmap-c)# class prec3
Router(config-pmap-c)# bandwidth remaining ratio 3
Router(config-pmap-c)# class class-default
Router(config-pmap-c)# bandwidth remaining ratio 1
Router(config-pmap-c)# random-detect
Router(config-pmap-c)#!
Router(config-pmap-c)# policy-map egress_policy
Router(config-pmap)# class vlan_2
Router(config-pmap-c)# shape average 100000000
Router(config-pmap-c)# service-policy child-vlan
Router(config-pmap-c)# class vlan_3
Router(config-pmap-c)# shape average 200000000
Router(config-pmap-c)# service-policy child-vlan
Router(config-pmap-c)# class vlan_4
Router(config-pmap-c)# shape average 300000000
Router(config-pmap-c)# service-policy child-vlan
Router(config-pmap-c)#!
Router(config-pmap-c)# policy-map ingress_policy
Router(config-pmap)# class vlan_2
Router(config-pmap-c)# police cir 80000000
Router(config-pmap-c-police)# conform-action set-prec-transmit 1
Router(config-pmap-c-police)# class vlan_2
Router(config-pmap-c)# set dscp AF21
Router(config-pmap-c)# class class-default
Router(config-pmap-c)# set dscp 0
Router(config-pmap-c)# end
```

例：ポート チャネル インターフェイスへの QoS の適用

```
Router# configure terminal
Router(config)# interface port-channel 1
Router(config-if)# service-policy output egress_policy
Router(config-if)# service-policy input ingress_policy
Router(config-if)# end
```

集約 EtherChannel サブインターフェイス QoS の設定方法

手順の概要

1. **enable**
2. **configure terminal**
3. **platform qos port-channel-aggregate** *port-channel-number*
4. **interface port-channel** *port-channel-number*
5. **interface port-channel** *port-channel-number.subinterface-number*
6. **service-policy** {**output**} *policy-map*
7. **service-policy** {**input**} *policy-map*
8. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**platform qos port-channel-aggregate** *port-channel-number* 例： `Device(config)# platform qos port-channel-aggregate 1`	集約ポートチャネルインターフェイスをイネーブルにします。
ステップ 4	**interface port-channel** *port-channel-number* 例： `Device(config)# interface port-channel 1`	特定のポートチャネルを設定するインターフェイス コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 5	**interface port-channel** *port-channel-number.subinterface-number* 例： `Device(config)# interface port-channel 1.2`	特定のポート チャネル サブインターフェイスを設定するインターフェイス コンフィギュレーション モードを開始します。
ステップ 6	**service-policy** {**output**} *policy-map* 例： `Device(config-if)# service-policy output egress_policy`	インターフェイスのサービス ポリシーとして使用するポリシーマップを出力インターフェイスに付加します。
ステップ 7	**service-policy** {**input**} *policy-map* 例： `Device(config-if)# service-policy input ingress_policy`	インターフェイスのサービス ポリシーとして使用するポリシーマップを入力インターフェイスに付加します。
ステップ 8	**end** 例： `Device(config)# end`	グローバル コンフィギュレーション モードを終了します。

集約 EtherChannel サブインターフェイス QoS の設定解除方法

手順の概要

1. **enable**
2. **configure terminal**
3. **no interface port-channel** *port-channel-number.subinterface*
4. **no platform qos port-channel-aggregate** *port-channel-number*
5. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 3	**no interface port-channel** *port-channel-number.subinterface* 例： `Device(config)# no interface port-channel 1.2`	特定のポートチャネルのサブインターフェイスの設定を解除します。
ステップ 4	**no platform qos port-channel-aggregate** *port-channel-number* 例： `Device(config)# no platform qos port-channel-aggregate 1`	集約ポートチャネルインターフェイスをディセーブルにし、そのインターフェイスにある必要なQoSポリシーを削除します。
ステップ 5	**end** 例： `Device(config)# end`	グローバル コンフィギュレーション モードを終了します。

集約 EtherChannel サブインターフェイス QoS の設定例

例：集約ポートチャネルインターフェイスとサブインターフェイスの設定

```
Device# configure terminal
Device(config)# platform qos port-channel-aggregate 2
Device(config)# interface port-channel 2
Device(config-if)# interface GigabitEthernet1/1/1
Device(config-if)# channel-group 2
Device(config-if)# interface GigabitEthernet1/1/0
Device(config-if)# channel-group 2
Device(config-if)# interface port-channel 2.200
Device(config-subif)# encapsulation dot1Q 200
Device(config-subif)# ip addr 15.0.1.2 255.255.255.0
Device(config-subif)# interface port-channel 2.300
Device(config-subif)# encapsulation dot1Q 300
Device(config-subif)# ip addr 15.0.2.2 255.255.255.0
Device(config-subif)# end
```

例：QoS に対するクラス マップの設定

```
Device# configure terminal
Device(config)# class-map vlan_2
Device(config-cmap)# match vlan  2
Device(config-cmap)# class-map vlan_3
Device(config-cmap)# match vlan  3
Device(config-cmap)# class-map vlan_4
Device(config-cmap)# match vlan  4
```

```
Device(config-cmap)# class-map prec1
Device(config-cmap)# match precedence 1
Device(config-cmap)# class-map prec2
Device(config-cmap)# match precedence 2
Device(config-cmap)# class-map prec3
Device(config-cmap)# match precedence 3
Device(config-cmap)# class-map prec4
Device(config-cmap)# match precedence 4
Device(config-cmap)# end
```

例：QoS に対するポリシー マップの設定

```
Device# configure terminal
Device(config)# policy-map subinterface_child
Device(config-pmap)# class prec1
Device(config-pmap-c)# police cir percent 30
Device(config-pmap-c-police)# exit
Device(config-pmap-c)# priority level 1
Device(config-pmap-c)# class prec2
Device(config-pmap-c)# police cir percent 30
Device(config-pmap-c-police)# exit
Device(config-pmap-c)# priority level 2
Device(config-pmap-c)# class prec3
Device(config-pmap-c)# bandwidth remaining ratio 3
Device(config-pmap-c)# class class-default
Device(config-pmap-c)# bandwidth remaining ratio 1
Device(config-pmap-c)#!
Device(config-pmap-c)# policy-map sub_egress_policy
Device(config-pmap-c)# class class-default
Device(config-pmap-c)# shape average 300000000
Device(config-pmap-c)# service-policy subinterface_child
Device(config-pmap-c)#!
Device(config-pmap-c)# policy-map sub_ingress_policy
Device(config-pmap)# class class-default
Device(config-pmap-c)# police cir 80000000
Device(config-pmap-c)# end
```

例：ポート チャネル サブインターフェイスへの QoS の適用

```
Device# configure terminal
Device(config)# interface port-channel 2.200
Device(config-if)# service-policy output egress_policy
Device(config-if)# service-policy input ingress_policy
Device(config)# interface port-channel 2.300
Device(config-if)# service-policy output egress_policy
Device(config-if)# service-policy input ingress_policy
Device(config-if)# end
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『*Cisco IOS Master Commands List, All Releases*』
QoS コマンド	『Cisco IOS Quality of Service Solutions Command Reference』

MIB

MIB	MIB のリンク
この機能によってサポートされる新しい MIB または変更された MIB はありません。	選択したプラットフォーム、Cisco ソフトウェア リリース、およびフィーチャ セットの MIB を検索してダウンロードする場合は、次の URL にある Cisco MIB Locator を使用します。 http://www.cisco.com/go/mibs

シスコのテクニカル サポート

説明	リンク
シスコのサポート Web サイトでは、シスコの製品やテクノロジーに関するトラブルシューティングにお役立ていただけるように、マニュアルやツールをはじめとする豊富なオンライン リソースを提供しています。 お使いの製品のセキュリティ情報や技術情報を入手するために、Cisco Notification Service（Field Notice からアクセス）、Cisco Technical Services Newsletter、Really Simple Syndication（RSS）フィードなどの各種サービスに加入できます。 シスコのサポート Web サイトのツールにアクセスする際は、Cisco.com のユーザ ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

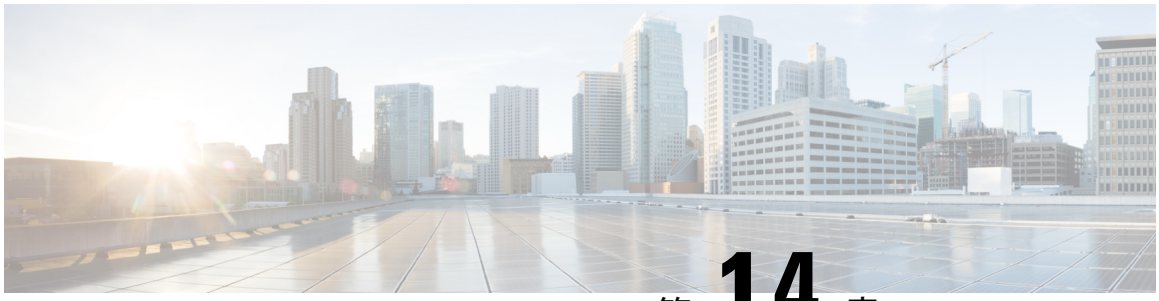
集約 EtherChannel QoS の機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *16 :* 集約 *EtherChannel QoS* の機能情報

機能名	リリース	機能情報
集約 EtherChannel QoS	Cisco IOS XE リリース 3.12S	集約 EtherChannel の QoS 機能を使用すると、ポートチャネルのメイン インターフェイスまたはサブインターフェイス上に集約出力キューイング ポリシーマップを適用できます。この機能によって、Cisco ASR 1000 シリーズ アグリゲーション サービス ルータに対する集約ポートチャネルのメイン インターフェイス上での QoS サポートが可能になります。 この機能は、Cisco IOS XE リリース 3.12S で、Cisco ASR 1000 シリーズ アグリゲーション サービス ルータに導入されました。
集約 GEC QoS 10G のサポート	Cisco IOS XE リリース 3.16.3S Cisco IOS XE Denali 16.3.1	この機能は、Cisco IOS XE リリース 3.16.3S で、Cisco ASR 1000 シリーズ アグリゲーション サービス ルータに導入されました。
ASR1K の GEC ポートチャネル サブインターフェイスでの QoS	Cisco IOS XE リリース 3.16.3S Cisco IOS XE Denali 16.3.1	この機能は、Cisco IOS XE リリース 3.16.3S で、Cisco ASR 1000 シリーズ アグリゲーション サービス ルータに導入されました。
ISR 4000 の GEC ポートチャネル サブインターフェイスでの QoS	Cisco IOS XE Everest 16.6.1	Cisco IOS XE Everest 16.6.1 リリースでは、この機能は Cisco ISR 4000 シリーズ サービス統合型ルータに導入されました。



第 14 章

PPPoGEC のセッション単位の QoS

PPPoGEC セッション単位の QoS 機能は、PPPoE/L2TP 環境（ブロードバンドの導入）内の PPP Termination and Aggregation（PTA）デバイス、L2TP アクセス コンセントレータ（LAC）デバイス、または L2TP ネットワーク サーバ（LNS）デバイス上での PPPoE セッションの特定の QoS ポリシーの設定をサポートします。PPPoE/L2TP 環境内で PTA デバイス、LAC デバイス、または LNS デバイスとして機能する Cisco ASR 1000 シリーズ ルータでは、EtherChannel アクティブ/スタンバイ機能による PPPoE セッションもサポートされています。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「Bug Search Tool」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

PPPoGEC のセッション単位の QoS について

PPPoGEC のセッション単位の QoS に関する制約事項

- QoS ポリシー マップは、メンバー リンク、ポートチャネルのメイン インターフェイス、または QoS を備えた PPPoE セッション用の送信パスと関連付けられたポートチャネル サブインターフェイス上には設定できません。

アクティブ/スタンバイ EtherChannel による PPPoGEC

アクティブ/スタンバイ EtherChannel による PPPoE セッションは 1 レベルまたは 2 レベルの階層型出力ポリシーマップを（キューイング設定を使用して）サポートし、また、フラット入力ポリシー マップも（キューイング設定なしで）サポートします。ポリシー マップは、以前に定義されたクラス マップを使用して設定されます。**class-map** コマンドを使用してトラフィック クラスが設定されている必要があります。

出力階層型ポリシー マップと入力ポリシー マップは、次のいずれかの方法で PPPoE セッションに関連付けることができます。

- 仮想テンプレート インターフェイスでの構成時の設定
- 認証、認可、およびアカウンティング（AAA）モデルで設定した外部ツール（RADIUS サーバなど）を使用したダイナミックな構成時設定。詳細については、『*Intelligent Services Gateway Configuration Guide*』と『*Cisco ASR 1000 Series Aggregation Services Routers Software Configuration Guide*』を参照してください。

ポートチャネル メイン インターフェイスにアクティブ/スタンバイ シナリオを作成する次のコマンドが含まれている必要があります。このような設定では、1 つのインターフェイスのみをアクティブにすることで、トラフィックをいつでも転送できます。

- **interface port-channel1**
- **lacp fast-switchover**
- **lacp max-bundle 1**

PPPoEGEC のセッション単位の QoS の設定方法

EtherChannel アクティブ/スタンバイによる PPPoE セッションでの QoS の設定

PPPoE セッションに QoS を設定するには、EtherChannel インターフェイスの PPP セッションに使用する仮想テンプレートを指定し、入力トラフィックに適用するサービスポリシーの名前を指定し、出力トラフィックを指定する必要があります。この設定では、仮想テンプレートインターフェイスを定義することによって、出力階層型ポリシー マップと入力ポリシー マップを PPPoE セッションに関連付ける方法を示します。

手順の概要

1. **enable**
2. **configure terminal**
3. **interface virtual-template** *number*
4. **service-policy output** *policy-map-name*
5. **service-policy input** *policy-map-name*
6. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**interface virtual-template** *number* 例： `Device(config)# interface virtual-template 99`	バーチャル アクセス インターフェイスの作成で動的に設定して適用できるバーチャル テンプレート インターフェイスを作成し、インターフェイス コンフィギュレーション モードを開始します。 • EtherChannel インターフェイスの PPP セッションに使用する仮想テンプレートを指定します。
ステップ 4	**service-policy output** *policy-map-name* 例：	出力トラフィックに適用するサービスポリシーの名前を指定します。

	コマンドまたはアクション	目的
	`Device(config-if)# service-policy output session_parent`	
ステップ 5	**service-policy input** *policy-map-name* 例： `Device(config-if)# service-policy input session_ingress`	入力トラフィックに適用するサービス ポリシーの名前を指定します。
ステップ 6	**end** 例： `Device(config-if)# end`	インターフェイス コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

PPPoGEC のセッション単位の QoS の設定例

例：EtherChannel アクティブ/スタンバイによる PPPoE セッションでの QoS

次の例に、session_parent 階層型ポリシー マップと session_ingress ポリシー マップを示します。これらのポリシー マップは、**service-policy** コマンドを使用して仮想テンプレート インターフェイスに適用します。

```
policy-map session_child
  class voice
    priority level 1
    police cir 256000
    set precedence 5
  class web
    bandwidth remaining ratio 10
  class p2p
    bandwidth remaining ratio 1
    set precedence 1
  class class-default
    set precedence 2
    bandwidth remaining ratio 5
!
policy-map session_parent
  class class-default
    bandwidth remaining ratio 1
    shape average 25000000
    service-policy session_child
!
policy-map session_ingress
  class voip
    police cir 256000
  class p2p
    police cir 256000 pir 512000
      conform-action set-prec-transmit 1
```

```
      exceed  set-prec-transmit 0
      violate drop
  class class-default
    police cir 5000000
      conform-action set-prec-transmit 2
      exceed  drop
!
interface Virtual-template 99
 service-policy output session_parent
 service-policy input session_ingress
```

PPPoGEC のセッション単位の QoS のその他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『*Cisco IOS Master Command List, All Releases*』
QoS コマンド：コマンド構文の詳細、コマンド モード、コマンド履歴、デフォルト設定、使用上のガイドライン、および例	『*Cisco IOS Quality of Service Solutions Command Reference*』
モジュラ QoS コマンドライン インターフェイス	「Applying QoS Features Using the MQC」モジュール
RADIUS ベースのポリシングの設定	『*Intelligent Services Gateway Configuration Guide*』
CISCO ASR 1000 シリーズ ソフトウェアの設定	『*Cisco ASR 1000 Series Aggregation Services Routers Software Configuration Guide*』

シスコのテクニカル サポート

説明	リンク
★枠で囲まれた Technical Assistance の場合★右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

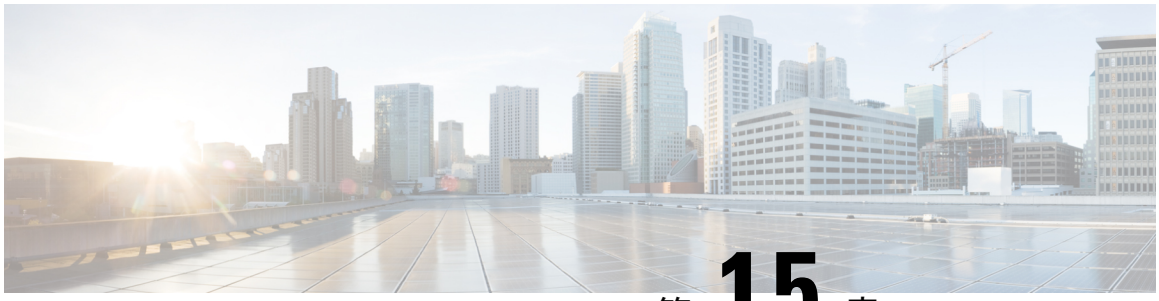
PPPoGEC のセッション単位の QoS の機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *17 : PPPoGEC* のセッション単位の *QoS* の機能情報

機能名	リリース	機能情報
PPPoGEC：セッション単位の QoS	Cisco IOS XE リリース 3.7S	この機能は、ブロードバンドの導入での PTA、LAC、および LNS での PPPoE セッションの特定の QoS ポリシーの設定をサポートします。 この機能は、Cisco ASR 1000 シリーズ ルータに追加されました。 Cisco IOS XE リリース 3.8S では、PPPoGEC の 1:1 モードのセッション単位の QoS に追加されました。また、PPPoGEC では、Point-to-Point Protocol（PPP）と IP over PPPoE に対するサポートも追加されました。 Cisco IOS XE リリース 3.9Sでは、1:1 モードの GEC 上の IP セッションに対するサポートが追加されました。



第 15 章

IPv6 選択的パケット廃棄

選択的パケット廃棄（SPD）メカニズムは、RP 上のプロセス レベルの入力キューを管理します。SPD では、プロセス レベル キューに輻輳が発生している間、ルーティング プロトコル パケットや、その他の重要なトラフィック制御レイヤ 2 キープアライブが優先されます。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「Bug Search Tool」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

IPv6 選択的パケット廃棄に関する情報

IPv6 での SPD の概要

SPD メカニズムは、RP 上でプロセス レベル入力キューを管理します。SPD では、プロセス レベル キューに輻輳が発生している間、ルーティング プロトコル パケットや、その他の重要なトラフィック制御レイヤ 2 キープアライブが優先されます。

SPD ステート チェック

RP 上の IPv6 プロセス入力キューでは、SPD ステート チェックが実行されます。IP プレシデンスが 6 などのプライオリティの高いパケットは、SPD の対象にはならず、決してドロップされることはありません。一方、それ以外のすべてのパケットは、IPv6 パケット入力キューの長さと SPD ステートに従ってドロップされる可能性があります。SPD ステートには次の種類があります。

- Normal：プロセス入力キューの長さは、SPD の最小しきい値未満です。
- Random drop：プロセス入力キューの長さは、SPD の最小しきい値と最大しきい値の間です。
- Max：プロセス入力キューの長さは、SPD 最大しきい値と同じです。

プロセス入力キューのサイズによって SDP ステートが normal（ドロップなし）か、random drop か、max かが決まります。プロセス入力キューが SPD の最小しきい値よりも小さい場合、SPD は何も行わず、normal ステートになります。normal ステートでは、パケットはドロップされません。入力キューが最大しきい値に到達すると、SPD は max ステートになります。このステートでは、通常プライオリティのパケットが廃棄されます。入力キューが最小しきい値と最大しきい値の間にある場合、SPD は random drop ステートになります。このステートでは、通常パケットがドロップされることがあります。

SPD モード

none（デフォルト）、aggressive drop、および OSPF の 3 つのIPv6 SPD モードがサポートされています。aggressive drop モードでは、IPv6 が random drop ステートのとき、フォーマットに誤りのあるパケットはドロップされます。OSPF モードでは、OSPF パケットが SPD プライオリティで処理されるメカニズムが提供されます。

SPD ヘッドルーム

SPD では、通常の IPv6 パケットの動作は変更されません。一方、ルーティング プロトコル パケットは、SPD が IPv6 precedence フィールドで認識するため、より高いプライオリティが与えられます。したがって、IPv6 プレシデンスが 6 に設定されていると、そのパケットが優先されます。

SPD では、プレシデンスが 6 の IPv6 パケットを優先させるために、それらを通常の入力キュー制限を超えてプロセス レベル入力キューにキューイングすることを Cisco IOS ソフトウェアに許可します。通常の制限を超えて許可されるパケットの数は、SPD ヘッドルームと呼ばれます。SPD ヘッドルームのデフォルトは 100 です。つまり、プレシデンスの高いパケットは、入力保持キューのサイズが 175（入力キューのデフォルト サイズ + SPD ヘッドルーム サイズ）よりも小さければドロップされません。

内部ゲートウェイ プロトコル（IGP）とリンクの安定性は微妙で重要なので、これらのパケットには最も高いプライオリティと、デフォルトで 10 パケットの追加 SPD ヘッドルームが与えられます。これらのパケットは、入力保持キューのサイズが 185（入力キューのデフォルト サイズ + SPD ヘッドルーム サイズ + SPD 拡張ヘッドルーム）未満であれば、ドロップされません。

Connectionless Network Service Intermediate System-to-Intermediate System（CLNS IS-IS）パケット、PPPパケット、およびハイレベルデータリンク コントロール（HDLC）キープアライブのような非 IPv6 パケットは、レイヤ 3 でなくレイヤ 2 であるために通常のプライオリティとして扱われます。さらに、レイヤ 3 以上で動作する IGP には、通常の IPv6 パケットよりも高いプライオリティが与えられますが、これはボーダ ゲートウェイ プロトコル（BGP）パケットと同じプライオリティです。したがって、BGP 輻輳中または BGP アクティビティが非常に活発な間は、IGP の Hello パケットや KeepAlive パケットがドロップされ、それによって IGP 隣接が失敗することがあります。

IPv6 選択的パケット廃棄の設定方法

SPD プロセス入力キューの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 spd queue max-threshold** *value*
4. **ipv6 spd queue min-threshold** *value*
5. exit
6. **show ipv6 spd**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 spd queue max-threshold** *value* 例： `Router(config)# ipv6 spd queue max-threshold 60000`	SPDプロセス入力キュー内の最大パケット数を設定します。
ステップ 4	**ipv6 spd queue min-threshold** *value* 例： `Router(config)# ipv6 spd queue max-threshold 4094`	IPv6 SPDプロセス入力キュー内の最小パケット数を設定します。

	コマンドまたはアクション	目的
ステップ 5	exit 例： `Router(config)# exit`	ルータを特権 EXEC モードに戻します。
ステップ 6	**show ipv6 spd** 例： `Router# show ipv6 spd`	IPv6 SPD 設定を表示します。

SPD モードの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **ipv6 spd mode** {**aggressive** | **tos protocol ospf**}

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ipv6 spd mode** {**aggressive**	**tos protocol ospf**} 例： `Router(config)# ipv6 spf mode aggressive`	IPv6 SPD モードを設定します。

SPD ヘッドルームの設定

手順の概要

1. **enable**
2. **configure terminal**

3. **spd headroom** *size*
4. **spd extended-headroom** *size*
5. exit
6. **show ipv6 spd**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Router> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Router# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**spd headroom** *size* 例： `Router(config)# spd headroom 200`	SPD ヘッドルームを設定します。
ステップ 4	**spd extended-headroom** *size* 例： `Router(config)# spd extended-headroom 11`	拡張 SPD ヘッドルームを設定します。
ステップ 5	exit 例： `Router(config)# exit`	ルータを特権 EXEC モードに戻します。
ステップ 6	**show ipv6 spd** 例： `Router# show ipv6 spd`	IPv6 SPD 設定を表示します。

IPv6 選択的パケット廃棄の設定例

例：SPD プロセス入力キューの設定

次に、SPDプロセス入力キュー設定の例を示します。最大プロセス入力キューしきい値は60,000で、SPD ステートは normal です。ヘッドルームおよび拡張ヘッドルームの値は、デフォルトです。

```
Router# ipv6 spd queue max-threshold 5000
Router# show ipv6 spd

Current mode: normal
Queue max threshold: 60000, Headroom: 100, Extended Headroom: 10
IPv6 packet queue: 0
```

その他の参考資料

関連資料

関連項目	マニュアル タイトル
IPv6 アドレッシングと接続	『*IPv6 Configuration Guide*』
Cisco IOS コマンド	『Master Commands List, All Releases』
IPv6 コマンド	『IPv6 Command Reference』
Cisco IOS IPv6 機能	IPv6 機能のマッピング

標準および RFC

標準/RFC	タイトル
IPv6 に関する RFC	*IPv6 RFCs*

テクニカル サポート

説明	リンク
★枠で囲まれた Technical Assistance の場合★右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

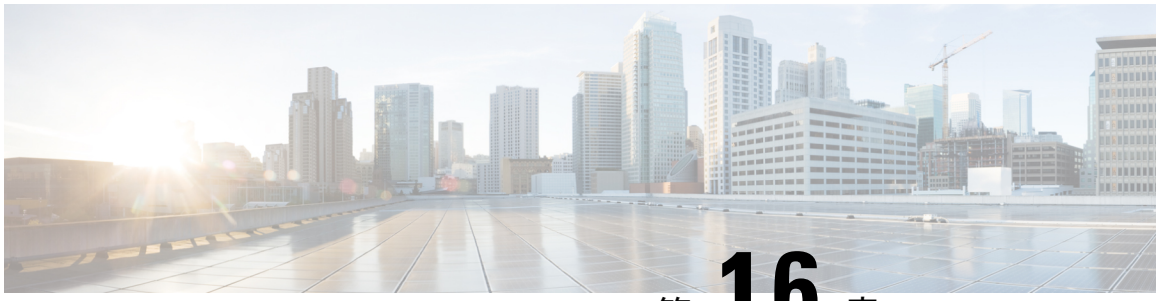
IPv6 選択的パケット廃棄の機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *18 : IPv6* 選択的パケット廃棄の機能情報

機能名	リリース	機能情報
IPv6：完全な選択的パケット廃棄のサポート	Cisco IOS XE リリース 2.6	SPD メカニズムは、RP 上でプロセス レベル入力キューを管理します。SPD では、プロセス レベル キューに輻輳が発生している間、ルーティング プロトコル パケットや、その他の重要なトラフィック制御レイヤ 2 キープアライブが優先されます。 次のコマンドが導入または変更されました。**clear ipv6 spd**、**debug ipv6 spd**、**ipv6 spd mode**、**ipv6 spd queue max-threshold**、**ipv6 spd queue min-threshold**、**monitor event-trace ipv6 spd**、**show ipv6 spd**、**spd extended-headroom**、**spd headroom**



第 16 章

ACE 単位の QoS 統計情報

ACE 単位の QoS 統計情報機能により、QoS パケット一致統計情報機能が拡張され、フィルタ内に使用されている個々のアクセス制御要素（ACE）に一致するパケット数とバイト数を追跡することができます。フィルタは、Quality of Service （QoS）ポリシー マップのクラスマップ定義の一部です。

show access-lists コマンドを使用すると、ACE 単位の統計情報を表示できます。

QoS パケット フィルタの定義と、そのフィルタに一致するパケット数とバイト数の表示については、「QoS パケット一致統計情報」モジュールを参照してください。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、「Bug Search Tool」およびご使用のプラットフォームおよびソフトウェア リリースのリリース ノートを参照してください。このモジュールで説明される機能に関する情報、および各機能がサポートされるリリースの一覧については、機能情報の表を参照してください。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

ACE 単位の QoS 統計情報の前提条件

platform qos match-statistics per-ace コマンドを設定して QoS の ACE 単位のパケット一致統計情報をイネーブルにする前に、**platform qos match-statistics per-filter** コマンドを設定して QoS のフィルタ単位のパケット一致統計情報をイネーブルにする必要があります。これを行わないと CLI がコマンドを拒否し、エラー メッセージが表示されます。

ACE 単位の QoS 統計情報の制約事項

platform qos match-statistics per-ace コマンドを設定するときに QoS ポリシー マップをデバイスに適用した場合、このコマンドは次のいずれかの時点まで有効になりません。

- デバイスのリロード。
- すべての QoS ポリシーの適用解除とコマンドの再設定。

ACE 単位の QoS 統計情報機能をイネーブルにすると、拡張設定では CPU 使用率が増加する可能性があります。この機能をイネーブルにする前に、統計情報の利点とシステムの CPU 使用率の増加とを比較して検討する必要があります。

（注） **platform qos match-statistics per-ace** コマンドを設定する前に、**platform qos match-statistics per-filter** コマンドを設定する必要があります。

ACE 単位の QoS 統計情報に関する情報

ACE 単位の QoS 統計情報の概要

ACE 単位の QoS 統計情報機能は、QoS ポリシーで使用する ACE のヒット カウンタを提供します。この機能を有効にすると、QoS ポリシーに使用された ACE の QoS ヒット カウンタを、その ACE の既存のセキュリティ アクセスリスト カウンタに追加します。**show ip access-lists** コマンドを使用すると、次の例に示すように、アクセスリスト カウンタを表示できます。

```
Device# show ip access-lists

Extended IP access list A1
10 permit ip 10.1.1.0 0.0.0.255 any (129580275 matches)
Extended IP access list A6and7
10 permit ip 10.1.6.0 0.0.0.255 any (341426749 matches)
20 permit ip 10.1.7.0 0.0.0.255 any (398245767 matches)
Extended IP access list source
10 permit ip any host 10.1.1.5 (16147976 matches)
```

QoS ヒット カウンタ（QoS ポリシーで使用された ACE の数）が、出力例に示したカウンタに追加されます。

ACE 単位の QoS 統計機能を有効にする場合は、次の条件に注意してください。

- **show ip access-lists** コマンドは、インターフェイス情報を表示しません。つまり、アクセスリスト カウントはインターフェイス固有ではありません。このカウントは、ACE を使用し、すべてのインターフェイスと方向全体のカウントをサポートするすべての機能のすべてのヒット数の集約カウンタです。
- **show policy-map interface** コマンドを使用すると、QoS のフィルタ単位のパケット一致統計情報がイネーブルになっている場合は、インターフェイス固有のカウントを表示できます。ただし、このコマンドは、次の例に示すように、ACE 単位のカウントではなく、フィルタ単位（アクセス制御リスト（ACL）またはアクセス グループ）のカウントのみを表示します。

```
Device# show policy-map interface GigabitEthernet0/0/2

 GigabitEthernet0/0/2

  Service-policy input: test-match-types

    Class-map: A1orA2-class (match-any)
      482103366 packets, 59780817384 bytes
      5 minute offered rate 6702000 bps
      Match: access-group name A1
        62125633 packets, 7703578368 bytes
        5 minute rate 837000 bps
      Match: access-group name A2
        419977732 packets, 52077238892 bytes
        5 minute rate 5865000 bps
```

- QoS フィルタ（つまり、クラス マップ内の match ステートメント）に ACE があっても、パケットがその ACE に一致しない場合、ACE カウンタはそのパケットについては増加しません。これは、次の状況で発生する可能性があります。
  - ACE を「deny」ステートメントで使用している。
  - 「match-all」クラス マップ定義の他の一致基準（「match ip prec 1」など）がパケットをクラスに一致させないようにしている。
  - 「match-any」クラス マップ定義の他の一致基準（「match ip prec 1」など）がパケットと一致し、そのパケットを ACE 一致基準と一致させないようにしている（そのフィルタが ACE フィルタよりも前にあり、そのパケットが両方のステートメントに一致している）。
- アクセスリスト カウントは、ACE を使用し、ACE 単位のカウントをサポートするすべての機能のその特定の ACE についてのヒット カウントの総計です。つまり、1 つのパケットが同じ ACE を使用している複数の機能にヒットしてカウントされ、それぞれの機能を通過するときに、同じパケットで複数のカウントになる可能性があります。

  次に、このような複数のカウントが発生する例を示します。インターフェイスで受信したパケットは 1,000 個のみでしたが、アクセスリストのカウントには、セキュリティ アクセス リストに 1,000、QoS サービス ポリシーに 1,000 で 2,000 のヒット数が表示されます。

```
Device(config)# ip access-list extended A1
permit ip 32.1.1.0 0.0.0.255 any
class-map match-all A1-class
match access-group name A1
interface GigabitEthernet0/0/2
ip address 10.0.0.1 240.0.0.0
ip access-group A1 in
duplex auto
speed auto
media-type rj45
no negotiation auto
service-policy input simple
end

Device# show access-lists

Extended IP access list A1
10 permit ip 10.1.1.0 0.0.0.255 any (2000 matches)

Device# show policy-map interface GigabitEthernet0/0/2

Service-policy input: simple
Class-map: A1-class (match-all)
1000 packets, 124000 bytes
5 minute offered rate 4000 bps
Match: access-group name A1
Class-map: class-default (match-any)
0 packets, 0 bytes
5 minute offered rate 261000 bps, drop rate 0 bps
Match: any
```

ACE 単位の QoS 統計情報の設定方法

ACE 単位の QoS 統計情報の設定

始める前に

QoS のフィルタ単位のパケット一致統計情報をイネーブルにするには、**platform qos match-statistics per-filter** コマンドを設定する必要があります。**show platform hardware qfp active feature qos config global** コマンドを使用して、パケット一致統計情報のステータスを確認できます。

```
Device# show platform hardware qfp active feature qos config global

Marker statistics are: disabled
Match per-filter statistics are: enabled <<<<<<<
Match per-ace statistics are: disabled <<<<<<
Performance-Monitor statistics are: disabled
```

手順の概要

1. **platform qos match-statistics per-filter**
2. **platform qos match-statistics per-ace**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**platform qos match-statistics per-filter** 例： `Device(config)# platform qos match-statistics per-filter`	クラス マップ内の個々のフィルタに対して QoS パケット一致統計情報をイネーブルにします。
ステップ 2	**platform qos match-statistics per-ace** 例： `Device(config)# platform qos match-statistics per-ace`	QoS フィルタに使用されている ACE に対して QoS パケット一致統計情報をイネーブルにします。

ACE 単位の QoS 統計情報のその他の参考資料

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『*Cisco IOS Master Command List, All Releases*』
QoS コマンド	『*Cisco IOS Quality of Service Solutions Command Reference*』
QoS パケット フィルタの定義とそれに一致するパケットおよびバイトの数の表示	「QoS パケット一致統計情報」

シスコのテクニカル サポート

説明	リンク
★枠で囲まれた Technical Assistance の場合★右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

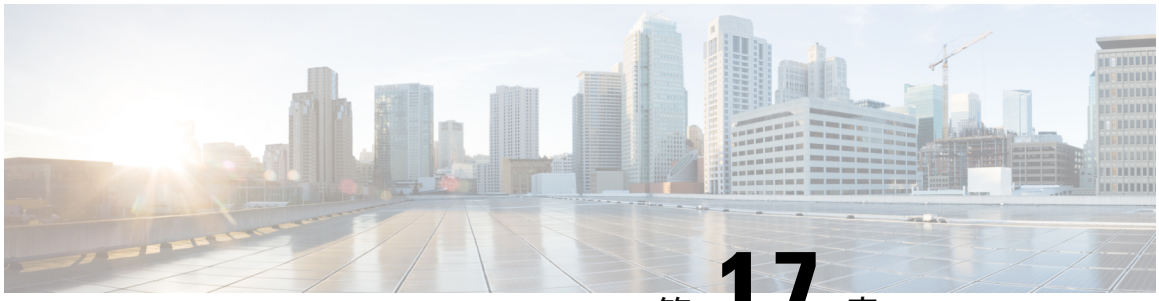
ACE 単位の QoS 統計情報の機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *19 : ACE* 単位の *QoS* 統計情報の機能情報

機能名	リリース	機能情報
ACE 単位の QoS 統計情報	Cisco IOS XE リリース 3.10S	ACE 単位の QoS 統計情報を設定して、QoS サービス ポリシー内のフィルタに使用されている個々の ACE に一致するパケットとバイトの数を追跡できます。 次のコマンドが導入または変更されました：**platform qos match-statistics per-ace**。



第 17 章

QoS パケット ポリシング

トラフィック ポリシングを使用すると、ネットワーク トラフィックが事前に決められたレートを上回っているか下回っているかを判断し、そのようなトラフィックに対して異なる処理を実行できます。最も単純な形式では、ポリサー（レート リミッタ）は、事前に決められたレートを超えるトラフィックをすべてドロップします。

QoS ポリシングについて

トラフィック ポリシングを使用する理由

インターフェイスで送受信されるトラフィックの最大レートの制御を可能にするトラフィック ポリシングは、一般に、ネットワークのエッジにあるインターフェイスで、ネットワークへのトラフィックを制限するように設定されます。ほとんどのトラフィック ポリシング設定では、レート パラメータ内に収まるトラフィックは送信されますが、パラメータを超えるトラフィックはドロップされるかマークされ（かつ送信され）ます。

（注） シェーパーとは異なり、ポリサーはパケットをバッファリングしません。代わりに、指定されたアクションがただちに実行されます。

シスコでは、通常、ポリサーをアドミッション制御（キューまたはネットワーク）に使用します。

キュー アドミッション制御 により、キューに入ることができるデータの量を制限できます。プライオリティ キュー は、キューに入るパケットのレートを制限して遅延を回避するというこのカテゴリの代表的な例です。

ネットワーク アドミッション制御 では、ネットワーク管理者（サービス プロバイダー）とその顧客の間の契約が適用されます。一般に、プロバイダーがトラフィックを受け入れる必要があるレートに両者が同意します。これはサービスレート（顧客がプロバイダーに送信するすべてのトラフィックの最大レート）またはクラスごとの制限（たとえば、顧客が送信できるプライオリティ トラフィックの量）である場合があります。

- ネットワーク アドミッション制御を使用する場合、超過トラフィックをただちにドロップするか「契約外」としてマークするかを決定できます。後者の場合は、このネットワーク内で輻輳が発生した場合に（そのときに）、そのトラフィックの処理量を減らすか、そのトラフィックを最初にドロップすることができます。

ポリサーの定義

（注） 「ポリサー」と「レート リミッタ」という 2 つの用語 通常、同じ QoS メカニズムを指します。このドキュメントでは、「ポリサー」（ポリシング）を使用します。

ポリサーは、パケットが指定されたレートを超えて受信されたか超えずに受信されたかに応じて同じトラフィック クラス内のパケットに関して異なる処理を定義することを可能にするデバイスです。

最も単純な形式では、ポリサーは、指定されたレートを超えたトラフィックをドロップする必要があることを示します。

```
policy-map police-all-traffic
  class class-default
    police 1m
```

このクラスを通過するトラフィックが 1 Mbps 未満のレートで着信する場合、そのトラフィックは適合（指定されたレートに準拠）していると見なされます。適合トラフィックに対するデフォルトのアクションは、パケットの転送です。

1 Mbps を超えるレートで到着するトラフィックは、設定されたレートを超過していると見なされます。超過トラフィックに対するデフォルトのアクションは、パケットのドロップです。

次の定義は、以降のセクションの説明に適用されます。

表 *20* : ポリサーのコア定義

用語	定義

Bc（適合バースト サイズ）	バースト許容値を定義する CIR と同時に使用して、パケットが適合していると見なすかどうかを決定します。
Be（超過バースト サイズ）	3 カラー ポリサーがシングル レートとディアル レートのどちらなのかによって意味が異なります。 • シングル レート ポリサーの場合、Be では追加のバースト許容値を適合バーストを超えて定義できます。これにより、適合レベルをごくわずかに上回ったトラフィックと、そのレベルを大幅に上回ったトラフィックを認識できます。 • デュアル レート ポリサーでは、Bc を CIR と同時に使用するのとまったく同じように、Be を PIR と同時に使用して、トラフィックがそのレートを超えている（またはそのレート未満）かどうかを判断できます。
Burst（バースト）	非常に多くのパケットが緊密に到着した時点。 （注） 短い間隔で測定されたレートは、長い間隔で測定した場合に、レートを正確に反映していない可能性があります。
CIR（認定情報レート）	所定の間隔で転送可能なデータの最大量。
Conform（適合）	CIR よりも低いレートで到着するトラフィック（バーストを考慮）。
Exceed（超過）	ポリサーのタイプがシングルかデュアルか、2 カラーか 3 カラーかによって異なります。 • シングル レート 2 カラー ポリサーで、CIR を超えるレートで着信するトラフィック（バーストを考慮）。 • シングル レート 3 カラー ポリサーで、CIR を超えているが（つまり、適合バケットが枯渇している）、超過バケットを枯渇させるほど高くないレートで着信するトラフィック。 • デュアル レート ポリサーで、CIR を超えているが、PIR よりも低いレートで着信するトラフィック（どちらのインスタンスでもバースト許容値を考慮）。
PIR（最大情報レート）	（デュアル レート ポリサーにのみ関連）このレート未満のトラフィックが超過。上回る、違反している。 PIR は、デュアル レート ポリサーで設定された高速のレートです。 このレートが差別化要因です。 超過に指定されたトラフィックと違反に指定されたトラフィック。

Violate（違反）	ポリサーのタイプに依存します。 • シングル レート 3 カラー ポリサーの場合は、超過バケットを枯渇させるのに十分な高いレートで着信するトラフィック。 • デュアル レート ポリサーの場合は、PIR よりも高いレートで到着するトラフィック（バーストを考慮）。

ポリサーのアクション

次にコピーした前の例では、ポリサーが最も基本的な形式で使用されています。

```
policy-map police-all-traffic
  class class-default
    police 1m
```

適合トラフィックはポリサーを通過することが許可され（送信トラフィックは1m未満）、超過トラフィックはドロップされていました。 トラフィックが 指定された レートを超えたことを認識したときには即時のアクションを実行していました。しかし、即時のアクションが望ましくない場合もあります。場合によっては、トラフィックをただちにドロップするのではなく、アクションを延期する 必要があります。

たとえば、事前に決められたレートを超えるトラフィックを、ネットワークが輻輳している場合にのみドロップするように決定できます。この場合、すべてのトラフィックを転送するものの適合トラフィックと超過トラフィックでは異なる方法でパケット内の何か（たとえば、DSCP）にマークすることを選択できます。ドロップするかどうかの決定は、その後に輻輳が発生した時点で行うことができます。

次の例では、 トラフィックをドロップする代わりにマークします。 トラフィック クラスを DSCP 値が AF41 の着信トラフィックと定義し、指定したレートを超えるトラフィックを AF42 に降格させます。

```
policy-map ma
rk-out-of-contract
  class AF41
    police 1m conform-action transmit exceed-action set-dscp-transmit AF42
```

この *conform-action* の内容 は、指定された 1 Mbps 以下のレートで着信するトラフィックの送信（デフォルト動作の単純転送）です。

この *exceed-action*（ 1 Mbps を超過したトラフィックに適用される）の内容は、トラフィックのドロップではなくパケットの DSCP 値のマークです。

transmit と drop は、指定されたレートに適合するトラフィック またはそれを超過するトラフィックに対して指定されたアクションを表します。 **police** コマンドを使用してアクションを指定します。サポートされているアクションを次の表に示します。

表 *21 : police* コマンドのアクション

指定された処理	結果
drop	パケットをドロップします。
set-clp-transmit	ATM セルの ATM セル損失率優先度（CLP）ビットを設定し、パケットを送信します。
set-cos-inner-transmit *cos-value*	Q-in-Q の導入では、パケットの内部サービス クラス（CoS）値を設定し、パケットを送信します。CoS 値の範囲は 0 ～ 7 です。
set-cos-transmit *cos-value*	パケットのサービス クラス（CoS）値を設定し、パケットを送信します。CoS 値の範囲は 0 ～ 7 です。
set-discard-class-transmit *discard-class-value*	廃棄クラス値を設定し、パケットを送信します。廃棄クラス値の範囲は 0 ～ 7 です。
set-dscp-transmit *dscp-value*	IP Diffserv コード ポイント（DSCP）値を設定して、パケットを送信します。有効値の範囲は 0 ～ 63 です。
set-dscp-tunnel-transmit *dscp-value*	現在のパケットにこのようなヘッダーが追加された場合はその時点でトンネル ヘッダーに書き込まれる DiffServ コード ポイント（DSCP）値を保存します。
set-frde-transmit	フレーム リレー廃棄適性（DE）ビットを設定し、フレームを送信します。
set-mpls-exp-imposition-transmit *mpls-exp-value*	インポーズされたラベル ヘッダーに MPLS EXP ビットを設定し、MPLS ラベルがインポーズされた場合はその時点でパケットを送信します。有効値の範囲は 0 ～ 7 です。
set-mpls-exp-topmost-transmit *mpls-exp-value*	最上位ラベルに MPLS EXP ビットを設定し、パケットを送信します。有効値の範囲は 0 ～ 7 です。
set-prec-transmit *precedence-value*	IP Precedence レベルを設定し、パケットを送信します。有効値の範囲は 0 ～ 7 です。
set-prec-tunnel-transmit *precedence-value*	現在のパケットにこのようなヘッダーが追加された場合はその時点でトンネル ヘッダーに書き込まれるトンネル IP プレシデンス値を保存します。有効値の範囲は 0 ～ 7 です。
set-qos-transmit *qroup-id*	「qos-group」値を設定し、パケットを送信します。有効値の範囲は 0 ～ 99 です。
transmit	パケット内のフィールドを変更せずにそのパケットを送信します。

（注） ポリサー アクションのルールは、**set** コマンドのルールと非常によく似ています。

マークできるのは、レイヤ 2 のヘッダーと外部レイヤ 3 のヘッダーだけです。

マルチアクション ポリサー

前のセクションでは、パケット内の特定のフィールドをマークするようにポリサーを設定する方法について説明しました。実際には、パケット内の複数のフィールドをマークすることができます。

トラフィック クラス内で複数の set アクションを設定する場合と同様の方法で、各レート指定内のトラフィックに複数のアクションを適用できます。たとえば、パケットが TCP/IP 環境とフレームリレー環境の両方で送信されることがわかっている場合、超過パケットまたは違反パケットの DSCP 値を変更し、フレームリレー廃棄特性（DE）ビットを 0 ～ 1 の値に設定して優先度が低いことを示すことができます。

複数のポリシング アクションを指定するときは、次の点に注意してください。

- ポリシーマップ クラス ポリシング コンフィギュレーション（config-pmap-c-police）サブモードを開始する必要があります。
- 最大 4 つのアクションを同時に指定できます（1 アクションにつき 1 行）。
- 矛盾するアクション（**conform-action transmit** と **conform-action drop** など）は指定できません。

set コマンドと同様に、同じパケットに複数のアクション（たとえば、レイヤ 2 フィールドとレイヤ3フィールドをマークする）を設定するか、異なるトラフィック タイプごとにアクション（たとえば、IPv4 パケットの DSCP 値をマークし、MPLS パケットの Experimental（EXP）ビットをマークする）を定義することができます。

次の例では、RTP トラフィック（`rtp-traffic`）の上限を 1 Mbps に設定し、そのレートを超えるトラフィックをドロップします（`exceed-action drop`）。適合トラフィックについては、IPv4 パケットの COS 値および DSCP 値と MPLS パケットの COS 値および EXP ビットの両方をマークします。

```
class rtp-traffic
  police cir 1000000
    conform-action set-cos-transmit 4
    conform-action set-dscp-transmit af41
    conform-action set-mpls-exp-topmost-transmit 4
    exceed-action drop
```

トラフィック クラスによって観測されるレートにはそのクラス内のすべてのパケットが考慮されますが、アクションは該当するトラフィックだけに適用されます。たとえば、IPv4 パケットと MPLS パケットが同じトラフィック クラスに分類され、ポリサーが特定の DSCP 値をマークするように設定されているとします。観測されるレートには IPv4 パケットと MPLS パケットの両方が考慮されますが、マークできるのは IPv4 パケットだけです。

（注） シングルおよびデュアル レート ポリサーでは、複数のアクションの設定がサポートされています（シングルレート 2 カラー ポリサー （306 ページ） およびデュアルレート 3 カラー ポリサー （309 ページ） を参照）。

CLI バリエーションに関する注意事項

ここでは、CLI の複数のバリエーションが同じ結果をどのように達成できるのかを示します。

コンテキスト

時間が経ち、ソフトウェア トレインがマージされるにつれて、各種 Cisco IOS ソフトウェア リリースのバリエーションが現れました。同じソフトウェア リリース内に、3 つの同等のバリエーションが存在します。下位互換性の問題を回避するために、シスコではバリエーションの保持を決定しました。ただし、ポリシングを実装するソフトウェアは、使用する CLI バリエーションにかかわらず同一であることに注意してください。

確認

次の例では、**police** を 10 Mbps に、**conform action** を送信（デフォルト）に、**exceed action** をドロップ（デフォルト）に設定します。「大まか」に言うと、**police** コマンドには同じ結果を達成する 3 つのバリエーション（**police** *value*、**police cir** *value*、および **police rate** *value*）があります。この一連のバリエーションは、**police [cir|rate]***value*（**cir** と **rate** はオプション）と同等です。レートを 10 Mbps とすると、**police [cir|rate] 10m** コマンドを作成できます。

各バリエーションを使用して、次のようにポリシングを設定できます。

```
policy-map policer-cli-example
  class class-default
    police 10000000


policy-map policer-cli-example
  class class-default
    police cir 10m


policy-map policer-cli-example
  class class-default
    police rate 10m
```

3 つのバリエーションが同じ結果をもたらすことを確認するために、2 段階の検証を使用できます。

1. **show policy-map interface** を発行して IOS 内の設定を表示します。
2. **show platform hardware qfp active feature qos interface** を発行してハードウェアのプログラミング方法を確認します。この表示は、使用されている CLI バリエーションに関係なく同じになります。

ステップ 1 を実行した結果は次のとおりです。

```
show policy-map int GigabitEthernet1/0/0

  Service-policy input: policer-cli-example

    Class-map: class-default (match-any)
      162 packets, 9720 bytes
      5 minute offered rate 2000 bps, drop rate 0000 bps
      Match: any
      police:
          cir 10000000 bps, bc 312500 bytes
        conformed 212 packets, 12720 bytes; actions:
          transmit
        exceeded 0 packets, 0 bytes; actions:
          drop
        conformed 2000 bps, exceeded 0000 bps
```

次に、ステップ 2 を実行した結果は次のとおりです。

```
show platform hardware qfp active feature qos int g1/0/0

Interface: GigabitEthernet1/0/0, QFP interface: 12
  Direction: Input
    Hierarchy level: 0
    Policy name: policer-cli-example
      Class name: class-default, Policy name: policer-cli-example
        Police:
            cir:  10000000 bps,  bc: 315392 bytes
            pir: 0 bps,  be: 315392 bytes
            rate mode: Single Rate Mode
            conformed: 16 packets, 960 bytes; actions:
              transmit
            exceeded: 0 packets, 0 bytes; actions:
              drop
            violated: 0 packets, 0 bytes; actions:
              drop
            color aware: No
            green_qos_group: 0, yellow_qos_group: 0
```

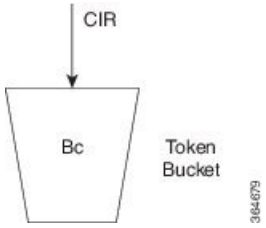
シングルレート 2 カラー ポリサー

シングルレート 2 カラー ポリサー（1R2C）は、事前に決めされたレート（bps 単位の CIR）をトラフィックが上回っているか下回っているかを判断します。さらに、どちらの場合にもアクションを実行することを可能にします。着信パケットに対して実行可能なアクションは、適合アクション（パケットのトラフィックがCIRを下回ったと見なされる場合）と超過アクション（パケットのトラフィックが CIRを超えたと見なされる場合）です。

あらゆる潜在的なバースト性を考慮に入れる必要があります。この現象は多数のパケットが同時に着信すると発生し、短い間隔では着信レートが CIR を超えますが、より長い間隔では着信レートが CIR に適合している場合があります。バーストに対応しつつ、事前に決められた CIR を長期的に適用するために、トークン バケット スキームを使用します。

このスキームを適用すると、シングルトークン バケットを使用してシングルレート 2 カラー ポリサーを示すことができます。

図 *64 :* シングルレート *2* カラー ポリサー



トークンは CIR で継続的に補充され、バケットの深さは Bc です。バケットが満杯になると、着信する追加のトークンは失われます。

パケットが着信すると、ポリサーは、その着信パケットに対応するために十分なトークン（バイト）がバケットに含まれている（パケット長に見合うバイト数がある）かどうかを評価します。含まれている場合は、パケットが適合していると見なされ、そのアクションが実行されて、適切な数のトークン（パケット長）がバケットから削除されます。

パケットが着信し、そのパケットに対応するために十分なトークンがバケットに含まれていない場合は、超過アクションが実行されます（バケット内のトークン数は変わりません）。後続のパケットが着信すると、依然として「適合」と指定されるために十分なだけのトークンがバケットが補充されていると判明する場合があります。パケットが着信しない場合、バケットはバースト制限（Bc）まで満たされた状態が継続します。

バケットの深さを指定すると、バケットに補充する時間があると仮定した、近接して着信する可能性がある適合トラフィックの許容バースト量（バイト/パケット数）が決定されます。

この例では、10 Mbps の CIR と 15000 バイトのバースト許容値を指定しました。そのため、イーサネットインターフェイスでの 10 MTU サイズのパケットのバーストが発生しても、「適合」と指定される可能性があります。

```
policy-map police-with-burst
  class class-default
    police cir 10m bc 15000
```

（注） 現在の IOS CLI では、ポリシングを複数の方法で設定して同じ結果を得ることができます。CLI バリエーションに関する注意事項 （305 ページ）の項を参照してください。

シングルレート 3 カラー ポリサー

シングルレート 3 カラー ポリサー（1R3C）は、3 つの可能な出力状態、つまり 適合、超過、および違反をサポートしています。「適合」の定義は、1R2C ポリサーでの定義（一定のバースト許容値を考慮して事前に決められたレートに準拠するトラフィック）と似ています。

その違いは、適合していないトラフィック（2 カラー ポリサーでは「超過」と指定されるトラフィック）を指定する方法にあります。このトラフィックが「超過」または「違反」と指定される、さらなるきめ細かさが導入されます。基本的に、CIR を「ごくわずかに」上回るトラ

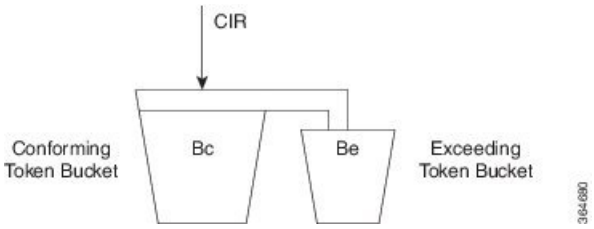
フィックは「超過」と指定されますが、CIR を上回るより持続的なバーストは「違反」と指定されます。

この動作を実現するために、2 つ目のトークン バケットが導入されます。適合トークン バケットを使用して適合トラフィックと超過トラフィックを区別することと同様に、超過トークン バケットを使用して 超過トラフィックと違反トラフィックを区別できます。

バケットのシナリオは、次のとおりです。

- 適合トークン バケットは、最初は満杯になっています。「満杯」の量は、Bc（適合バースト サイズ）として指定されているバイト数です。
- 超過トークン バケットは、最初は満杯になっています。「満杯」の量は、Be（超過バースト サイズ）として指定されているバイト数です。
- トークンが着信したときに適合トークン バケットが満杯になると（1R2C ポリサーと同様に CIR で）、それらは超過トークン バケットにオーバーフローします。
- 両方のバケットが満杯になると、それ以上のトークンは失われます。

図 *65 :* シングルレート *3* カラー ポリサー



この場合の適合バケットも、1R2C シナリオの場合と同じように動作します。着信パケットに対応するために十分なトークンがバケットに含まれている場合、そのパケットは「適合」と見なされ、適合アクションが実行され、バケットから該当する数のトークンが削除されます。超過バケットは影響を受けず、適合バケット（Bc）がCIR で引き続き補充されます。

ただし、適合バケットが満杯のときに追加のトークンが着信しても、それらはすぐには失われません。代わりに、それらは超過バケットにオーバーフローします。このバケットが満杯になると、超過したトークンは失われます。

同様に、パケットが着信したときに、そのパケットに対応するために十分なトークンが適合バケットにない場合、すぐに超過とは宣言されません。「超過」の場合と「違反」の場合があります。パケットに対応するために十分なトークンが超過バケットにある場合は、超過アクションが実行され、超過バケットから必要な数のトークンが削除されます。適合バケットから削除されるバイトはありません。

適合バケットと超過バケットのどちらにもパケットに対応するために十分なトークンがない場合、「違反」として分類され、該当するアクションが実行されます。適合バケットも超過バケットもトークンは削減されません。

適合バケットと超過バケットのどちらにもパケットに対応するために十分なトークンがない場合、「違反」として分類され、該当するアクションが実行されます。適合バケットも超過バケットもトークンは削減されません。

```
policy-map ingress-enforcement
  class af41-metering
    police cir percent 10 bc 5 ms be 10 ms
    conform-action set-dscp-transmit af41
    exceed-action set-dscp-transmit af42
    violate-action drop
```

この例では、インターフェイスの帯域幅の 10 % にトラフィックをポリシング（クラス af41 用）しており、次が適用されます。

- 5 ミリ秒までのトラフィック（**conform-action set-dscp-transmit af41**）バーストは転送され、依然として af41 としてマークされます。
- 5 ミリ秒を超えるトラフィック（**exceed-action set-dscp-transmit af42**）バーストの場合、10 ミリ秒までの追加バーストは af42 としてマークされます。ネットワークの他の場所で af42 が検出されると、合意された契約を超えてそれが受信された（ネットワークのエッジで）ことがわかります。輻輳が発生した場合は、それが最初にドロップされます。
- CIR に対して 15 ミリ秒を超えるトラフィック（**violate-action drop**）バーストは「違反」と見なされ、ただちにドロップされます。

（注） 適合バケットが満杯の場合は、超過バケットだけが補充されます。そのため、CIR を超えるレートで非バースト ストリームを送信すると、すぐに適合バケットと超過バケットの両方がオーバーフローします。超過バケットは補充されません。後続のすべてのパケットは、適合または違反と見なされます。

デュアルレート 3 カラー ポリサー

トラフィック レートは、トラフィック バーストよりも容易に理解できます。ネットワーク アドミッション制御の契約を指定する場合（トラフィック ポリシングを使用する理由 （299 ページ）を参照）、シングル レートを超える複数のバースト サイズに関して期待値を記述することが困難な場合があります。 デュアルレート 3 カラー（2R3C）ポリサーは、主にレートを使用して適合、超過、違反を区別することによって問題を簡素化します。また、セカンドレートの PIR（最大情報レート）も導入されます。

CIR と PIR には次の特性があります。

- CIR を下回るトラフィックは「適合」です。
- CIR を上回り、PIR を下回るトラフィックは「超過」です。
- PIR を上回るトラフィックは「違反」です。

これらのレートは、**police** コマンドの **cir** キーワードと **pir** キーワードを使用して指定します（詳細については、police （334 ページ） コマンドのページを参照）。

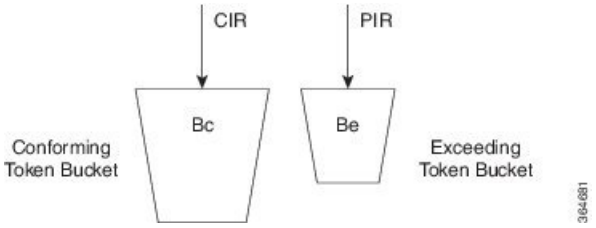
2R3C ポリサーの場合、1R3C とは異なり、パケットがポリサーに着信するたびにトークン バケットが個別に補充されます。 適合バケットをレート CIR で補充します。バケットは値 Bc まで含むことができます。 超過バケットをレート PIR で補充します。バケットは値 Be まで含むことができます。

（注） PIR は CIR を超えている必要があり、バケット間のオーバーフローは許可されません。

パケットの安定したストリームが、CIR を上回るものの PIR を下回るレートで着信した場合、すべてのパケットが「適合」または「超過」としてマークされます。1R3C ポリサーを使用する場合、このシナリオでは、わずかの数のパケットが「超過」としてマークされ、大多数のパケットが「適合」または「違反」としてマークされます。

2R3C ポリサーでは、パケットごとに 3 つの可能なアクション（適合、超過、違反）がサポートされます。デュアルレートポリサーで設定されたインターフェイスに入るトラフィックは、これらのアクション カテゴリの 1 つに分類されます。これにより、パケットの処理方法が決まります。たとえば、最も一般的な設定では、適合するパケットまたは超過するパケット（プライオリティを下げて）を送信し、違反するパケットはドロップするように設定できます。

図 *66* : デュアルレート *3* カラー ポリサー



パケットが着信すると、そのパケットに対応するために十分なトークンが適合パケットおよび超過バケットに存在するかどうかが評価されます。存在する場合は、適合アクション（通常、送信または送信とマーク）が実行され、両方のバケットからパケットを送信するのに必要なトークンが削除されます。

パケットに対応するために十分なトークンが超過トークン バケット（ただし、適合トークン バケットではなく）に含まれている場合は、超過アクション（通常、送信または送信とマーク）が実行されます。該当する数のトークンが超過バケットだけから削除されます。

どちらのバケットにもパケットに対応するために十分なトークンがない場合は、違反アクション（通常、送信、送信とマーク、またはドロップ）が実行されます。

```
policy-map ingress-enforcement
  class af41-metering
    police cir 100k bc 3000 pir 150k be 3000 conform-action set-dscp-transmit af41
exceed-action set-dscp-transmit af42 violate-action drop
```

上記の例のコードとシングルレート 3 カラー ポリサー （307 ページ）の対応するコードの違いを確認してください。

```
cir 100k bc 3000 pir 150k be 3000
cir percent 10 bc 5 ms be 10 ms
```

直前の例では、次のルールに従ってトラフィックが処理されます。

- 100 Kbps 以下の場合（最大 3,000 バイトのバーストを許可）は「適合」となり、DSCP が af41 としてマークされ、転送されます。
- 100 Kbps を超え、150 Kbps 未満の場合（この場合も 3,000 バイトのバーストを許可）は「超過」となり、DSCP が af42 としてマークされ、転送されます。
- 150 Kbs を超える場合は「違反」となり、ドロップされます。

レートおよびバースト パラメータの設定

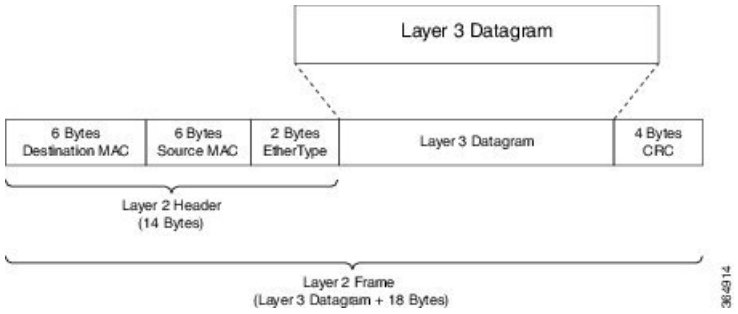
ポリサー レート計算（オーバーヘッド アカウンティング）に含まれるもの

レートまたはバースト値を指定する際は、それらの値への適合性を評価するときに、ポリサーがパケットの長さを評価する方法（以下、「ポリシング長」と呼びます）を理解している必要があります。簡単に説明すると、ポリサーには、レイヤ 3 データグラムのレイヤ 2 ヘッダー長が含まれますが CRC とパケット間オーバーヘッドはどちらも含まれません。

さらに詳しく説明するために、GigabitEthernet リンクを介して転送される IP データグラムについて検討します。

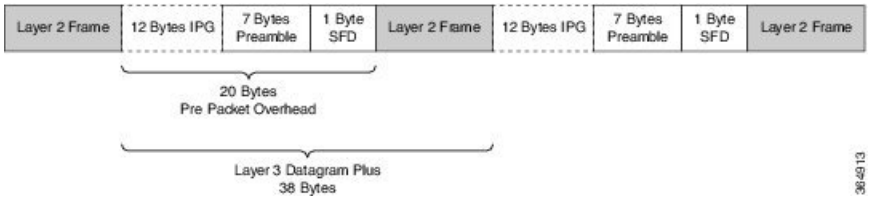
レイヤ 3 データグラム

まず、このデータグラムをイーサネット フレームにカプセル化します。これにより、各データグラム（18 バイト）に 14 バイトのレイヤ 2 ヘッダーが追加され、さらに 4 バイトの CRC が追加されます。



イーサネット オーバーヘッド

物理メディアを介してこのフレームを送信するために、イーサネットは、12 バイトのデータの送信時間に相当する最小限のパケット間ギャップ を必要とします。このパケット間ギャップ（IPG）の後に、7 バイトのプリアンブルとそれに続く 1 バイトのフレーム開始デリミタ（SFD）が必要です（イーサネット パケット間オーバーヘッド = 12 バイトの IPG + 7 バイトのプリアンブル + 1 バイトの SFD = 20 バイト）。

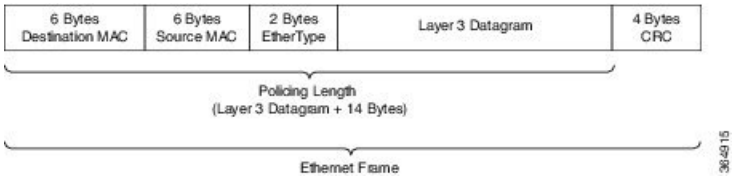


そのため、複数のイーサネットフレームを連続して送信する場合、各レイヤ3データグラムのパケットごとのオーバーヘッドにより38バイト追加されます（カプセル化の18バイト+イーサネットのパケット間オーバーヘッドの20バイト）。たええば、GigabitEthernetリンクで、100バイトのIPデータをラインレートで送信する場合、次の式を使用すると、予想されるスループット（1秒あるのパケット数）は次のようになります。

```
Linerate / Bits Per
  Byte / ( Layer 3 length + Per Packet Overhead) = Packets Per Second

1 Gbps / 8 / ( 100 + 38 ) = 905, 797 pps
```

ポリサーの観点からは、パケット長は、レイヤ3データグラム+レイヤ2ヘッダ長（GigabitEthernetインターフェイスでは14バイト）です。

ポリシング長



ここで、GigabitEthernetインターフェイスで設定された500 Mbpsのポリサーについて検討します。前の例と同様に、100バイトのIPデータグラムのすべてをポリサーに送信します。その結果、ポリシング長は100バイトのデータグラム長+14バイト（イーサネット レイヤ2ヘッダー）になります。次の式により、予想されるスループットが得られます。

```
Policer Rate / Bits
  per Byte / (Layer 3 length + Layer 2 header length) = Packets Per Second

500 Mbps / 8 / (100 + 14) = 548,246 pps
```

（注） 500 Mbpsのポリサーによって適合としてマークされるパケットは、500 Mbpsをはるかに超える物理帯域幅を消費します。

論理インターフェイス上のポリサー

出力時には、ポリサーは最終的な物理インターフェイスタイプを認識しない（トンネルはインターフェイス間を移動可能）ため、ポリサーは最終的なレイヤ2オーバーヘッドを認識しません。そのため、後者はポリシング長から除外されます。同様に、ポリサーはオーバーヘッドによるパケット拡張の程度を予測できないため、暗号化を設定する場合、ポリサー レートの計算に暗号化オーバーヘッドは含まれません。出力ポリサーには、レイヤ3データグラムとすべてのトンネル ヘッダ（追加のIP ヘッダ、GRE ヘッダなど）が含まれます。

入力時には、ポリサーが受信インターフェイス タイプを認識するため、トンネル インターフェイスでのポリシングには、レイヤ 2 オーバーヘッドとすべてのトンネル ヘッダーが含まれます。

次の表に、 Cisco ASR 1000 シリーズ アグリゲーション サービス ルータでのポリサー レート計算の依存関係を示します。すべての順列のサブセットだけが示されていることに注意してください。

表 *22 : GRE/QoS* ポリシー長の計算

GRE/QoS ポリシー長を計算する QFP 式 **（イーサネット インターフェイスの例）**			
トンネル タイプ	**ポリシング**		**キューイング**
	入力	出力	出力
ip_gre	レイヤ 3 + **24** +**14**	レイヤ 3 + 24	レイヤ 3 + 24 +14
ip_ip	レイヤ 3 + **20** +14	レイヤ 3 + 20	レイヤ 3 + 20 +14
ipsec	レイヤ 3 + 24 + 14	レイヤ 3 + 24	レイヤ 3 + 24 +14 + crypto_oh
ipsec ipv4	レイヤ 3 + 14	レイヤ 3 + **0**	レイヤ3 + 14 + crypto_oh

ここで、値は次のように定義されます。

- 0：svti（「トンネルモード ipsec ipv4」）にはオーバーヘッドがない
- 14：レイヤ 2 イーサネット ヘッダーのサイズ
- 20：IP/IP ヘッダーのサイズ
- 24：IP/GRE ヘッダーのサイズ（20 + 4）

ATM インターフェイス上のポリサー

ATM インターフェイスでポリサーが設定されている場合、ポリシング長にはレイヤ 3 データグラムと ATM アダプテーション層（AAL）ヘッダーが含まれます。AAL5SNAP カプセル化の長さの場合、これは、ポリシング長に 8 バイトのヘッダーを含めることを意味します。AAL5NLPIDカプセル化の場合は 2バイトです。

この計算は、完全な *AAL PDU* と セル タックスを含むスケジューリングに適用される計算とは明確に異なります。

含まれるものの変更：オーバーヘッド アカウンティングの調整

前のセクションでは、ポリサー レートの計算にデフォルトで含まれるものについて説明しました。ただし、デフォルトから逸脱させる必要がある場合もあります。たとえば、CIRをリンクで消費される物理帯域幅として表現する必要がある場合があります。イーサネット インターフェイスの場合は、4バイトのCRCと、パケットごとに必要な20バイトのパケット間オーバーヘッドが含まれます。

あるいは、サービスプロバイダーがレイヤ3レートで顧客のトラフィックをポリシングする必要がある場合もあります。パケットがさまざまなインターフェイスタイプ（またはカプセル化プロトコル）を通過するときにデータグラムの長さは変わらないため、ポリサー レートの計算にレイヤ 2 ヘッダーの長さは含まれません。

（注） QoS ポリシーをサポートするインターフェイスは、いずれもオーバーヘッド アカウンティングの調整をサポートします。

（注） オーバーヘッド アカウンティングの変更はネットワークに影響を与える可能性があります。たとえば、ネットワーク アドミッション制御にポリサーを使用するときは、そのネットワークに接続する機器で対応するシェーパーを設定する必要がある場合があります。CIRに含まれるものの 2 つのビュー（シェーパーとポリサー）は一致する必要があります。

次の例では、ポリサーが物理リンク上のトラフィックの最大 50% を適合させることができるように、すべてのパケット間オーバーヘッドを含めます。1 パケットあたり 24 バイトを追加（**user-defined 24**）することにより、4 バイトの CRC と 20 バイトのパケット間オーバーヘッドに対処します。

```
policy-map ethernet-physical-example
  class class-default
    police cir percent 50 account user-defined 24
```

police account コマンドの **atm** キーワードを使用すると、 レート計算で ATM のセル分割とセル パディング（ATM セル タックス）を補正するようにポリサーに指示できます。

セル タックスを含め、AAl5 トレーラに対応するために、ルータは最初にポリシング長に 8 バイトを追加します。次に、パケットの伝送に必要な ATM セル数を計算し（53 バイトのセルあたり 48 バイトのデータ伝送）、その数に 53 を掛けます。たとえば、46 バイトのデータグラムには 2 つのセルが必要なので、セル タックスが含まれている場合、ポリシング長は「106 バイト」と見なされます。

次の例では、レート計算にセル タックスを含める必要がある 5 Mbps ポリサーを示します。

```
policy-map include-cell-tax-example
  class class-default
    police cir 5000000 account user-defined 0 atm
```

設定内の **atm** は、セル タックスを含めるように指示します。

オーバーヘッド アカウンティングの調整の制約事項

- 子ポリシーでオーバーヘッドアカウンティングをイネーブルにする場合は、親ポリシーでオーバーヘッド アカウンティングをイネーブルにする必要があります。
- ポリシー マップで、ポリシーのすべてのクラスに対してオーバーヘッド アカウンティングをイネーブルまたはディセーブルにする必要があります。同じポリシー内で、一部のクラスに対してオーバーヘッドアカウンティングをイネーブルにし、残りのクラスに対してオーバーヘッド アカウンティングをディセーブルにすることはできません。
- オーバーヘッド アカウンティングは、QoS カウンタ（分類、ポリシング、キューイングなど）のいずれにも反映されません。
- 最上位親ポリシーでオーバーヘッドアカウンティングをイネーブルにできるだけでなく、中位子ポリシーと最下位子ポリシーの両方でオーバーヘッドアカウンティングをイネーブルにできます。子ポリシーは、親レベルまたは親の親レベルで設定されたオーバーヘッド アカウンティング ポリシーを継承します。
- ポリシー マップ内、および（階層型ポリシー マップ構造の）親ポリシー マップと子ポリシー マップの間では、使用されるオーバーヘッド アカウンティングのタイプまたは値が一貫している必要があります。

オーバーヘッド アカウンティングの調整（事前定義オプション）

いくつかの事前定義CLIオプション（ブロードバンドユースケースに基づく）によって、ルータが該当するバイト数を増減する間のカプセル化を指定できます（次の表を参照）。

ネットワークの他の場所にある DSLAM（デジタル加入者線アクセス マルチプレクサ）に、イーサネット インターフェイス上のトラフィックを送信（または受信）したとします。その際、イーサネット フレーム（Dot1Q や Q-in-Q など）でカプセル化しますが、DSLAM は何らかの形式の ATM カプセル化でカプセル化します。そのため、DSLAM の後に現われるトラフィックでポリサーを実行する必要があります。すべての場合において、ポリシング長にセルタックスを追加します。

表 *23 :* オーバーヘッド アカウンティング調整に事前に定義されたオプションの表

CLI	値（dot1q/qinq）	ATM	詳細（dot1q/qinq）
account dot1q|qinq aal5 mux-1483routed	-15/-19	○	dot1q: 3 byte 1483 routed - 18 byte dot1q qinq: 3 byte 1483 routed - 22 byte qinq
account dot1q|qinq aal5 mux-dot1q-rbe	0/-4	○	dot1q: 0 byte mux_rbe + 18 byte dot1q - 18 byte dot1q qinq: 0 byte mux_rbe + 18 byte dot1q - 22 byte qinq

CLI	値（dot1q/qinq）	ATM	詳細（dot1q/qinq）
account dot1q|qinq aal5 mux-pppoa	-22/-26	○	dot1q: 2 byte mux_pppoa - 6 byte pppoe - 18 byte dot1q qinq: 2 byte mux_pppoa - 6 byte pppoe - 22 byte dot1q
account dot1q|qinq aal5 mux-rbe	-4/-8	○	dot1q: 0 byte mux_rbe + 14 byte 802.3 - 18 byte dot1q qinq: 0 byte mux_rbe + 14 byte 802.3 - 22 byte qinq
account dot1q|qinq aal5 snap-1483routed	-12/-16	○	dot1q: 6 byte snap 1483 routed - 18 byte dot1q qinq: 6 byte snap 1483 routed - 22 byte qinq
account dot1q|qinq aal5 snap-dot1q-rbe	10/6	○	dot1q: 10 byte snap_rbe + 18 byte dot1q - 18 byte dot1q qinq: 10 byte snap_rbe + 18 byte dot1q - 22 byte qinq
account dot1q|qinq aal5 snap-pppoa	-20/-24	○	dot1q: 4 byte snap_pppoa - 6 byte pppoe - 18 byte dot1q qinq: 4 byte snap_pppoa - 6 byte pppoe - 22 byte qinq
account dot1q|qinq aal5 snap-rbe	6/2	○	dot1q: 10 byte snap_rbe + 14 byte 802.3 - 18 byte dot1q qinq: 10 byte snap_rbe + 14 byte 802.3 - 22 byte qinq
account user-defined <value>	<value>	x	
account user-defined <value> atm	<value>	○	

次の例では、事前定義されたオーバーヘッドアカウンティング値を適用します。イーサネットインターフェイスで Dot1Q カプセル化パケットを受信する場合、アップストリーム DSLAM は、 *AAL5-Mux 1483* ルーテッド カプセル化パケットを受信 してから ATM を削除し、イーサネット ヘッダーを追加します。ATM インターフェイスでは、データグラムに 3 バイトの追加 AAL ヘッダーがありますが、18 バイトのイーサネット ヘッダー（Dot1Q を含む）はありません。そのため、ATM インターフェイスでは PDU は 15 バイト少なくなります（ポリシング長から 15 バイトを引いてセル タックスを追加します）。

```
policy-map atm-example
  class class-default
    police 5000000 account dot1q aal5 mux-1483routed
```

デフォルトのバースト サイズ

バースト許容値（Bc または Be）を明示的に設定していない場合は、IOS によってデフォルトが設定されます。このデフォルトのバースト許容値は、該当するレートに基づく 250 ミリ秒のデータ量です。たとえば、CIR が 100 Mbps の場合、このレートの 250 ミリ秒のデータ量は 100000000/8 x 0.250 = 3125000 バイトになります。

シングルレート ポリサーの Bc と Be は、 常に CIR に基づきます。デュアルレート ポリサーの Be は、PIR に基づきます。

（注） キュー アドミッション制御用にポリサーを設定する場合は （トラフィック ポリシングを使用する理由 （299ページ）を参照）、Bc をそのキュー内のアプリケーションに適したものに設定します（たとえば、音声アプリケーションの場合は、Bc を 10 ミリ秒以下に設定します）。

```
policy-map policer-default
  class af41
    police cir 20000000 pir 40000000 conform-action transmit  exceed-action
                \ set-dscp-transmit af42 violate-action set-dscp-transmit af43

 show policy-map interface
 GigabitEthernet1/0/0

  Service-policy input: policer-default

    Class-map: af41 (match-all)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match:  dscp af41 (34)
      police:
          cir 20000000 bps, bc 625000 bytes           1
          pir 40000000 bps, be 1250000 bytes          2
        conformed 0 packets, 0 bytes; actions:
          transmit
        exceeded 0 packets, 0 bytes; actions:
          set-dscp-transmit af42
        violated 0 packets, 0 bytes; actions:
          set-dscp-transmit af43
        conformed 0000 bps, exceeded 0000 bps, violated 0000 bps
```

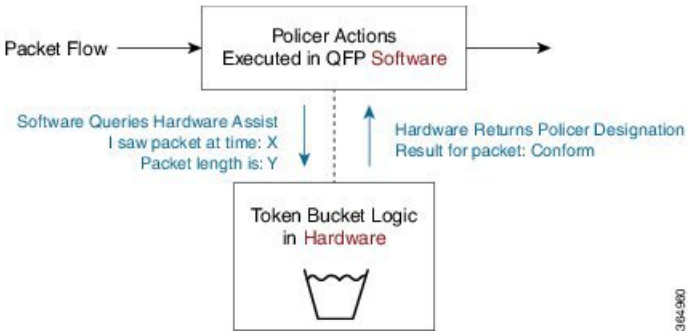
（注） デュアルレート ポリサーと Bc および Be は、それぞれ CIR（**1**）と PIR（**2**）に基づいて 250 ミリ秒にデフォルト設定されます。

ハードウェアでプログラムされたレートとバースト サイズ

Cisco ASR 1000 ルータ プラットフォームでは、ポリサー レートの計算は専用ハードウェアで実行されます。

ハードウェア アシストを使用すると、パフォーマンスへの影響とは無関係にポリサーの数を拡張できますが、プログラム可能なレートとバースト値の組み合わせに関するいくつかの制約事項があります。

図 *67 :*



シングルレート 2 カラー ポリサーを使用した簡単なポリシーの場合について検討します。

```
policy-map hardware-example
  class class-default
    police cir 1m bc 3000
```

show policy-map interface コマンドからの出力により、設定された CIR および Bc 値を IOS が受け入れたことを確認できます。

```
show policy-map interface g1/0/0


GigabitEthernet1/0/0

  Service-policy input: hardware-example

    Class-map: class-default (match-any)
      337 packets, 167152 bytes
      5 minute offered rate 2000 bps, drop rate 0000 bps
      Match: any
      police:
          cir 1000000 bps, bc 3000 bytes                    *
        conformed 337 packets, 167152 bytes; actions:
          transmit
        exceeded 0 packets, 0 bytes; actions:
          drop
        conformed 2000 bps, exceeded 0000 bps
```

***** 予期どおりに設定された CIR と Bc

ただし、データプレーンを見ると、ハードウェアで実際にプログラムされている値を確認できます。次に、ハードウェアの実際のポリサー値を表示する **show platform qfp active feature qos interface** コマンドの出力を示します。

```
show platform hardware qfp active feature qos interface gig1/0/0


Interface: GigabitEthernet1/0/0, QFP interface: 9

  Direction: Input
```

```
Hierarchy level: 0
Policy name: hardware-example
  Class name: class-default, Policy name: hardware-example
    Police:
        cir:  1000000 bps,  bc: 3264 bytes              *
        pir: 0 bps,  be: 3008 bytes
        rate mode: Single Rate Mode
        conformed: 19 packets, 9424 bytes; actions:
          transmit
        exceeded: 0 packets, 0 bytes; actions:
          drop
        violated: 0 packets, 0 bytes; actions:
          drop
        color aware: No
        green_qos_group: 0, yellow_qos_group: 0
```

* ハードウェア アシスト用に修正された Bc

（注） ハードウェアアシストに対応するためにレートとバーストのパラメータを多少変更する場合がありますが、プラットフォームは常にレートとその結果の精度を要求される値の 1% 以内に維持することを目的とします。

パーセントベースのポリサー

パーセントベース ポリシング機能を使用すると、インターフェイスで使用可能な帯域幅のパーセンテージに基づいてトラフィック ポリシングを設定できます。そのため、帯域幅の量が異なる複数のインターフェイス タイプに同じポリシー マップを使用できます。インターフェイスごとに帯域幅を再計算したり、インターフェイスのタイプごとに異なるポリシーマップを設定する必要はありません。

（注） インターフェイスがシェープド ATM 相手先固定接続（PVC）の場合、総帯域幅は次のように計算されます。

- 可変ビット レート（VBR）の仮想回線（VC）の場合は平均セル レート（SCR）が使用されます。
- 使用可能ビット レート（ABR）の VC の場合は最小セル レート（MCR）が使用されます。

CIR と PIR の両方にパーセンテージベースのポリサーを使用できます。その際、インターフェイス帯域幅または親シェーパー（存在する場合）のいずれかの指定されたパーセンテージから計算します。

パーセントベースのポリシングでは、バースト パラメータ（Bc と Be）を指定する場合、バイト単位ではなくミリ秒単位で指定する必要があります。ターゲット インターフェイスの速度を提示すると、IOS は 2 つのステップで値をバイトに変換します。

1. IOS が、ターゲット インターフェイスの速度を使用して、パーセンテージを bps CIR に変換します。
2. bps CIR と 時間内のバーストによって、バーストがバイトに変換されます。

Bc を 10 ミリ秒（ポリシング レートに応じた値）に設定し、CIR を利用可能なインターフェイス帯域幅の 10 % に設定します。

```
policy-map police-percent
  class class-default
    police cir percent 10 bc 10 ms
```

police-percent を GigabitEthernet インターフェイス（1 Gbps の公称帯域幅）に適用すると、IOS は、CIR を 100 Mbps に変換し、Bc を 125,000 バイト（100 Mbps x 10 ミリ秒/8）に変換します。

```
show policy-map interface GigabitEthernet1/0/0

  Service-policy input: police-percent

    Class-map: class-default (match-any)
      834 packets, 413664 bytes
      5 minute offered rate 13000 bps, drop rate 0000 bps
      Match: any
      police:
        cir 10 %, bc 10
        cir 100000000 bps, bc 125000 bytes  Configured CIR and Bc converted to bps and
bytes, respectively.
```

ここで、police-percent を POS OC3 インターフェイスに適用すると、レートは 155 Mbps の公称帯域幅に基づいたものになります。CIR は 15.5 Mbps と計算され、Bc は 19375 バイト（15.5 Mbps x 10 ミリ秒/8）と計算されます。

```
show policy-map interface POS1/1/0

  Service-policy input: police-percent

    Class-map: class-default (match-any)
      0 packets, 0 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
      police:
          cir 10 %, bc 10
          cir 15500000 bps, bc 19375 bytes  Configured CIR and Bc converted to bps and
bytes, respectively.
```

カラー対応ポリサー

カラー対応ポリサーは、前のノードのポリサーによって契約内または契約外と判断された既存のマーキングを詳しく調べます。（前のノードは、通常、ネットワークのエッジにあります）。カラー対応ポリサーが設定されている場合は、そのようなマーキングを使用してパケットの適切なポリシングアクションが決定されます。契約外として指定されたトラフィックは、常に契約外のままになります。契約内として指定されたトラフィックは、この新しいポリサーによって契約外に降格される場合があります。

ASR 1000 では、カラー対応ポリシングが限定的に実装されます。既存のトラフィックのカラーを判断するために使用されるクラス マップの内容が制限されます。

- カラー対応クラス マップでは、QoS グループの照合だけがサポートされています（QoS グループに基づく分類だけがサポートされています）。
- カラー対応クラスごとに 1 つのフィルタ（1 つの**match qos-group** 値ステートメント）だけがサポートされています。

  子ポリシーを使用して、受信したパケット内の必要なフィールドに基づいて QoS グループを設定できます。
- カラー対応「固有」の統計は、 サポートされていません。
- カラー対応ポリサーで参照されているカラー対応マップを **no class-map** コマンドで削除することはできません。まず、すべてのカラー対応ポリサーを削除する必要があります（**no conform-color** コマンドまたは **no exceed-color** コマンドを使用）。

カラー対応ポリシングの「カラー」は、受信したパケット内の既存マーキングを解釈する方法をポリサーに示す仕組みを指します。一般に、適合または契約内として事前にマークされたトラフィックを表すには、緑色を使用します。同様に、超過または契約外として事前にマークされたトラフィックを表すには、黄色を使用します。

緑色または黄色は代表的なものにすぎないことに注意してください。 CLI では代わりに *conform-color* と *exceed-color* が 使用されます。**police** コマンド全体を通じて、これらのキーワードにより、そのパケットの既存のカラーを判別するために使用されるクラスマップが指定されます。

次の例は、子ポリシーマップによって、受信したパケットの任意のフィールドに基づいて既存のカラーを指定する方法を示しています。このカラー対応ポリサーは、DSCP 相対的優先転送トラフィック クラスの 1 つである AF4 からのすべてのパケットに一致するクラスで設定されます。

この例では、AF41 とマークされたパケットは契約内（適合または緑色）、AF42 は契約外（超過または黄色）、AF43 は違反となります。子ポリシー mark-existing-color は、受信した DSCP に基づいてパケットを分類し、内部的に AF41 パケットを qos-group 1、AF42 パケットを qos-group 2 としてマークします。

カラー対応ポリサーは、pre-conform（緑色のパケットを分類）クラス マップと pre-exceed（黄色のパケットを分類）クラスマップを使用して、着信パケットの既存のカラーを判別します。これらのクラス マップは QoS フィルタしかサポートしていませんが、子ポリシーを使用すると、受信したパケットの DSCP 値に基づいて既存のカラーを判別できます。

```
class-map af4
  match dscp af41 af42 af43
!
class-map af41
  match dscp af41
class-map af42
  match dscp af42
!
class-map pre-conform                !These are policer
  match qos-group 1                  !class-maps that
```

```
class-map pre-exceed                  !only support qos-group
  match qos-group 2
!
policy-map mark-existing-color        !We use a child policy
  class af41                          !to set qos-group
    set qos-group 1                   !based on DSCP in the
  class af42                          !received packet
    set qos-group 2
!
policy-map dual-rate-color-aware
  class af4
    police cir 1m bc 5000 pir 2m be 5000
      conform-action set-dscp-transmit af41
      exceed-action set-dscp-transmit af42
      violate-action drop
      conform-color pre-conform exceed-color pre-exceed
    service-policy mark-existing-color
```

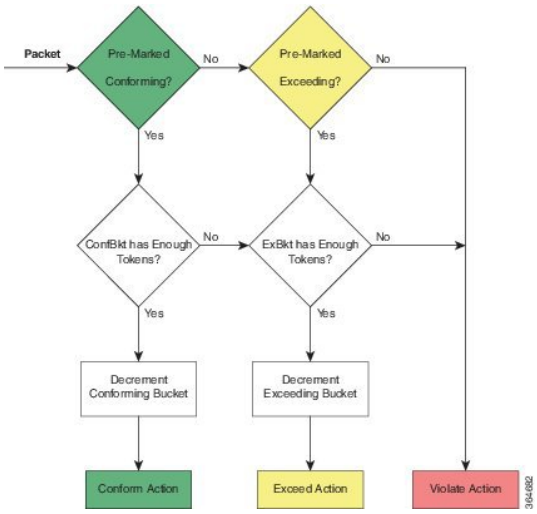
シングルレート カラー対応 3 カラー ポリサー

シングルレート3カラーポリサーのカラー対応モードにより、標準のシングルレート3カラーポリサーが拡張されます（シングルレート 3 カラー ポリサー （307 ページ） を参照）。

このタイプのポリサーの「カラーブラインド」バージョンと同様に、「カラー対応」モード用の2つの異なるトークンバケットが維持および補充されます。違いは、これらのバケットに対してパケットを評価する方法にあります。カラー対応ポリサーでは、前のルータによる決定（パケットの現在の指定）が尊重され、前のルータの決定が元に戻されないことが保証されていました（超過パケットまたは違反パケットが適合パケットに昇格されることはありませんでした）。

次のフローチャートは、シングルレート カラー対応トラフィック ポリシングでトラフィックを処理するために使用されるアルゴリズムを示しています。「ConfBkt」は 適合トークン バケットを表し、「ExBkt」は超過トークン バケットを表します。

図 *68 :* シングルレート カラー対応 *3* カラー ポリサー



パケットが着信すると、ポリサーはそのカラー対応クラスマップを使用して、そのパケットの既存のカラーを判別します。このカラーは、適合（conform-color クラス マップに一致）、超過（exceed-color クラス マップに一致）、違反（これらのクラス マップのいずれにも一致しない）のいずれかです。

パケットが適合として事前にマークされている場合、最終的に適合、超過、または違反になる可能性があります。評価は、ポリサーがカラーブラインドモードで動作しているかのように進められます。

- 適合トークン バケットに十分なトークンがある場合、パケットは適合アクションを実行し、バケットのトークンがパケットのサイズだけ減少します。
- 適合トークン バケットには十分なトークンがないものの、超過トークン バケットにはある場合、パケットは超過アクションを実行し、超過トークンバケットのトークンがパケットのサイズだけ減少します。
- 適合トークンバケットと超過トークンバケットのどちらにも十分なトークンがない場合、パケットは違反アクションを実行します。

パケットが超過として事前にマークされている場合、適合に昇格されることはないため、適合トークン バケットの評価は不要です。

- 超過トークン バケットに十分なトークンがある場合、パケットは超過アクションを実行し、バケットのトークンがパケットのサイズだけ減少します。

パケットが違反として事前にマークされている場合

- 違反アクションが実行され、トークン バケットは変更されません。

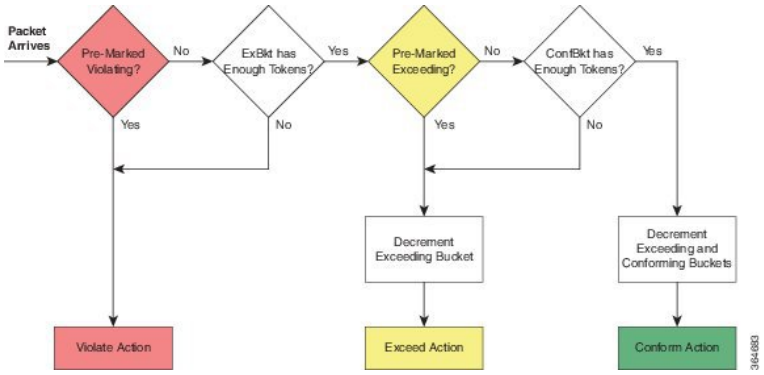
デュアルレート カラー対応 3 カラー ポリサー

デュアルレート 3 カラー ポリサーのカラー対応モードにより、標準のデュアルレート 3 カラー ポリサーが拡張されます （デュアルレート 3 カラー ポリサー （309 ページ）を参照）。

このタイプのポリサーの「カラーブラインド」バージョンと同様に、「カラー対応」モード用の 2 つの異なるトークン バケットが維持および補充されます。違いは、これらのバケットに対してパケットを評価する方法にあります。カラー対応ポリサーでは、前のルータによる決定（パケットの現在の指定）が尊重され、前のルータの決定が元に戻されないことが保証されていました（超過パケットまたは違反パケットが適合パケットに昇格されることはありませんでした）。

次の図は、デュアルレート カラー対応ポリシングでトラフィックを処理するために使用されるアルゴリズムを示しています。「ConfBkt」は 適合トークン バケットを表し、「ExBkt」は超過トークン バケットを表します。

図 *69 :* デュアルレート カラー対応 *3* カラー ポリサー



パケットが着信すると、ポリサーはそのカラー対応クラスマップを使用して、そのパケットの既存のカラーを判別します。このカラーは、適合（conform-color クラス マップに一致）、超過（exceed-color クラス マップに一致）、違反（これらのクラス マップのいずれにも一致しない）のいずれかです。

パケットが違反として事前にマークされている場合

- 違反アクションが実行され、どちらのバケットも変更されません（トークンが減少しません）。

パケットが超過として事前にマークされている場合

- 超過バケットに十分なトークンがあるときは、パケットは超過のままになり、超過バケットのトークンがパケットのサイズだけ減少します。
- 超過バケットに十分なトークンがないときは、パケットが違反アクションを実行し、どちらのバケットも変更されません。

パケットが適合として事前にマークされている場合

- 超過バケットに十分なトークンがないときは、パケットが違反アクションを実行し、どちらのバケットも変更されません。
- 超過バケットに十分なトークンがあるものの、適合バケットにないときは、パケットは超過アクションを実行し、超過バケットのトークンがパケットのサイズだけ減少します。
- 超過バケットと適合バケットの両方に十分なトークンがあるときは、適合アクションが実行され、両方のバケットのトークンがパケットのサイズだけ減少します。

ポリサーを含む階層型ポリシー

階層型トラフィック ポリシングでは、トラフィック クラスをよりきめ細かく制御し、いくつかの QoS アクションを多数のそれらのクラスの集合に対して実行する方法として、階層型ポリシーが導入されています。

ASR 1000 シリーズは階層型ポリシーで最大 3 つのレベルをサポートします。そのポリシーの任意のレベルで、ポリシング機能（1 つの特定 QoS アクション）を設定できます。

階層型ポリシーの説明では、場合により、その階層内の異なるレベルを説明するために、さまざまな用語が使用されます（上/中/下、親/子/孫、ルート/リーフ、子/親/祖父母）。これではあいまいさが生まれるため、ここではレベルを常に「親/子/孫」と呼び、その意味を次のように定義します。

親ポリシーは、**service-policy** コマンドを使用してインターフェイスに適用されるポリシーマップです。
子ポリシーは、親ポリシーのクラスに直接埋め込まれる（クラス内で **service-policy** コマンドを使用）ポリシーです。
孫ポリシーは、子ポリシーのクラスに直接埋め込まれるポリシーです。

場合によっては、「ポリシーの子」または「ポリシーの親」という表現を使用します。これらは相対的な用語です（たとえば、「孫ポリシーの親」は、絶対的な用語を使用する場合の「子ポリシー」を指します）。

ポリサーだけを含む入力階層型ポリシー

階層型ポリシーでのポリサーの最も単純でおそらく最も一般的な使用方法の一つは、ポリサーだけを含む入力ポリシーです。ポリサーが多くの場合にネットワーク アドミッション制御のためにどのように使用されるのかについては、すでに説明しました（トラフィック ポリシングを使用する理由（299 ページ）で定義されています）。単純なポリサーを階層型ポリサーに置き換えると、ネットワーク オペレータは、ネットワーク アドミッションの集約レートを設定できるだけでなく、ネットワーク上で伝送されるトラフィックの個別のクラスのレートも指定できます。

たとえば、次のようなポリシーがあるとします。

```
policy-map child
  class voice
    police cir percent 10 bc 5 ms
 !
policy-map parent
  class class-default
    police cir 50000000
    service-policy child
```

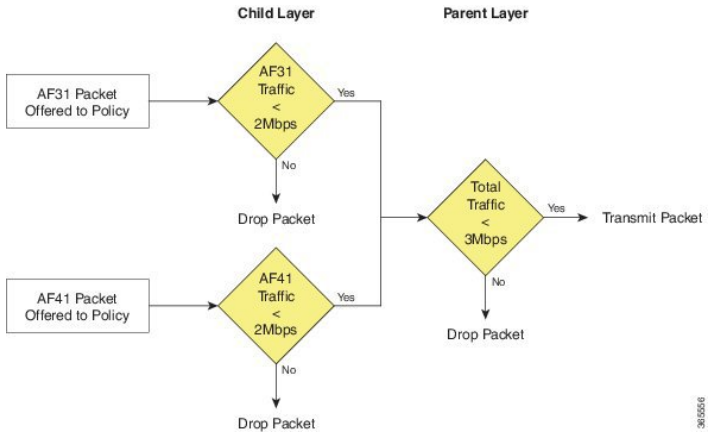
インターフェイスに適用されるポリシーマップ parent は集約ネットワーク アドミッション レート（またはサービス レート）を定義し、顧客はこれに基づいて接続されます。この例では、顧客は 50 Mbps のネットワーク サービスを契約しています。そのネットワーク レート内で、子ポリシーが個別のトラフィック クラスを制限します。たとえば、音声クラスでは、5 Mbps（親の 10%）を超えるレートで着信したトラフィックが単純にドロップされるように指定されます。

階層型ポリサーの動作順序

ASR 1000 シリーズアグリゲーション サービス ルータでは、最初に子について、次に親について、階層型ポリサーが評価されます。このスキームは従来の IOS とは異なりますが、はるかに有用な構成概念が提供されます。次の例により、この概念が明確に示されます。

```
policy-map child
  class AF41
    police 2m
  class AF31
    police 2m
!
policy-map parent
  class AF41_or_AF31
    police 3m
    service-policy child
```

図 *70 :* 階層型ポリシング



AF31 とマークされたパケットが着信する場合、このパケットは最初に子ポリシーの AF31 ポリサーを通過する必要があります。これにより、そのようなパケットは最大 2 Mbps のレートで通過できます。その後、このパケットは親ポリサーを通過する必要があります。親ポリサーは、AF31 トラフィックと AF41 トラフィックの両方をモニタします。

子ポリシーからの AF31 トラフィックと AF41 トラフィックの合計レートは、それぞれ 2 Mbps のポリサーが設定されているため、最大 4 Mbps になります。パケットは、子ポリサーを通過しても、親ポリサーでの着信レートが設定された 3 Mbps レートを超えている場合、親ポリサーによってドロップされる可能性があります。

ポリサーが、スケジューリング ポリシー（帯域幅/シェープ/プライオリティ）を持つ出力ポリシーマップで使用される場合、すべてのポリサーが、パケットがキューに入れられる前に評価されます。さらに、親レベルのポリサーは、子レベルのシェープ値の前に適用されます（スケジューリングが行われます）。

階層型ポリシングにおけるパーセントベースのポリサー

パーセントベースのポリサーがポリシーマップの親レベルで使用される場合、そのパーセントの意味は非常に直感的なもの（ポリシーマップが適用されているインターフェイスで利用可能な帯域幅のパーセント）になります。子レベルまたは孫レベルでパーセントベースのポリサーが使用される場合は、その意味がもう少しあいまいになる可能性があります。

パーセントベースのポリサーが子レベルで設定されている場合、ポリサーは、親レベルのクラスを調べて、そのクラスの帯域幅がシェーパーとポリサーのどちらによって制限されているかどうかを評価します。制限されている場合、子ポリサーの CIR は、親レベルで設定されたシェープまたはポリシング レートのパーセントです。制限されていない場合、パーセントは、ポリシー マップが適用されているインターフェイスで利用可能な帯域幅のパーセントと解釈されます。

パーセントベースのポリサーが孫レベルで設定されている場合、ポリサーは、最初に子レベル クラスでシェーパーまたはポリサーを探します。見つかった場合は、子レベルのそのレートを使用します。存在しない場合は、孫ポリサーが親レベルでそのクラスを探します。レートが見つからない場合は、ポリシーが適用されているインターフェイスのレートを使用します。

パーセントポリサーが孫レベルで設定されており、レート制限機能（シェーパーやポリサーなど）が子レベルと親レベルの両方で設定されている場合、孫は常に子レベルで設定されたレートを使用します。ポリシー内の任意のレベルのシェーパーまたはポリサーの合計が、利用可能な物理帯域幅より大きくなる可能性があるため、これは非常に重要です。

シェーパーとポリサーの両方が、パーセントベースのポリサーを持つクラスの親で設定されている場合、そのパーセントベースのポリサーは、設定されている低い方のレート（シェーパーまたはポリサー）に基づきます。

次の階層型ポリシー マップは、これらの考慮事項を示しています。

```
policy-map grandchild
  class AF11
    police cir percent 60
  class AF12
    police cir percent 40
!
policy-map child
  class AF1
    bandwidth percent 50
    service-policy grandchild
!
policy-map parent
  class class-default
    shape average 50000000
    service-policy child
```

孫ポリシーのポリサーは、パーセントベースのポリサーです。これらのポリサーは、レート制限機能に関して親クラス（親ポリシーのクラス AF1）に従います。ここには何も存在しないため、ポリサーは親クラスに「ステップアップ」します（**class class-default**、親ポリシー内で）。

そこには、スループットを 50Mbps に制限するシェーパーがあります。そのため、クラス AF11 のポリサーは、30 Mbps（50 Mbps の 60%）の CIRで設定されます。 クラス AF12 のポリサーは、20 Mbps の CIR で設定されます。

ポリシング機能の設定と動作の確認

すべての MQC QoS 機能と同様に、ポリシング機能の設定とパフォーマンスは 3 つの方法で確認できます。

- **show policy-map** *policy-name*

  ユーザが入力した設定が表示されます。ルータ上の実行コンフィギュレーションの内容に似ていますが、設定内で明示的に呼び出されないデフォルト値とアクションが表示されます。

- **show policy-map interface** *interface-name*

  そのポリシー マップ内のすべての機能に関する統計が表示されます。QoS ポリシーが予期どおりに動作していることを確認するための主要な手段です。

- **show platform hardware qfp active feature qos interface** *interface-name*

  データプレーンからのリアルタイム情報が表示されます。ハードウェアでプログラムされている正確なレートとバースト サイズが示されます。

例 1：show policy-map *policy-name* コマンド

ポリシー マップ simple_policer を次のように設定するとします。

```
policy-map simple_policer
  class AF1
    police cir 20000000
```

show policy-map コマンドの出力は次のようになります。

```
show policy-map simple_policer

Policy Map simple_policer
  Class AF1
    police cir 20000000 bc 625000
      conform-action transmit
      exceed-action drop
```

明示的な適合バーストおよび適合（または超過）アクションのほかに、統計またはインターフェイス情報の欠如に注意してください。単に、ポリシー（複数のインターフェイスに適用可能なアクション）を定義します。

例 2：show policy-map interface *interface-name* コマンド

次に、特定のインターフェイスに適用されるポリシー マップのインスタンスに関する **show policy-map interface** コマンドの出力例を示します。

```
show policy-map interface GigabitEthernet1/0/0

GigabitEthernet1/0/0

 Service-policy input: simpler_policer

  Class-map: AF1 (match-any)                                --+
   1000 packets, 1496000 bytes                                |Classification
   5 minute offered rate 0000bps, drop rate 0000bps           |Section
   Match: :dscp af11 (10) af12 (12) af13 (14)                 |
   police:                                                  --+
     cir 20000000 bps, bc 625000 bytes                        |
    conformed 447 packets, 668712 bytes; actions:             |Policing
     transmit                                                 |Section
    exceeded 553 packets, 827288 bytes; actions:              |
     drop                                                     |
    conformed 0000 bps, exceeded 0000 bps                   --+

  Class-map: class-default (match-any)
    0 packets, 0 bytes
    5 minute offered rate 0000 bps, drop rate 0000 bps
    Match: any
```

この出力の編成は、 ポリシー マップおよびクラス階層を表す階層型出力と組み合わされたポリシー マップ定義を反映しています。各クラス内で、分類セクションに分類カウンタ （このクラスに属すると判断されたパケットの統計）と分類基準（このクラスに属するパケットを定義するクラスマップの要約）が表示されます。分類セクションの後に、そのクラス内で設定されている各 QoS アクションを表すブロックがあります。この例ではポリシングが唯一のアクションであるため、ポリシング統計のブロックだけが表示されます。

例に見られるように、出力には統計とともに設定の要約が示されます。ルータは統計（長期）使用してレートを計算し、それらを表示します。これらの計算結果に含まれるレートは、減衰平均（レート）を表します。計算の頻度（デフォルトでは300秒）は、そのインターフェイスのロード間隔に依存します。**clear counters** コマンドを発行しないかぎり、**show policy-map interface** の出力の統計は保持されます。データプレーンは 10 秒ごとに統計を更新します（例 3：show platform hardware qfp active feature qos interface コマンド （331 ページ） を参照）。

分類とポリサーアクションの統計はデータプレーン内のさまざまなエンティティから発生することに注意してください。その結果、それらは、時間がわずかに異なるだけで更新される可能性があります（簡単に言うと、アクションカウンタが分類カウンタを超える可能性があります）。

次の表に、**show** コマンドの出力に含まれるさまざまなフィールドの意味をまとめます。

表 *24 : show policy-map interface* コマンド出力のフィールド

セクション	フィールド	意味
	Service-policy input（または output） :	指定したインターフェイスに適用されたポリシーマップの名前と、それが入力（または出力）トラフィック上で動作するかどうか

例 2：**show policy-map interface interface-name** コマンド

セクション	フィールド	意味
Classification セクションの先頭部	Class-map：	Classification ブロックの先頭 トラフィックがポリシーマップ内の特定のクラスに属しているかどうかを決定するために使用するクラスマップの名前。 （注） 括弧内には、match-any または match-all で、複数のフィルタがどのように評価されるのかが示されます。
	Packets or Bytes	これらの分類カウンタは、現在のクラスのメンバーとして分類されていたパケットとバイトの数を表します。
	Offered Rate	このクラスのメンバーとして分類されているバイトから計算された減衰平均レート（1秒あたりのビット数）。
	Drop Rate	このクラス内で計算されたすべてのアクションからドロップされたアクションがある場合にその合計から計算された減衰平均レート（1秒あたりのビット数）。
	Match	パケットが現在のクラスに属しているかどうかを判断するために使用した分類基準（クラス マップのコンテンツ）の概要。
Police セクションの先頭	Police：	Police アクションの統計情報ブロックの先頭
	cir	認定情報レート
	pir	最大情報レート
	bc	適合バースト サイズ

セクション	フィールド	意味
	be	超過バースト サイズ
	conformed、packets、bytes、actions	ポリサーによって適合していると見なされたパケットとバイトの統計情報 これらのパケットに適用されたアクションの概要 経時的に適合している見なされたバイトから計算された減衰平均レート（1 秒あたりのビット数）
	exceeded packets、bytes、actions、rate	ポリサーによって超過していると見なされたパケットとバイトの統計情報 これらのパケットに適用されたアクションの概要 経時的に超過している見なされたパケットから計算された減衰平均レート（1秒あたりのビット数）
	violated packets、bytes、actions、rate	ポリサーによって違反しているとみなされたパケットとバイトの統計情報 これらのパケットに適用されたアクションの概要 経時的に違反しているとみなされたバイトから計算された減衰平均レート（1秒あたりのビット数）

例 3：show platform hardware qfp active feature qos interface コマンド

このコマンドは、通常、ルータが正しく設定されているにもかかわらず予期しない動作を行っていると考えられる場合にのみ必要になります。データプレーンから直接情報を表示すると、ハードウェアに対応するためにレートまたはバーストパラメータの量子化が必要かどうかを評価するために役に立つ場合があります。

次の例は、前の例にある **show policy-map interface** の出力に対応しています。

```
show platform hardware qfp active feature qos interface g1/0/0

Interface: GigabitEthernet1/0/0, QFP interface: 9
  Direction: Input
    Hierarchy level: 0
    Policy name: simple_policer
      Class name: AF1, Policy name: simple_policer
        Police:
            cir:  20000000 bps,  bc: 638976 bytes
            pir: 0 bps,  be: 638976 bytes
            rate mode: Single Rate Mode
            conformed: 447 packets, 668712 bytes; actions:
              transmit
            exceeded: 427 packets, 638792 bytes; actions:
              drop
            violated: 126 packets, 188496 bytes; actions:
              drop
            color aware: No
            green_qos_group: 0, yellow_qos_group: 0
      Class name: class-default, Policy name: simple_policer
```

前の 2 つのコマンド（例 1 と例 2）の出力を理解できていれば、この出力は一目瞭然です。ただし、このコマンドの使用に関連する次の点に注意する必要があります。

シングルレート 2 カラー ポリサーを設定しましたが、dataplane コマンドの出力はシングルレート 3 カラー ポリサーに対応しています。ハードウェアは常に 3 カラー モードで動作します。2 カラー機能を実現するためには、単に違反アクションと超過アクションを一致させます。統計をコントロール プレーンにプッシュする場合、IOS は超過統計と違反統計を集約して、2 カラー ポリサーの予期される外観を生成します。

データプレーン内の統計は一時的なものです。データプレーンは 10 秒ごとに統計を IOS にプッシュし、ローカル カウンタをクリアします。基本的に、dataplane コマンドによって確認されるすべての統計は、最後のプッシュ以降に発生したものの数です。これは、dataplane コマンドがハードウェアの動作をリアルタイムで表示するために役立つことを意味しています。ただし、意味のある（永続的な）統計については、常に通常の IOS コマンドの **show policy-map interface** を使用する必要があります。

QoS パケット ポリシングの設定例

例 1：単純なネットワーク アドミッション制御

最も単純な形式では、ポリサーを使用して、インターフェイス（したがってネットワーク）に入るすべてのトラフィックのレートを制限できます。この例では、トラフィックを送信するネットワークが、その出力インターフェイスから出るものを「シェーピング」し、契約レートに適合するトラフィックだけを送信すると仮定しています。送信者のネットワークで出力スケジューリングを使用して、 契約レートをさまざまなトラフィック クラスに割り当てることができます。

このポリシングの最も単純な例では、ポリサーがすべてのトラフィックを制限することを目的としているため、分類は不要です。ユーザ定義クラスがない場合は、すべてのトラフィックが class-default に属すると見なされます。

次の例では、GigabitEthernet 接続がありますが、顧客は 100 Mbps のサービスレートでのみ契約しています。設定は次のようになります。

```
policy-map ingress_cap_all_100m
  class class-default
    police cir 100000000
!
interface GigabitEthernet1/0/0
  service-policy ingress_cap_all_100m
```

例 2：ネットワーク アドミッション制御：階層型ポリサー

例 1：単純なネットワーク アドミッション制御 （332 ページ） では、送信者がその契約レート内の帯域幅を割り当てると仮定して、すべてのトラフィックを契約サービス レートにポリシングしました。ただし、送信者が個別のクラス内のトラフィックを制限することを常に信用できるわけではありません。たとえば、プライオリティ サービス（トラフィックがネットワーク全体で低遅延であることが保証される）を提供しているものの、異なるレベルのプライオリティ アクセスに関してユーザに課金するとします。例 1 の単純なポリサーを単純に適用することでは、送信者が契約よりプライオリティの高いトラフィックを転送しないことを保証できませんでした。この例を拡張することにより、個別のトラフィック クラスにも上限を適用することができます。

次の例では、総アドミッションを 100 Mbps に制限しつつ、音声トラフィックが確実に 5 Mbps のトラフィックに制限されるようにします。

```
class-map match-all voice
  match dscp ef
!
! child policy to enforce 5Mbps Voice Traffic
!
policy-map ingress_police_child
  class voice
    police cir percent 5 bc 5 ms
!
policy-map police_ingress_parent
  class class-default
    police cir 100000000
  service-policy ingress_police_child
!
interface GigabitEthernet1/0/0
  service-policy in police_ingress_parent
```

例 3：ネットワーク アドミッション制御：カラー対応ポリサー

例 2：ネットワーク アドミッション制御：階層型ポリサー （333 ページ） では、契約サービス レート内で特定のトラフィック クラスを制限するスキームを導入しました。このスキームは、トラフィックをサービス レートにシェーピングする顧客に依存します。

親ポリサー（契約サービス レート）を超えるレートでトラフィックを受信した場合、子ポリサーによって許可される音声トラフィックの一部がドロップされないという保証はありません。子ポリサーによって許可されるトラフィックが親によっても確実に許可されるようにするには、親ポリサーにカラー対応ポリサーを使用します。

次の例は、ポリサーを組み合わせることによって複雑な成果を得る方法を示しています。ここでは、子ポリサーが許可するすべての音声トラフィックを緑色（qos-group1）としてマークし、音声以外のすべてのトラフィックを黄色（qos-group2）としてマークします。親ポリサーは、すべての緑色のトラフィックを転送することを保証する CIR と、契約サービス レートを適用することを保証する PIR を使用して設定されます。

```
class-map match-all voice
  match dscp ef
!
!child policy to enforce 5Mbps Voice Traffic
!
policy-map ingress_police_child
  class voice
    !conforming voice marked Green, Excess Dropped
    police cir 5m bc 3125 conform-action set-qos-transmit 1
  class class-default
   !all traffic other than voice marked Yellow
   set qos-group2
!
class maps needed for color-aware policer
!
class-map policer-green
  match qos-group1
class-map policer-yellow
  match qos-group2
!
!parent policy to enforce 100Mbps service rate
!
policy-map ingress_police_parent
  class class-default
    police cir 5m bc 3125 pir 100m be 625000
      conform-action transmit
      exceed-action transmit
      violate-action drop
      conform-color policer-green exceed-color policer-yellow
    service-policy ingress_police_child
!
interface GigabitEthernet1/0/0
  service-policy in ingress_police_parent
```

コマンド リファレンス

police

この章で前述したように、同じ結果を得るポリシー コマンドには次の3つのバリアントがあります。

[no] police *cir* [**bc** *conform-burst*] [**pir** *pir*] [**be** *peak-burst*] [**conform-action** *action* [**exceed-action** *action* [**violate-action** *action*]]]

[no police cir *cir* [**bc** *conform-burst*] [**pir** *pir*] [**be** *peak-burst*] [**conform-action** *action* [**exceed-action** *action* [**violate-action** *action*]]]

[no] police rate*cir* [**bc** *conform-burst*] [**pir** *pir*] [**be** *peak-burst*] [**conform-action** *action* [**exceed-action** *action* [**violate-action** *action*]]]

以降、これを次のように表します。

[no] police [cir | rate]*cir* [**bc** *conform-burst*] [**pir** *pir*] [**be** *peak-burst*] [**conform-action** *action* [**exceed-action** *action* [**violate-action** *action*]]]

この同じコマンドを使用して異なるタイプのポリサーを設定する方法はすでに示しています。

紛らわしい単一のCLIや、適切ではない可能性があるオプションの組み合わせを示すのではなく、設定するポリサーのタイプに応じたオプションのサブセットを示します。

シングルレート 2 カラー ポリサー

指定された適合レベルごとに1つのアクションだけが必要な場合は、このポリサー タイプを1行で表すことができます。

[no] police [cir | rate]*cir***[percent** *percent*]**[[bc]** *conform-burst* **[ms]]** [**account** *options*]**[conform-action** *action* **[exceed-action** *action***]]**

複数のアクションが必要な場合は、複数の行（サブモードを使用）が必要です。

[no] police [cir | rate]*cir***[percent** *percent*]**[[bc]** *conform-burst* **[ms]][account** *options*] **<return>**
[no] conform-action *action* **<return>**
[no] conform-action *action* **<return>**
[no] exceed-action *action* **<return>**
[no] exceed-action *action* **<return>**

シングルレート 3 カラー ポリサー

指定された適合レベルごとに1つのアクションだけが必要な場合は、このポリサー タイプを1行で表すことができます。

[no] police [cir | rate]*cir***[percent** *percent*]**[[bc]** *conform-burst* **[ms]]** [**[be]** *exceed-burst* **[ms]][account** *options*]**conform-action** *action* **exceed-action** *action* **[violate-action** *action***]**

複数のアクションが必要な場合は、複数の行（サブモードを使用）が必要です。

[no] police [cir | rate]*cir***[percent** *percent*]**[[bc]** *conform-burst* **[ms]]** [**[be]** *exceed-burst* **[ms]][account** *options*] **<return>**
[no] conform-action *action* **<return>**
[no] conform-action *action* **<return>**
[no] exceed-action *action* **<return>**
[no] exceed-action *action* **<return>**
[no] violate-action *action* **<return>**
[no] violate-action *action* **<return>**

デュアルレート 3 カラー ポリサー

指定された適合レベルごとに 1 つのアクションだけが必要な場合は、このポリサー タイプを 1 行で表すことができます。

[no] police [cir | rate]*cir***[percent** *percent***][[bc]** *conform-burst* **[ms]] [pir]** *peak-rate* **[ms][[be]** *exceed-burst* **[ms]][account** *options*]**conform-action** *action* **exceed-action** *action* **[violate-action** *action***]**

複数のアクションが必要な場合は、複数の行（サブモードを使用）が必要です。

[no] police [cir | rate]*cir***[percent** *percent***][[bc]** *conform-burst* **[ms]] [[pir]** *peak-rate* **[ms]][[be]** *exceed-burst* **[ms]][account** *options*] **<return>**
[no] conform-action *action* **<return>**
[no] conform-action *action* **<return>**
[no] exceed-action *action* **<return>**
[no] exceed-action *action* **<return>**
[no] violate-action *action* **<return>**
[no] violate-action *action* **<return>**

シングルレート 3 カラー カラー対応ポリサー

[no] police [cir | rate]*cir***[percent** *percent***][[bc]** *conform-burst* **[ms]] [[be]** *exceed-burst* **[ms]][account** *options*] **<return>**
[no] conform-action *action* **<return>**
[no] conform-action *action* **<return>**
[no] exceed-action *action* **<return>**
[no] exceed-action *action* **<return>**
[no] violate-action *action* **<return>**
[no] conform-color *conform-color* **exceed-color** *exceed-color***<return>**

デュアルレート 3 カラー カラー対応ポリサー

[no] police [cir | rate]*cir***[percent** *percent***][[bc]** *conform-burst* **[ms]][[pir]** *peak-rate* **[ms]] [[be]** *exceed-burst* **[ms]][account** *options*] **<return>**
[no] conform-action *action* **<return>**
[no] conform-action *action* **<return>**
[no] exceed-action *action* **<return>**
[no] exceed-action *action* **<return>**
[no] violate-action *action* **<return>**
[no] conform-color *conform-color* **exceed-color** *exceed-color***<return>**

police コマンドのデフォルトおよびモード：キーワード/引数の説明

コマンド デフォルト 無効化

コマンド モード マークされたパケットに適用される単一のアクションを指定する場合は、ポリシー マップ クラス コンフィギュレーション（config-pmap-c）

マークされたパケットに適用される複数のアクションを指定する場合は、ポリシー マップ クラス ポリシング コンフィギュレーション（config-pmap-c-police）サブモード

構文の説明

次の表に、**police** コマンドのキーワード/引数とその目的を示します。

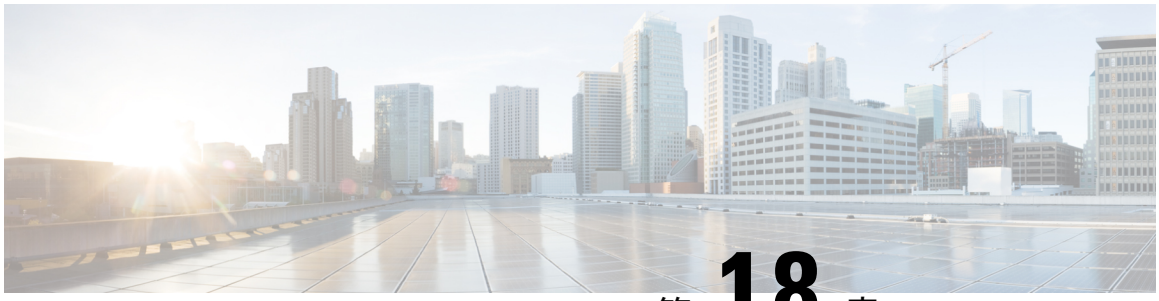
キーワード/引数	定義
bc	適合バーストサイズ（Bc）を指定します。
be	超過バーストサイズ（Be）を指定します。
cir	認定情報レート（CIR）を指定します。
Conform-Action	「適合」と判断されたトラフィックに対して実行されるアクションを指定します。
Exceed-Action	「超過」と判断されたトラフィックに対して実行されるアクションを指定します。
pir	最大情報レート（PIR）を指定します。
Violate-Action	「違反」と判断されたトラフィックに対して実行されるアクションを指定します。

次の表に、Account キーワードのオプションを示します。

表 *25 : Account* キーワードのオプション

オプション	目的
qinq	queue-in-queue カプセル化を BRAS-DSLAM カプセル化タイプとして指定します。
dot1q	IEEE 802.1Q VLAN カプセル化を BRAS-DSLAM カプセル化タイプとして指定します。
aal5	コネクション型可変ビット レート（VBR）サービスをサポートする ATM アダプテーション層 5 を指定します。
aal3	コネクションレス型リンクとコネクション型リンクの両方をサポートする ATM アダプテーション層 3 を指定します。
subscriber-encapsulation	加入者線でのカプセル化タイプを指定します。

オプション	目的
user-defined	ポリシング長の調整時に指定されたオフセット値をルータが使用することを示します。
offset	オーバーヘッドを計算する際にルータが使用するバイト数を指定します。-63 ～ 63 バイトの範囲内の値を指定できます。
atm	ATM オーバーヘッド計算に ATM セルタックスを適用します。



第 18 章

キュー制限と WRED

概要

Cisco IOS XE デバイスでは、出力インターフェイスまたは QoS キューに入れられるパケットが、専用のメモリにを格納されます。メモリは、個別のインターフェイスによって切り分けられたり所有されるのではなく、すべてのインターフェイスが利用できるグローバルプールとして扱われます。

キュー制限により、 特定のキューの深さが制限されます。これは、2 つの目的に適います。1 つ目は、個々のキューが使用可能なパケットメモリの量を制限することです。これにより、他のインターフェイスまたはキューもこの共有リソースに公平にアクセスできる状態が確保されます。2 つ目は、キューが輻輳状態のときに格納されるデータ量を制限することです。これにより、そのキュー内のアプリケーションに対して発生する遅延が制限されます。

パケットをキューに入れる準備ができると、そのキューの現在の深さと設定されているキュー制限が確認されます。前者がすでに後者に達している場合は、パケットがドロップされます（テール ドロップ）。

キュー制限

ASR 1000 シリーズ アグリゲーション サービス ルータ（以前の ASR 1000 シリーズ ルータ）のパケット メモリは共有リソースです。個々のインターフェイスおよびキューがこのメモリの一部に割り当てられることはありません。むしろ、それらは、先着順ですべてのキューが利用可能なグローバル プールを表します。

個々のキューが共有パケット メモリに格納できるデータ量を制御するために、キュー制限（キューごとに設定可能な値）が使用されます。これは、2 つの目的に適います。1 つ目は、キューがほぼ満杯になるまで、着信パケットの遅延を制限することです（場合によっては、受信側でパケットが実用的でなくなるほど低速でパケットを配信するよりもドロップする方が適していることがあります）。2 つ目は、単一のインターフェイスが多数のパケットを共有メモリに格納するために他のインターフェイスが使用できるメモリがなくなる可能性がないようにすることです。

これにより、共有メモリが非常に効率的に管理されます。バッファのプールを事前に決められたサイズに切り分ける代わりに、このハードウェアは、メモリ ブロック（元の QFP では 32 バイトのブロック）を管理し、パケットを格納するために必要な最小ブロック数を割り当てます。

次の表に、プラットフォームによるパケット メモリの量と設定可能な最大キュー数の違いを示します。

ESP（埋め込み型サービス プロセッサ）ルータ ハードウェア	パケット メモリ	最大キュー数
ASR1001	64 MB	16,000
ASR1001-X	512 MB	16,000
ASR1002-F	64 MB	64,000
ASR1002-X	512 MB	116,000
ESP5	64 MB	64,000
ESP10	128 MB	128,000
ESP20	256 MB	128,000
ESP40	256 MB	128,000
ESP100	1 GB（512 MB X 2）	232,000*
ESP200	2 GB（512 MB X 4）	464,000*

ESP100 および ESP200 の場合、物理ポートは ESP カード上の特定の QFP（Quantum Flow Processor）コンプレックスに関連付けられます。すべてのキューを最大限に使用するには、それらをシャーシ内の異なるスロットと SPA に分散させる必要があります。

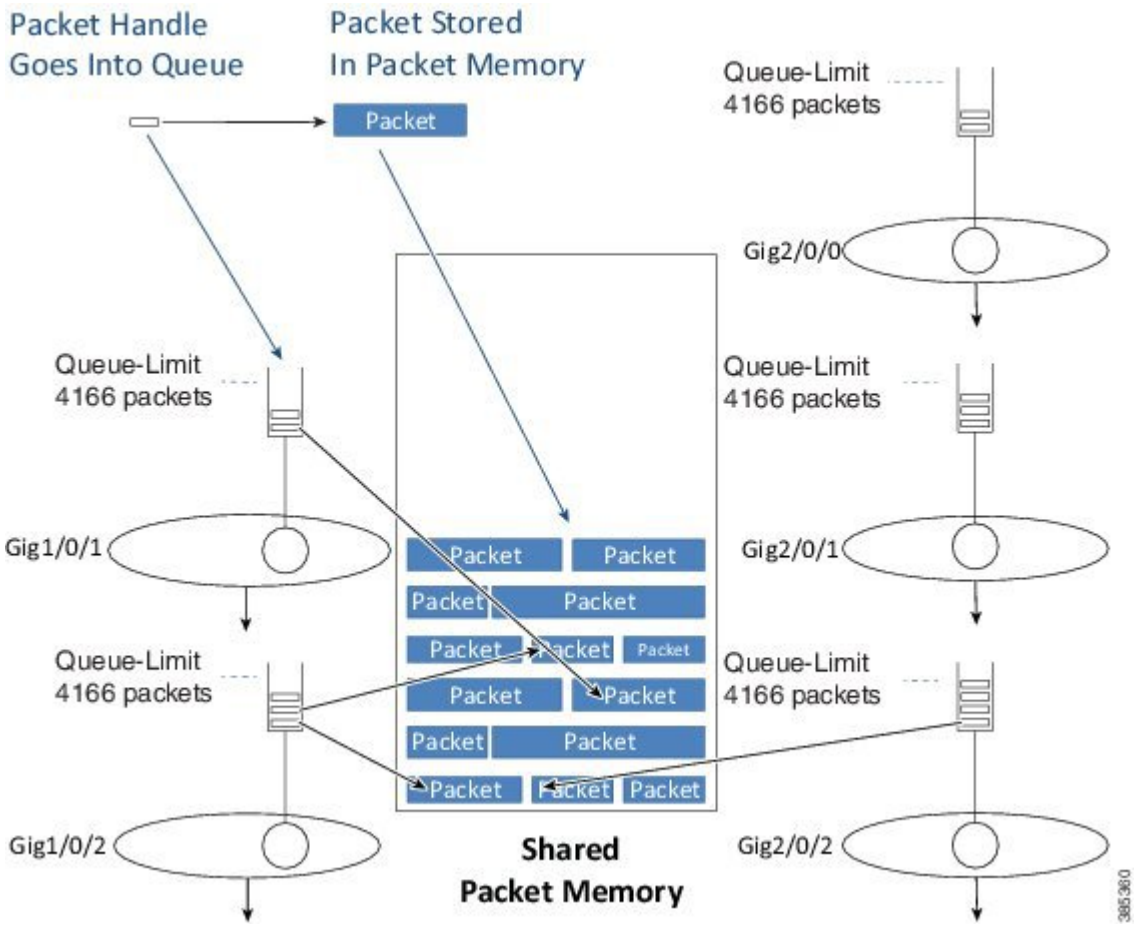
ASR 1000 シリーズ ルータのパケット メモリの量は、さまざまな要因（コスト、テクノロジーのアベイラビリティ、重視される事柄など）によって決まります。QFP が最初にリリースされたときは、パケット メモリに必要な 1 秒間に数十ギガビットの読み取り（および書き込み）を処理するために使用できるメモリ テクノロジーの選択肢はほとんどありませんでした。メモリが速度要件に対処できたとしても、サイズの選択肢は限られており、モジュールのコストも非常に高いため、より多くのメモリを搭載したシステムを設計することは可能でしたが、それは現実的な利点のない非常に高価なものでした。

サポートされるキューの数だけでなく、パケットがシステムに出入り可能なレートも考慮する必要があります。たとえば、ESP10 の場合、「128MB と 128,000 のキュー」は「1 キューあたり 1KB のメモリ」と言い換えることができます。128,000 のキューのすべてを同時に輻輳させることが決してない場合、このことは大きな意味を持ちません。

ただし、サイズを別の観点から見ると、ESP10 は 10Gbps の最大レートでデータを送受信できます。この速度では 128 MB のメモリによって 100 ミリ秒を超えるバッファリングが提供され、これは非常に妥当な値です。

これらのことから、システム内のすべてのキュー制限を合計することはオーバーサブスクライブになると明らかに予測されます。

図 *71* : キュー制限

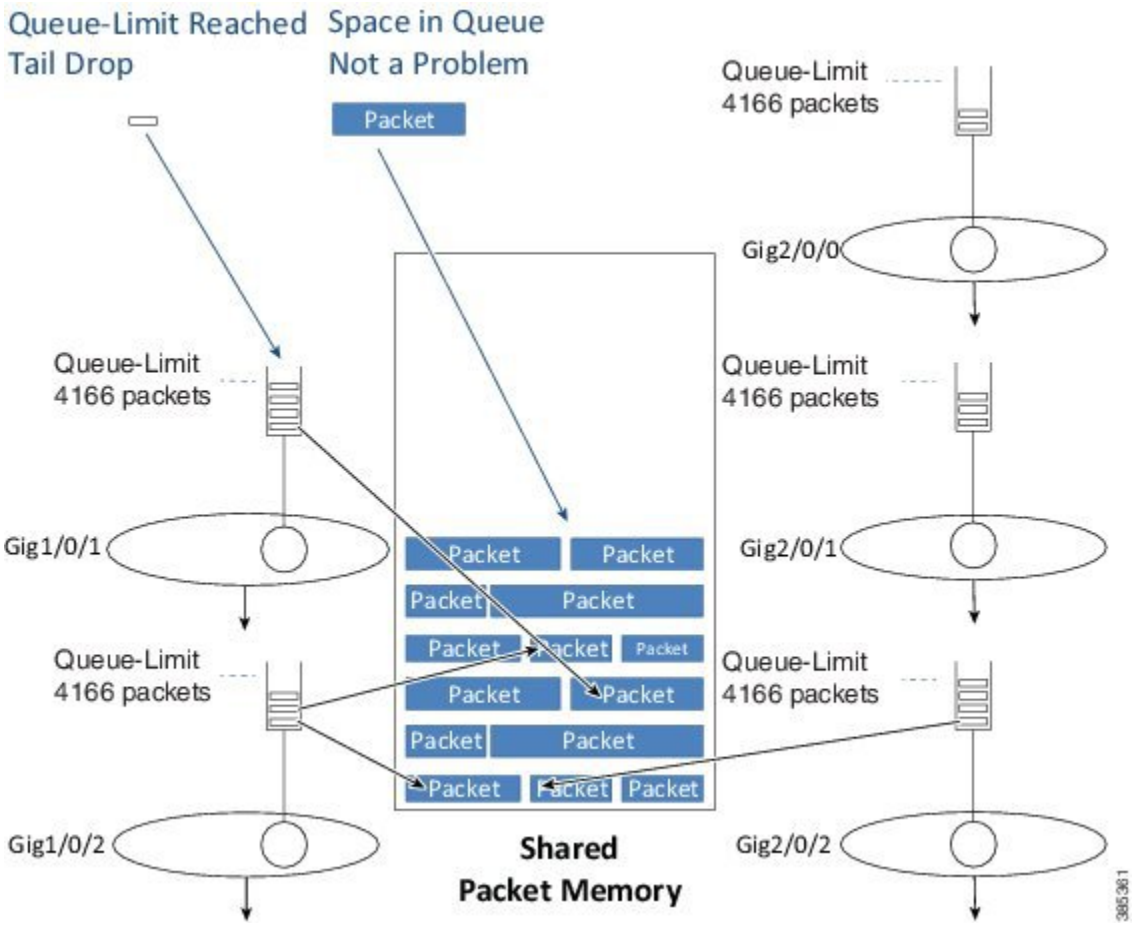


パケットの出力インターフェイスを決定すると、そのインターフェイスのキュー情報が分かります。そのキューに、パケットのスケジューリング長を含む小さなパケット ハンドルと、共有パケット メモリ内のパケットが格納される場所を示すポインタを配置します。実際のパケット自体は共有パケット メモリに格納されます。

テール ドロップ

パケットのエンキュー時には、最初に、設定されているキュー制限と、そのインターフェイスが現在バッファしているデータの量（瞬間キュー深度）が確認されます。

図 *72*：テール ドロップ



キューの深さが、事前設定された制限にすでに達している場合は、パケットがドロップされ、**テール ドロップ**が記録されます。

QoS が設定されていない場合は、**show interface** コマンドの出力でドロップを確認できます。

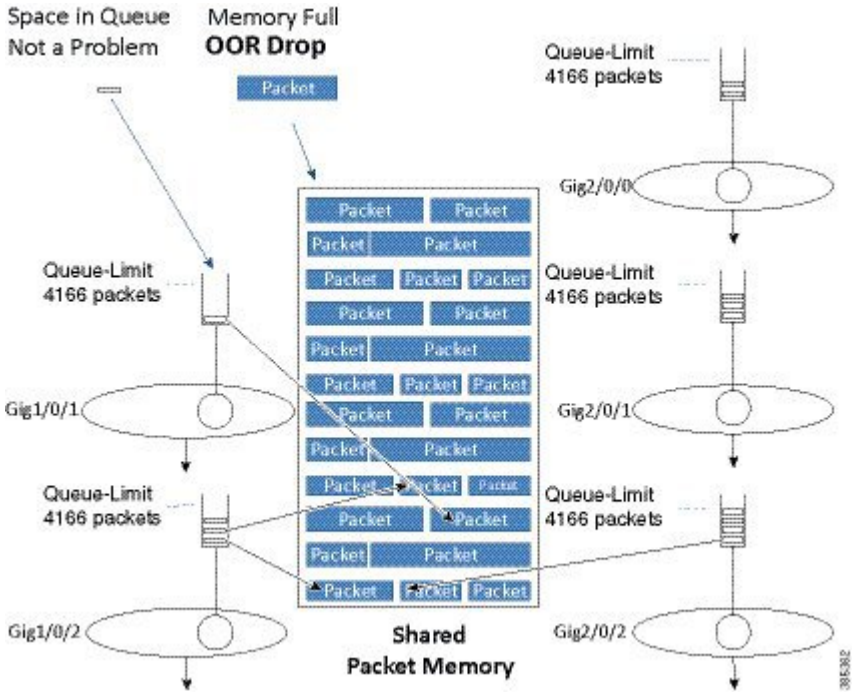
QoS が設定されている場合は、**show policy-map interface** のクラス出力でドロップを確認できます。

図に示されているように、テール ドロップは、パケットを格納するメモリが存在しないことを意味するわけではなく、キューが、格納可能なデータ量の個別の上限にすでに達していることを意味します。

リソース不足のドロップ

キューがまだ個別のキュー制限に達していないものの、共有パケットメモリが満杯になっている可能性がある場合は、エンキュー時に別のシナリオが考えられます。この場合、パケットを格納する場所が存在しない場合は、そのパケットをドロップする必要があります。このドロップは「バッファなし」ドロップとして記録され、リソース不足（*OOR*）状態として Syslog にレポートされます。

図 *73 : ORR* ドロップ

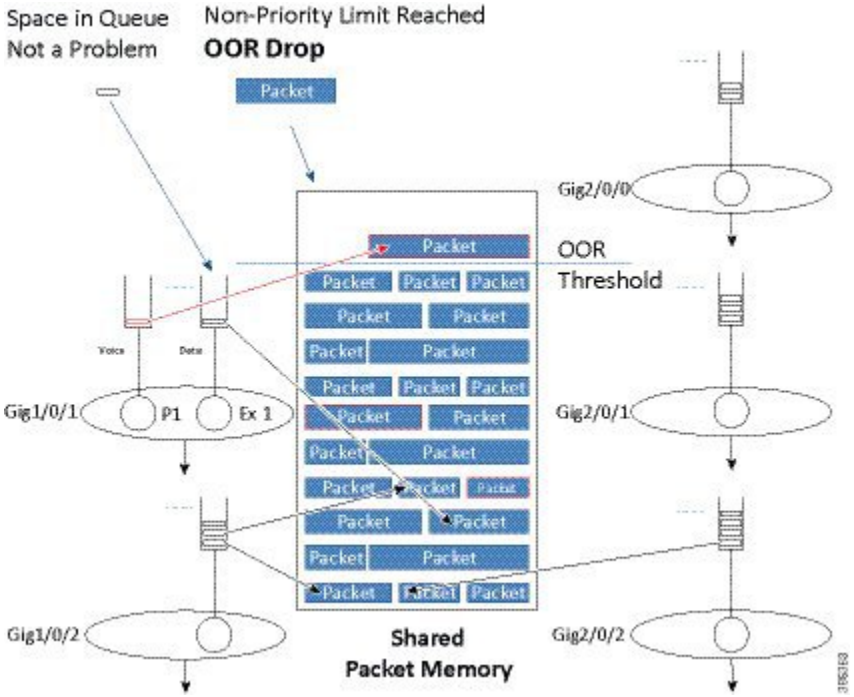


OOR ドロップがごくまれにしか発生しない場合は、それらを無視できます。ただし、これが通常の状態となっている場合は、キュー制限を調べて、個別のキューまたはインターフェイスで過剰な量のメモリの消費が可能になっていないか確認する必要があります。この状況を回避するには、1 つ以上のキューのキュー制限を低くする必要がある場合があります。

プライオリティ パケット用に予約されるメモリ

「パケットメモリが 100％消費される場合」という言い方は、実際には正確ではありません。一部のパケット（プライオリティ クラスのパケットや pak_priority パケット）は他のパケットよりも重要であるため、それらの重要なパケットをメモリに格納するために常にメモリにスペースを確保することが望まれます。これを実現するには、通常データのキューからのパケットを総パケット メモリの 85 % に制限します。

図 *74 :* プライオリティ パケット用に予約されるメモリ



上の図は、プライオリティ パケットとデータ パケットの処理方法の違いを示しています。このシナリオでは、パケット メモリの 85 % が消費されています。通常のデータ パケットが着信すると、OOR しきい値に達しているためにドロップされます。ただし、プライオリティ パケットが着信すると、利用可能な物理スペースがあるため、引き続きエンキューされます。

プライオリティ パケットがメモリの小さなスペースの制限されていないことに注意してください。その代わりに、メモリがほぼ満杯になると非プライオリティ パケットがドロップされます。

バイタルしきい値

メモリ使用率が 98 % を超えるとプライオリティ パケットを含むすべてのユーザ トラフィックをドロップする第 2 レベルの保護も提供されます。これは「バイタルしきい値」と呼ばれ、内部制御パケット（システム内の異なる制御プロセッサ間を移動する必要がある可能性のあるインバンド パケット）をエンキューすることを可能にします。プライオリティ パケットは、通常、キューに入れられると転送されるため、98 % のしきい値を超えることは予期されません。

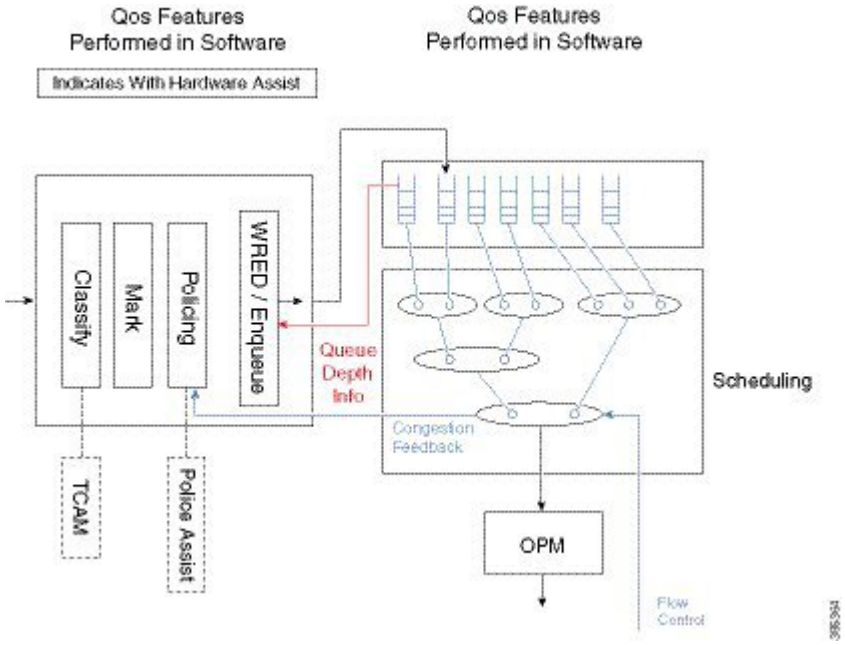
show platform hardware qfp active bqs 0 packet-buffer utilization コマンドを使用すると、システム内のメモリ容量とそのメモリのリアルタイム使用率を確認できます。

```
show platform hardware qfp active bqs 0 packet-buffer utilization
Packet buffer memory utilization details:
  Total:     256.00 MB
  Used :    2003.00 KB
  Free :     254.04 MB

  Utilization:    0 %
```

```
Threshold Values:
  Out of Memory (OOM)      :     255.96 MB, Status: False
  Vital (> 98%)             :     253.44 MB, Status: False
  Out of Resource (OOR)    :     217.60 MB, Status: False
```

ASR 1000 シリーズアグリゲーションサービスルータでは、すべてのキューイング、スケジューリング、およびパケットメモリ管理は専用ハードウェアによって実行されます。パケットがエンキューされるときに、ソフトウェアからハードウェアに制御が渡されます。ハードウェア、特に BQS（バッファリング、キューイング、およびスケジューリング）サブシステムがメモリを管理するため、そのハードウェアによって、各キューがパケットメモリに現在格納しているデータの量がモニタされます。パケットにエンキューできるようになると、ハードウェアに対して現在のステータスのクエリが行われます。ハードウェアは、そのキューの瞬間キュー深度と平均キュー深度をレポートします。その後、ソフトウェアによって、エンキューを続行するかパケットをドロップするかが決定（およびレポート）されます。テールドロップの決定は、ハードウェアによってレポートされる瞬間キュー深度を使用して行われます。一方、WRED では、平均キュー深度が使用されます（平均キュー深度 （355 ページ） を参照）。

図 *75*：バイタルしきい値



パケット モードとバイト モード

ハードウェアは、パケット モードとバイト モードのいずれかで動作します。ハードウェアが瞬間キー深度や平均キュー深度がレポートするときは、それらがパケット単位またはバイト単位でレポートされますが、両方でレポートされることはありません。このモードはキューの作成時に設定され、ポリシー マップを削除して適用しなおさないかぎり変更できません。

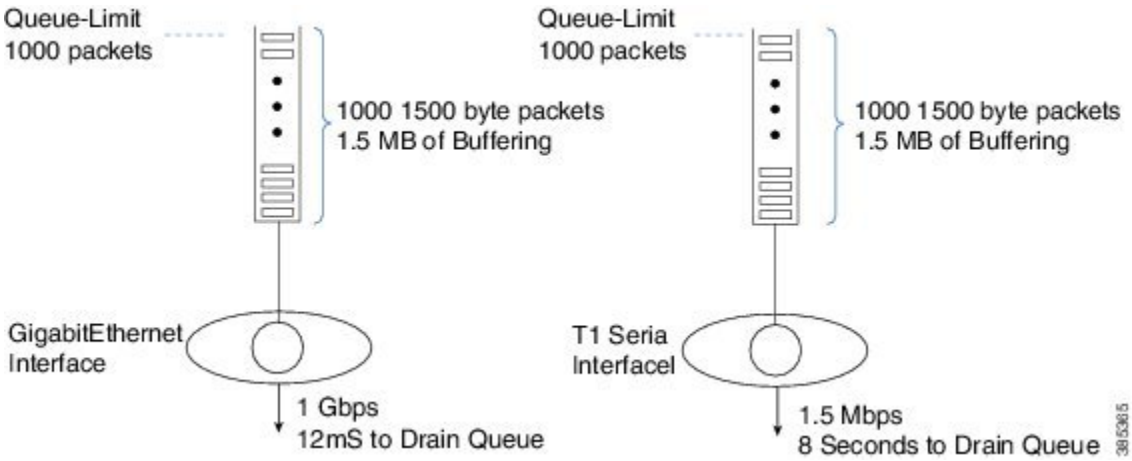
上の図は、一部の QoS 機能がソフトウェアで実行され、その他の機能がハードウェアで実行される仕組みを示しています。エンキューは、実際には2つの境界で行われます。ソフトウェ

アは、ハードウェアからキューの深さに関する情報を受信し、パケットをドロップするかパケットをパケットメモリに移動させてキューにパケットハンドルを追加するかを決定します。WRED はより高度な形式のドロップ判断であり、この章の後半で説明します。

デフォルトのキュー制限

次の図は、可変キュー制限の必要性を示しています。

図 *76 :* 可変キュー制限



左側のキューは 1 Gbps で処理されます。1000 の 1,500 バイト パケットが送信を待っている場合、キューのドレインには12ミリ秒かかります。これは、ほぼ満杯のキューに到着したパケットが転送される順番を待っている間に 12 ミリ秒の遅延が発生する可能性があることを意味します。

右側のスケジュールは T1 インターフェイスを表しています。これは約 1.5 Mbps で動作するかなり遅いインターフェイスです。同じ 1000 パケットが T1 インターフェイスを介した送信を待っている場合、キューのドレインには8秒かかります。明らかに、そのような遅延では、ほとんどのユーザ（およびアプリケーション）の要求に応えることができません。

図には、前述のキュー制限の2つ目の役割（キュー内のアプリケーションの遅延を制限すること）が示されています。

デフォルトのキュー モードとキュー制限を決定する方法は、QoS が設定されているかどうかによって異なります。

可能なかぎり多くのユーザに適したデフォルトのキュー制限を選択しますが、それが常に最適な選択であるとはかぎらないことに注意してください。システムで設定されている物理インターフェイスと論理インターフェイスの数、キュー内のバーストトラフィックの量、キュー内のアプリケーションの遅延要件などを考慮しない場合はデフォルトが優れた出発点となりますが、通常は、キュー制限をさらに調整します。

Qos が設定されていない場合

スケジューリングに関する章では、QoSが設定されていない場合にすべてのパケットが「インターフェイス デフォルト キュー」と呼ばれる単一のFIFOを通過することを説明しました。インターフェイス デフォルト キューのキュー制限はバイト単位で設定され、インターフェイス速度に基づいて 50 ミリ秒分のバッファリングとして計算されます（25 ミリ秒が使用される ESP-40 は例外です）。

GigabitEthernetインターフェイスを例に説明します。このインターフェイスの速度は 1 Gbps ですが、内部オーバードライブにより 1.05 Gbps で送信されます。

50 ミリ秒分のバッファリングはバイト単位では、1.05 Gbps/8 ビット（1 バイト） * 0.05 秒 = 6,562,500 バイトです。

show platform hardware qfp active infrastructure bqs queue output default interface gig1/0/0 | inc qlimit コマンドを使用すると、インターフェイス デフォルト キューのキュー制限を表示できます。

QoS が設定されている場合

（注） MQC CLI を使用して作成されるキューは、デフォルトでパケット モードになります（これはパケット モードが優れていることの承認ではなく歴史的な結果です）。

キュー制限の計算は、いくつかの要因に依存します。

キューがプライオリティ キューの場合、デフォルトのキュー制限は 512 パケットです。もちろん、これは大きな制限ですが、これらの値には意味がありません。キューアドミッション制御により、パケットが確実に、送信されるよりも低いレートでエンキューされるため、プライオリティ キューは常にほぼ空になります。そのため、キュー制限を任意に大きく設定し、それをすべてのインターフェイス速度で使用することができます。

帯域幅キューについては、バッファリングされるデータの最大 50 ミリ秒分を目標としますが、低速キューでは、これが非常に少量のデータに当たるす可能性があるために例外になります。送信されるデータの量（50 ミリ秒単位）を計算するには、サービスの速度を知る必要があります。インターフェイス デフォルト キュー（前述のように、QoS のないシナリオでは唯一の選択肢）の場合、これは単純です。つまり、1 つのキューがインターフェイスの帯域幅全体を「所有」します。QoS が設定されている場合は、状況が複雑になります。

まず、「可視帯域幅」の概念を導入する必要があります。 これは設定から確定される値であり、これによって、かかっている負荷を考慮せずにキューのサービス レートが取得されます。次の表は、使用されるコマンドに可視帯域幅がどのように依存するのかを示しています。

表 *26 :* 使用されるコマンドに依存する可視帯域幅の表現

コマンド	可視帯域幅
shape	シェーピング レート

コマンド	可視帯域幅
bandwidth	帯域幅レート
shape and bandwidth	帯域幅レート
bandwidth remaining	親から直接継承されます。 • ポリシー マップが物理インターフェイスに適用されている場合、継承される値はインターフェイス速度です。 • ポリシーが親シェーパーを持つ子ポリシーである場合、可視帯域幅は親シェーピング レートになります。

次に、ポリシーが適用されるインターフェイスの最大伝送ユニット（MTU）が必要です。パケット単位（前述のように、これがデフォルトです）のキュー制限を設定しており、潜在的な遅延を制限することが目的であるため、キューが MTU サイズのパケットで満杯になるという最悪のシナリオを調べます（**show interface** コマンドの出力で MTU を確認します）。

可視帯域幅、MTU、およびバッファリングされるデータの最大 50 ミリ秒分が与えられると、キュー制限を次のように計算できます。

キュー制限 =（可視帯域幅/8 ビット）* 50 ミリ秒/MTU

GigabitEthernet インターフェイスで 100 Mbps にシェーピングされるキューについて検討します。可視帯域幅はシェーピング レート（100 Mbps）、MTU は 1500 バイト（イーサネット タイプのインターフェイスで予期される値）になります。

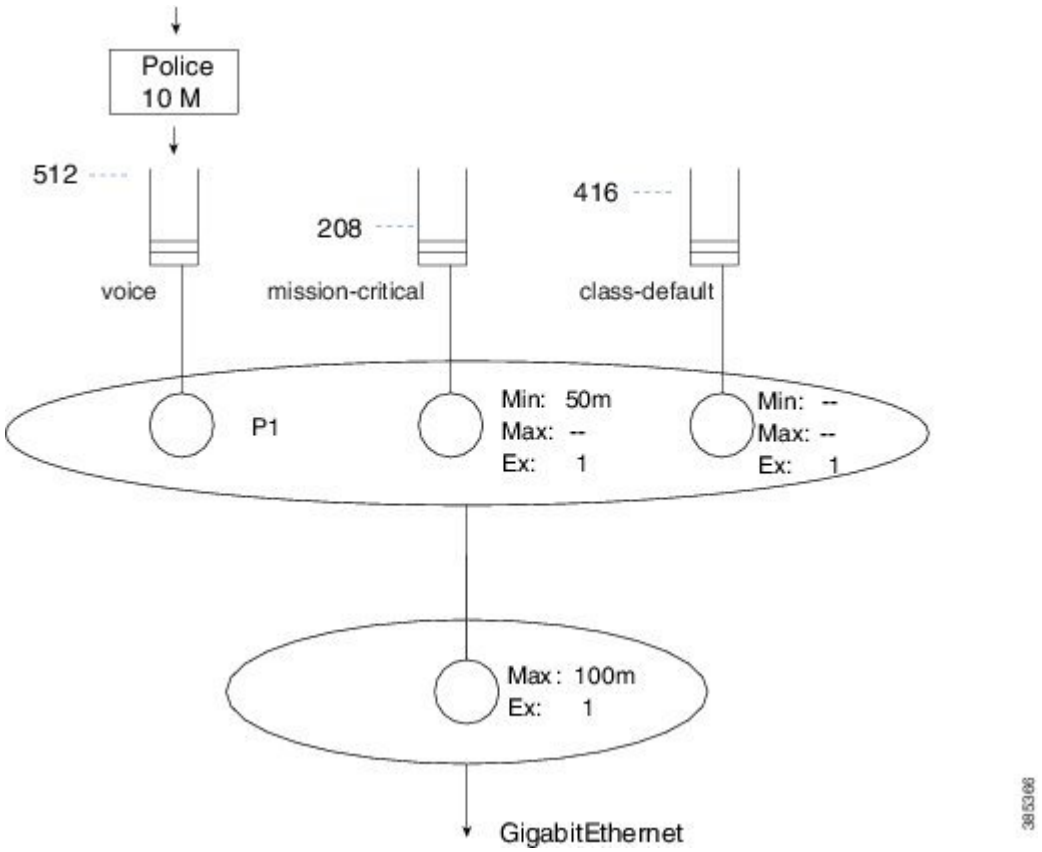
キュー制限 = 100 Mbps/8 ビット * 0.05 秒/1500 バイト = 416 パケット

前述のように、低速キューは例外です。計算されたキュー制限が 64 パケット未満の場合は、キュー制限として 64 パケットが使用されます。

デフォルトのキュー制限を計算する方法のより包括的な例について説明します。GigabitEthernet インターフェイスに次のような階層型ポリシー マップが適用されるとします。

```
policy-map child
  class voice
    priority
    police cir 10m
  class mission-critical
    bandwidth 50000
policy-map parent
  class class-default
    shape average 100m
    service-policy child
interface GigabitEthernet1/0/0
  service-policy out parent
```

完全を期すために、このポリシー マップのスケジューリング階層を次に示します。



子ポリシー マップには、voice、mission-critical、および class-default の 3 つのキューイング クラスがあります。それぞれ、次のようなものです。

voice キューはプライオリティ キューであるため、キュー制限はデフォルトで 512 パケットです。

mission-critical キューは **bandwidth** コマンドによって 50 Mbps のレートで設定されているため、可視帯域幅は 50 Mbps です（上の表を参照）。これはイーサネット タイプのインターフェイスであるため、MTU は 1500 バイトです。

キュー制限 = 50 Mbps/8 ビット * 0.05 秒/1500 バイト = 208 パケット

暗黙の class-default は queuing コマンドで設定されていませんが、暗黙の超過の重みは **bandwidth remaining ratio 1** の設定と同じです。これは、class-default がその可視帯域幅を親から継承することを意味します（上の表を参照）。親で 100 Mbps の値によって設定されるシェープに注意してください。そのため、子の class-default の可視帯域幅は 100 Mbps であり、前述のように、このインターフェイス タイプの MTU は 1500 バイトです。

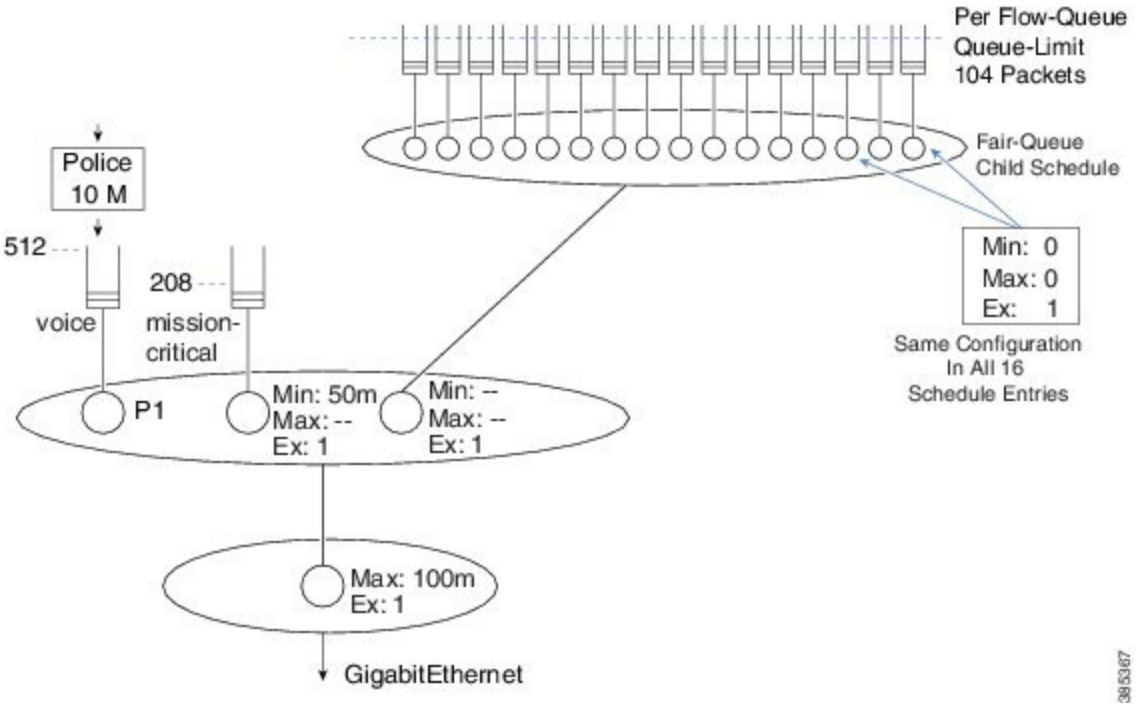
キュー制限 = 100 Mbps/8 ビット * 0.05 秒/1500 バイト = 416 パケット

均等化キューが設定されている場合

フローベースの均等化キューイングには、クラスに 16 の個別フロー キューが設定され、各フローキューが同じキュー制限で設定される、フローベースの均等化キューイングが導入されています。デフォルトでは、この制限は、均等化キュー機能が設定されているクラスの可視帯域幅に基づいて計算される値の 1/4 です。

例として、前の設定例の class-default に均等化キューを追加します（アスタリスクを参照）。

```
policy-map child
  class voice
    priority
    police cir 10m
  class mission-critical
    bandwidth 50000
  class class-default                          ****
    fair-queue                                 ****
policy-map parent
  class class-default
    shape average 100m
    service-policy child
interface GigabitEthernet1/0/0
  service-policy out parent
```



以前は、親のシェーパーから継承された可視帯域幅に基づいて、class-default のキュー制限を 416 パケットと計算しました。

フローベースの均等化キューイングが設定されているため、その 1 つのクラスに対して 16 のフローキューが作成されます。各個別フローキューのキュー制限は、104 パケット（計算された 416 パケットの 1/4）に設定されます。

キュー制限の変更

前述のように、プラットフォームによって設定されるデフォルトのキュー制限を大半のユーザに適用できますが、場合によってはそれらを調整する必要があります。

キュー制限を変更する理由と状況

キュー制限の調整になる一般的な状況は、OOR ドロップ、テール ドロップが発生するバースト性の高いトラフィック、および遅延に関する問題の 3 つです。

OOR ドロップが発生する場合は、その状況を回避するためにキュー制限を低下させる必要がある可能性があります。各帯域幅残存キューは親から可視帯域幅を継承することが予期されるため、そのようなキューが多数作成されると OOR ドロップが発生する可能性があります。さらに、キュー制限をバイト モードに変更すると、特定のキューが消費できるパケット メモリの量をより詳細に制御できる場合があります。

場合によっては、長期的なストリームのレートがキューの最小サービス レートよりも小さいと判明します。その場合でも、依然としてパケットのテール ドロップが発生します。このことは、キュー制限を大幅に増やすことによって実験できます。システムのオーバーサブスクリプションが原因である場合は、キュー制限をどれだけ大きくしてもテール ドロップが発生します。バースト性によってドロップが発生している場合は、パケット損失が発生しなくなります。出発点として、キュー制限を2倍にすることをお勧めします。ドロップがなくなった場合は、元のキュー制限の 1.5 倍に減らすことを試みてください。このようにして、ドロップが発生せず、OORの問題が発生する可能性のある不当に大きなキュー制限でもない値を探ります。同じスケジュール内で非常に低いレートと高いレートを混在させることによって発生するスケジュールのバースト性も原因になる可能性があることに注意してください。

最後に、キューが輻輳状態になったときの過度の遅延を避けるために、キュー制限を調整する必要がある場合があります。可視帯域幅が約 15Mbps 未満のキューがある場合、それらにはデフォルトの 64 パケットの最小キュー制限が割り当てられます。低速インターフェイスに複数のキューを追加すると、それらのキューの最低保証サービス レートが非常に低くなる可能性があります。この場合は、キュー制限をバイト モードに変更することをお勧めします。

QoS キューの場合

queue-limit コマンドを使用すると、キューイング アクション（帯域幅、帯域幅残存、プライオリティ、またはシェープ）を含む任意のクラスのキュー制限を変更できます。キュー制限は、パケット数（デフォルト）、バイト数、または時間で指定できます（それぞれの例を説明します）。次は、パケット モードで制限を設定する例です。

```
policy-map packet-mode-example
  class critical-data
    bandwidth percent 50
    queue-limit 2000
```

queue-limit コマンドを、バイト オプションを指定して使用すると（2つ目のオプション）、前述のようにキューのモードがパケットからバイトに変更されます。この変更を実行するには、ポリシーマップを削除して再適用（または設定を保存してルータをリロード）する必要があります。WRED しきい値をバイト単位で指定する場合は、最初に **queue-limit** コマンドを使用してキューのモードをバイトに変更する必要があります。

```
policy-map byte-mode-example
  class critical-data
    bandwidth percent 50
    queue-limit 5000 bytes
```

（注） ポリシーがインターフェイスに適用されているときにキュー制限のモードを変更しようとすると、エラー メッセージが表示されます。

```
queue-limit 5000 bytes
Runtime changing queue-limit unit is not supported,please remove service-policy first
```

3つ目のオプションは、キュー制限を時間（ミリ秒単位）で指定することです。実際には、ハードウェアは、パケット単位またはバイト単位だけをサポートしています。ミリ秒単位で指定すると、ルータはこれをバイト単位に変換します。つまり、事実上モードをバイトに変更することになります。ルータはクラスの可視帯域幅を使用します（QoS が設定されている場合 （347 ページ）を参照）。

```
policy-map time-mode-example
  class critical-data
    shape average 20m
    queue-limit 50 ms
```

この例では、キューの可視帯域幅は 20 M ビット/秒（2.5 M バイト/秒）です。2.5 M バイト/秒のレートの場合、50 ミリ秒 では 125000 バイト（0.05 秒 * 2.5 Mbps）のデータが生成されます。そのため、この例では、キュー制限を 125000 バイトに設定します。**show policy-map interface** コマンドの出力で計算された値を確認できます。

インターフェイス デフォルト キューの場合

QoS ポリシーが適用されていないインターフェイスのキュー制限を直接変更することはできません。従来の IOS では、**hold-queue** コマンドによってこれが実現されました。IOS XE では、ホールド キューは IOSd デーモン内に存在しますが、通常のパケット転送パスは意味を持ちません。ただし、非常に多くのルーティング ピアを含むトポロジがあり、そのすべてのピアからの同時更新を処理するために IOSd 内でより多くのバッファリングが必要な場合は、ホールド キューの調整が依然として、IOSd にパントされるパケットに対して意味を持ちます。

インターフェイス デフォルト キューのキュー制限を変更するために、class-default だけの単純なポリシー マップを適用できます。

```
policy-map modify-interface-queue
  class class-default
    queue-limit 100 ms
!
```

```
interface gigabitethernet1/0/0
  service-policy out modify-interface-queue
```

WRED

WRED はキューの使用率をモニタする機能です。輻輳が発生している場合、WRED は、さらなる輻輳を緩和するために、エンドポイントに信号を送るパケットをランダムにドロップして転送レートを低下させます。

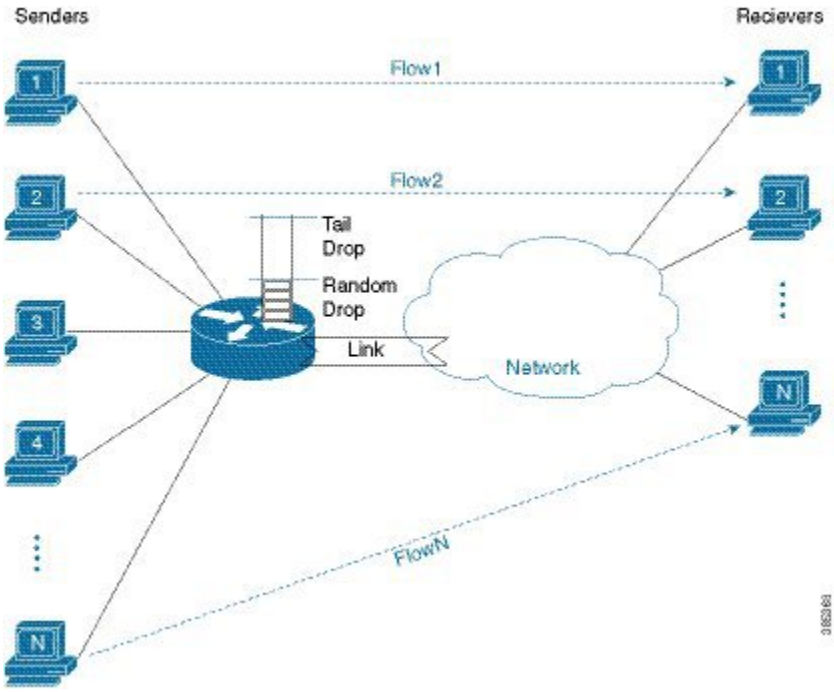
IP フローの弾力性への依存

WREDは、多数の*IP*フローの弾力性に依存します。この場合、「弾力性」は、受信者がパケット損失を検出したときに送信レートを増減させるフローを意味します。TCPは、弾力性の非常に良い例です。TCPは、低速で動作を開始し、送信者の輻輳ウィンドウ（許可される未処理の未確認トラフィックの量）を、受信者の最大受信ウィンドウサイズに達するかネットワーク内のパケットを失うまで増加させます。後者の場合は、輻輳回避アルゴリズムに切り替わり、、パケットを失うことなく達成可能な最大輻輳ウィンドウサイズの確定を試みます（詳細については、RFC 5681 を参照）。

弾力性のあるトラフィックのもう一つの良い例はビデオです（任意のビデオ ストリーミング アプリケーションを思い浮かべてください）。ビデオが開始されると、一般に、レートが上がるにつれてビデオの品質が改善されます。アプリケーションがネットワークの容量を認識するまでレートが上がりつづけます。ネットワーク内でドロップが検出されると、レートの上昇が停止し、優先されるネットワーク条件を考慮して可能な最高の品質が実現されます。

WRED の仕組み

図 *77:*



上の図は、WRED の仕組みを示しています。

ルータの背後に存在する送信者は、ネットワーク内の他の場所に存在する受信者にトラフィックを送信します。WREDは、ルータをネットワークに接続するリンク（インターフェイス）で設定されます。すべてのフローの送信レートの合計がリンク容量を超えると、そのインターフェイスに対して設定されたキューへのパケットのバックアップが発生します。

テール ドロップ（342 ページ） で、キューのテール ドロップしきい値について説明しました。WRED は、下限（最小）しきい値を使用して、輻輳がいつ発生しているのかを判断します。キューの深さがこのしきい値に達すると、キュースペースがまだ利用可能であっても、ランダムにパケットがエンキューされずにドロップされます。このドロップのランダム特性により、少数のフローからのみパケットがドロップされることが保証されます。

たとえば、最初にフロー 1 から 1 つのパケットがドロップされたとします。TCP（または任意の弾力性トランスポートメカニズム）は、そのドロップを検出し、そのフローの送信レートを低下させます。これにより、リンク レートが集約送信レートを超えると、キューの深さは減少しはじめます。キューの深さがWREDの最小しきい値を下回ると、WREDはパケットのドロップを中止します。

集約送信レートが依然としてリンク レートを超えている場合、キューの深さは増えつづけ、WREDはパケットをランダムにドロップしつづけます。たとえば、フロー4からパケットがドロップされると、フロー 1 とフロー 4 の両方がバックオフされます。このプロセスは、輻輳を緩和するために十分なストリームがバックオフされるまで継続されます。

WREDのランダム要素により、すべてのフローが同時にバックオフされないことが保証されます。これが実行されると、多くの場合、再び同時に送信レートを上げることが試みられます。その結果、すべての送信者が送信レートを上げ下げする「のこぎり歯効果」が発生します。ドロップするパケットがランダムに選択されることにより、異なるフローを異なるタイミングでバックオフする必要があることを示す信号がランダムに送信されます。

平均キュー深度

WREDのこれまでの説明では、キューの深さが事前に決められたしきい値を超えたときに発生するランダム ドロップについて述べました。実際には、瞬間キュー深度ではなく、抑制された平均キュー深度が使用されます。つまり、テール ドロップのチェックには瞬間キュー深度が使用され、WRED には平均キュー深度が使用されます。

インターネット トラフィックにはバースト性があるため、これは当然のことです。瞬間キュー深度を使用して輻輳をモニタする場合、パケットが早急にドロップされるため、実際の輻輳ではなくトラフィックにおける通常のバーストにも反応してしまう可能性がありました。

平均キュー深度における変動を抑制する方法を決定するために、*WRED*指数加重定数が使用されます。。ルータは、平均キュー深度の現在の値を記憶します。パケットがエンキュー ステージに到達するたびに、瞬間キュー深度が調べられ、平均キュー深度が再計算されます。平均キュー深度の新しい値を計算する式は次のとおりです。

Avg = OldAvg +（Instantaneous - OldAvg）/2 ^ 指数加重定数

ここで、Avg は現在のエンキュー時間に計算された平均キュー深度、Instantaneous は現在のキューの深さ、OldAvgは前回のエンキュー以降に記憶された以前に計算された平均です。

たとえば、OldAvg が 12.0 パケット、Instantaneous が 14 パケット（パケットのキューイング時にモニタ）、指数加重定数が 6（ASR 1000 ルータでのパケット モード WRED のデフォルト）の場合、Avg は次のようになります。

Avg = 12 +（14 - 12）/2 ^ 6 = 12 + 0.03125 = **12.03125**

（注） キューがバイト モードで動作している場合の指数加重定数は 9 です。

その後、別のパケットがエンキューされます。その間にキューの先頭から 1 つのパケットが送信された場合、瞬間キュー深度は 14 のままになります。このとき、Avg の計算は次のようになります。

Avg = 12.03125 +（14 – 12.03125）/2 ^ 6 = 12.03125 + 0.0308 = **12.06201**

この例は、平均キュー深度が抑制されていることを示しています。瞬間キュー深度は、平均をはるかに超えて大きくなる可能性があります。その結果、WRED最大しきい値は常にキュー制限よりかなり小さくなります。この例には、キューの深さがある程度の時間一貫している場合でも平均が瞬間値に収束するために必要な時間も示されています。このように平均キュー深度を抑制することが、WREDがトラフィックの通常のマイクロバーストへの反応を回避する方法です。

指数加重定数の値の変更を検討するには、非常に高度な数学的知識を必要とします。そのようなことは「無駄に専門的なこだわり」であり、行わないでください。指数加重定数を変更するコードは次のようなものですが、これを示すのは完全を期すためだけであり、使用は推奨されません。

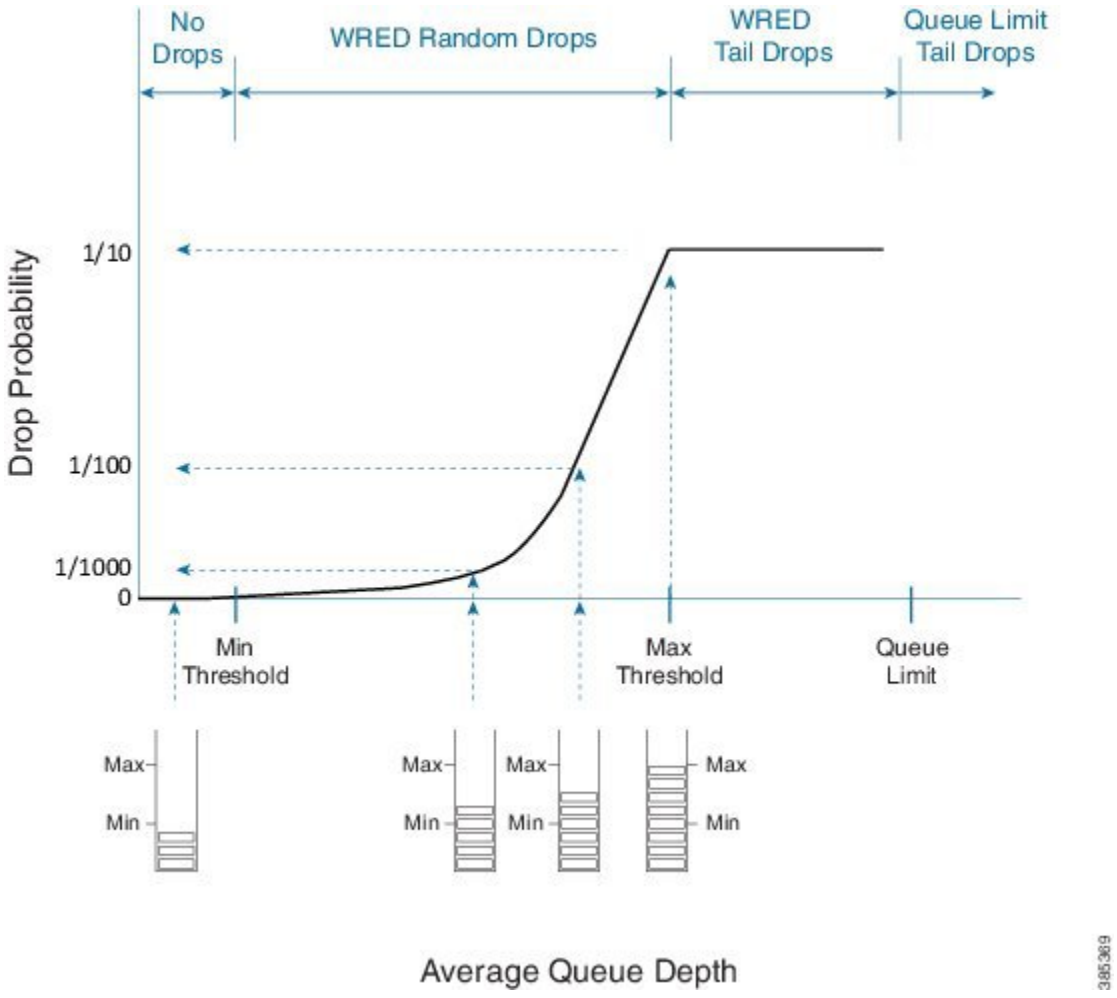
```
policy-map ewc-example
  class class-default
    random-detect
    random-detect exponential-weighting-constant 5
```

WRED しきい値とドロップ曲線

WREDのドロップ判断は、エンキュー時に計算される平均キュー深度を利用して行われます。

WRED を設定するときは、プレシデンス値（または DSCP、discard-class など）ごとに、最小しきい値、最大しきい値、およびドロップ確率を設定します。

次の図は、サンプル プレシデンス値のドロップ曲線を示しています。



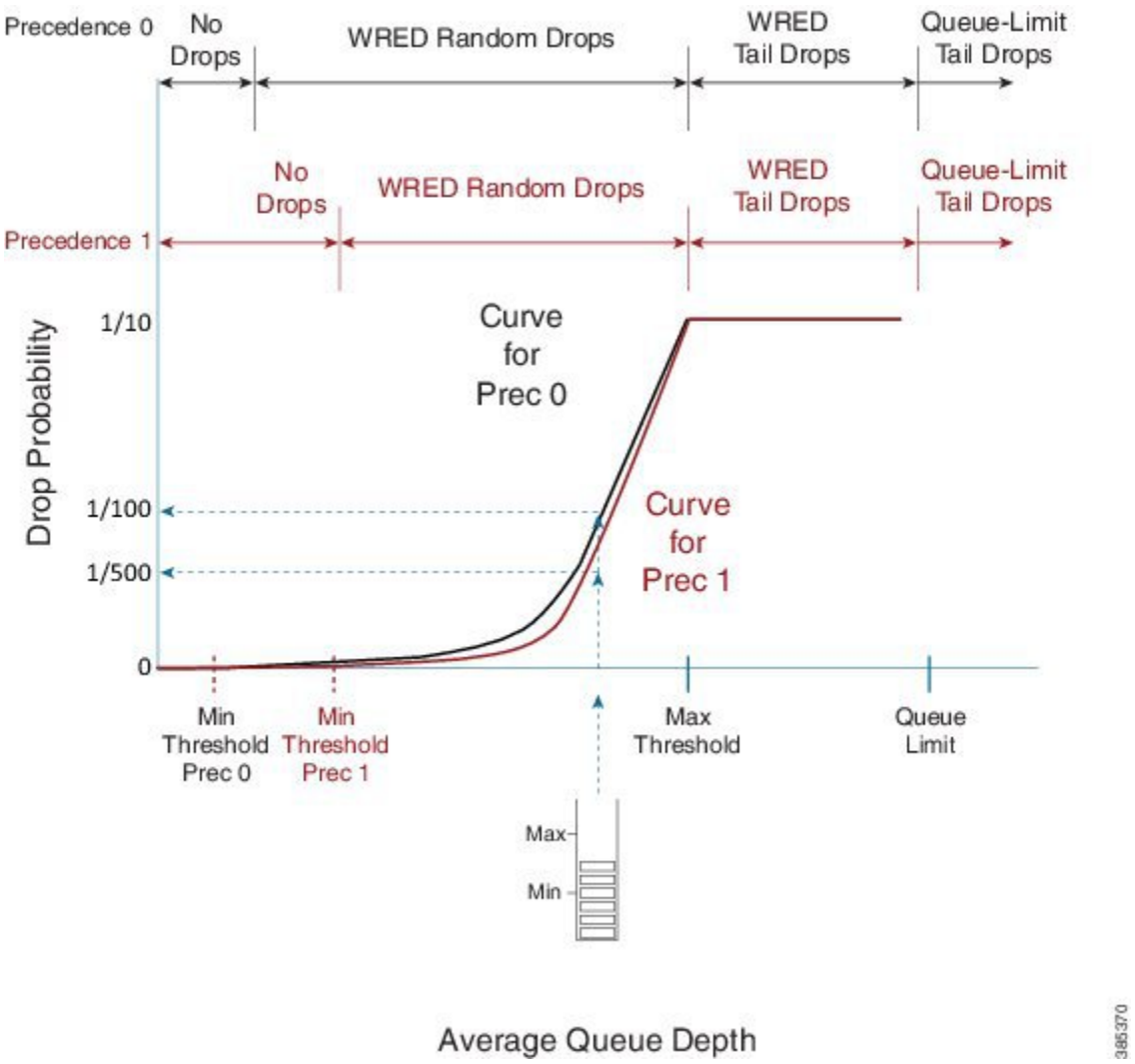
平均キュー深度の計算された値が WRED の最小しきい値よりも小さい場合、その範囲では WRED によってパケットがドロップされません。

最小しきい値と最大しきい値の間の場合、その範囲では WRED によってランダムにドロップされます。最小しきい値（ゼロ）から最大しきい値（設定された WRED ドロップ確率（デフォルトでは 10 パケットに 1 つ））までドロップ確率が指数関数的に上昇していることを確認してください。この曲線は指数関数的なので、平均キュー深度が最小しきい値に近づくと、WRED がドロップするパケットの数は非常に少なくなります。図に反映されているように、平均キュー深度が増えるにつれてドロップ確率も増えます。平均キュー深度が分かれば、対応するドロップ確率も分かります。その後、パケットがドロップされるかエンキューされるかが決定されます。

計算される WRED 平均キュー深度が最大しきい値を超える場合、パケットの *WRED* テール ドロップが発生します。これはキュー制限テール ドロップとは少し異なり、瞬間キュー深度ではなく平均キュー深度が利用されます。瞬間キュー深度がクラスのキュー制限に達すると、ドロップは、WRED テール ドロップではなくキュー制限テール ドロップとして記録されます。

WRED の「W」は加重を表すことに注意してください（一部のトラフィックは他のトラフィックよりも積極的にドロップされる場合があります）。

次に、複数のドロップ曲線を使用する方法を示します。



IP プレシデンス 0 のパケットが着信すると、ルータは黒色の曲線を適用してドロップ確率を計算します。この例では、ドロップ確率は 100 パケットに 1 つとして計算されます。

IP プレシデンス 1 のパケットが着信すると、紫色の曲線が適用されます。同じ平均キュー深度の場合、ドロップ確率がわずかに「500 パケットに 1 つ」であることが分かります。

各プレシデンス値に対してデフォルトの最大しきい値が同じであることに注意してください。各プレシデンス値に対する WRED 最小しきい値の違いは、プレシデンス 0 のトラフィックのドロップが他のトラフィックのドロップよりも前に開始されることを意味します。さらに、ランダム ドロップの範囲では、このトラフィックが特定のキューの深さでより積極的にドロップされます。

（注） WREDを設定すると、ドロップ曲線ごとに、最小しきい値、最大しきい値、およびドロップ確率の適切な値がルータによって選択されます。これらの値は設定されたキュー制限に依存するため、インターフェイス速度が考慮されます。変更による影響を十分に理解していないかぎり、デフォルト値を使用することを強くお勧めします。

WRED：ドロップ曲線の変更

WREDモードに関係なく、個々のドロップ曲線を調整できます。同じコマンドを使用して、そのドロップ曲線の最小しきい値、最大しきい値、または最大しきい値でのドロップ確率を変更できます。最小しきい値および最大しきい値とドロップ確率を使用して、ルータは、平均キュー深度に対するドロップ確率を決定するために必要な指数曲線を作成できます。WREDパラメータの調整は一般的な作業ではありません。調整がそのクラスのアプリケーションに与える影響を十分に理解していないかぎり試みないでください。ほとんどの使用例ではデフォルト値で十分です。

WRED ドロップ曲線を調整する場合は、しきい値をパケット数（デフォルト）、バイト数、または時間で指定できます。WRED設定をクラスに追加する前に、選択した単位でキュー制限を設定する必要があります。また、キューがすでに目的のモードで動作している場合にのみ、その単位のしきい値を変更できます。さらに、WRED がそのモードで動作している場合にかぎり、特定の DSCP、プレシデンス、または discard-class の値の曲線だけを変更できます。

前述のように、ドロップ確率は整数です。平均キュー制限が最大しきい値にある場合、パケットは、その整数値に対して *1* つの確率でドロップされます。たとえば、ドロップ確率が 20 の場合、パケットが WRED によってドロップされる確率は 20 分の 1（5%）です。

ドロップ曲線を変更するためのコマンドは、次に示すように、**random-detect [dscp|precedence|discard-class]** *value min-threshold max-threshold drop-probability* です。

```
policy-map tuneprecedence
  class bulk-data
    bandwidth remaining percent 30
    random-detect
    random-detect precedence 1 1301 2083 10
```

この例では、キューがパケット モード（デフォルト）で動作しており、WRED がプレシデンス モード（デフォルト）で動作している場合に、プレシデンス 1 とプレシデンス 2 の最小しきい値が区別されないように設定します。プレシデンス 1 の曲線を変更し、最小しきい値を 1301、最大しきい値を 2083、最大しきい値でのドロップ確率を 10 パケットに 1 つに設定します。

random-detect precedence 1 1301 2083 10

この場合も、**show policy-map interface** コマンドによって設定を確認できます。

```
show policy-map interface g1/0/0
 GigabitEthernet1/0/0

  Service-policy output: tuneprecedence

    Class-map: bulk-data (match-all)
```

```
0 packets, 0 bytes
5 minute offered rate 0000 bps, drop rate 0000 bps
Match: access-group name bulkdata
Queueing
queue limit 4166 packets
(queue depth/total drops/no-buffer drops) 0/0/0
(pkts output/bytes output) 0/0
bandwidth remaining 30%
  Exp-weight-constant: 4 (1/16)
  Mean queue depth: 1086 packets
  class  Transmitted   Random drop   Tail drop   Minimum   Maximum   Mark
         pkts/bytes    pkts/bytes    pkts/bytes  thresh    thresh    prob

  0         0/0           0/0           0/0       1041      2083      1/10
  1         0/0           0/0           0/0       1301      2083      1/10
  2         0/0           0/0           0/0       1301      2083      1/10
  3         0/0           0/0           0/0       1431      2083      1/10
  4         0/0           0/0           0/0       1561      2083      1/10
  5         0/0           0/0           0/0       1691      2083      1/10
  6         0/0           0/0           0/0       1821      2083      1/10
  7         0/0           0/0           0/0       1951      2083      1/10
```

プレシデンス 1 に新しい値が設定されていることに注意してください。

次に、キューが時間ベース モードで動作しており、WRED が DSCP モードで動作している場合のしきい値の変更について説明します。具体的には、af21 の最小しきい値が af11 の最小しきい値を超えるように設定します。この設定は次のようになります。

```
policy-map tunedscp
  class bulk-data
    bandwidth remaining percent 30
    queue-limit 50 ms
    random-detect dscp-based
    random-detect dscp af21 22 ms 25 ms 10
```

show policy-map interface の出力によって設定を確認します。

```
show policy-map interface g1/0/0
 GigabitEthernet1/0/0

  Service-policy output: tunedscp

    Class-map: bulk-data (match-all)
      148826 packets, 223239000 bytes
      5 minute offered rate 2358000 bps, drop rate 0000 bps
      Match: access-group name bulkdata
      Queueing
      queue limit 50 ms/ 6250000 bytes
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 148826/223239000
      bandwidth remaining 30%

        Exp-weight-constant: 9 (1/512)
        Mean queue depth: 0 ms/ 992 bytes
        dscp    Transmitted     Random drop  Tail drop   Minimum    Maximum     Mark
                pkts/bytes      pkts/bytes   pkts/bytes  thresh     thresh      prob
                                                         ms/bytes   ms/bytes
        af11    96498/144747000   0/0          0/0       21/2734375 25/3125000  1/10
        af21    52328/78492000    0/0          0/0       22/2750000 25/3125000  1/10
```

DSCP ベースの WRED では、そのクラス内で検出された DSCP 値の曲線統計のみが表示されます（モード：プレシデンス、DSCP、discard-class （363 ページ）を参照）。

プライオリティ エンキューの WRED 最大しきい値

WRED：ドロップ曲線の変更（359ページ）では、WRED 曲線の最小しきい値を調整する方法を示しました。別の選択肢として、最大しきい値を修正できます。その際、異なる DSCP 値に対して異なるしきい値を設定すると、輻輳の発生時に常に 1 つのタイプのトラフィックをドロップすることを効果的に指定できます。

Af11 を使用して契約内のバルク データ トラフィックを指定し、af12 を使用して契約外バルク データ トラフィックを指定するとします。輻輳の発生時には、常に af12 よりも af11 を優先的に処理する必要があります。af12 に低い WRED 最大しきい値を指定すると、このトラフィックをドロップする一方で af11 を依然としてエンキューすることが可能です。

次の設定では、af12 の最大しきい値をデフォルトの 624 パケット（この帯域幅で）から 580 パケットに変更します。

```
policy-map maxthreshold
  class bulk-data
    bandwidth percent 30
    random-detect dscp-based
    random-detect dscp af12 468 580 10
```

設定を確認します。

```
show policy-map interface g1/0/0
 GigabitEthernet1/0/0

  Service-policy output: maxthreshold

    Class-map: bulk-data (match-all)
      359826 packets, 539739000 bytes
      5 minute offered rate 7208000 bps, drop rate 0000 bps
      Match: access-group name bulkdata
      Queueing
      queue limit 1249 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 359826/539739000
      bandwidth 30% (300000 kbps)
        Exp-weight-constant: 4 (1/16)
        Mean queue depth: 0 packets
        dscp   Transmitted     Random drop    Tail drop    Minimum   Maximum   Mark
               pkts/bytes      pkts/bytes     pkts/bytes   thresh    thresh    prob

        af11  154689/232033500    0/0            0/0         546       624      1/10
        af12  205137/307705500    0/0            0/0         468       580      1/10
```

設定を見ると、平均キュー深度が 580 パケットを超える場合にすべての af12 パケットに対して *WRED* テール ドロップが行われるものの af11 パケットは依然としてエンキューされることが分かります。

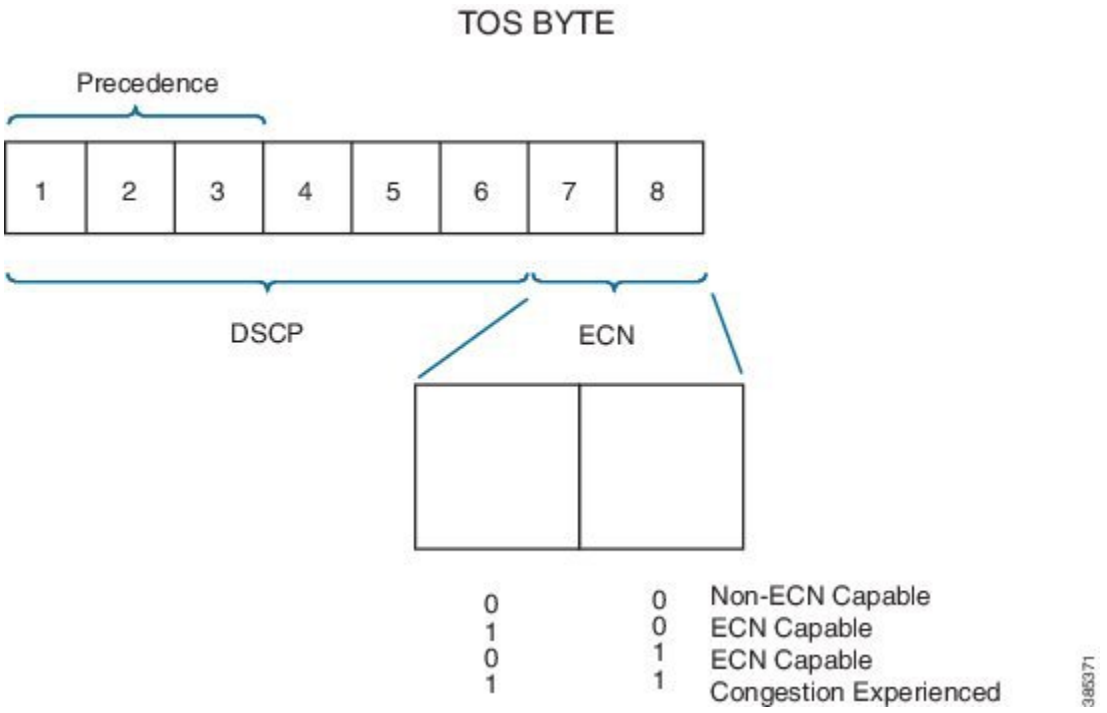
最大しきい値を変更するときは、動作が確実に予期どおりになるように注意してください。この例の場合、輻輳が続き、平均キュー深度が 580 パケットを超えたままになると、輻輳が継続している間にすべてのサービスの af12 トラフィックが完全に枯渇します。

ECN：明示的輻輳通知

明示的輻輳通知（ECN）は、IPプロトコルの拡張機能です。これにより、ネットワークは、パケットを早期にドロップして輻輳を通知するのではなく、パケットをマークしてエンドポイントに輻輳を通知することができます。このようなパケットを受信すると、エンドポイントはその輻輳通知を送信者にエコーバックします。

（注） ECN モードは WRED で明示的にイネーブルにする必要があります。

ECN を理解するには、まず IP ヘッダーの TOS バイトを把握する必要があります。このバイトは、本来は最上位 3 ビットで IP プレシデンス ビットを伝送するために使用されていました。最近では、このバイトの最上位 6 ビットで DSCP コードポイントを伝送するために使用されています。RFC3168 では残りの 2 ビットが ECN ビットとして定義されています。



WRED を ECN モードで設定すると、パケットをドロップする前に ECN ビットを確認するようになります。これらのビットが両方とも 0 に設定されている場合、ルータはエンドポイントが ECN に対応していないと見なし、WRED はパケットをドロップして輻輳の発生を通知します。

どちらかの ECN ビットが 1 に設定されている場合、ルータはエンドポイントが ECN に対応していると見なし、両方の ECN ビットを 1 に設定することによって、パケットをドロップするのではなく発生した輻輳をマークできます。エンドポイントは、上位層プロトコルが本質的に弾力性を持つ場合にのみ、トランスポートが ECN 対応であることを通知する必要があります。

（注） ルータは、パケットをマークするかドロップするかを判断するときに、ECN ビットだけを調べます。

次に、ECN モードで WRED を設定する例を示します。

```
policy-map ecn-example
  class bulk-data
    bandwidth remaining percent 30
    random-detect dscp-based
    random-detect ecn
```

モード：プレシデンス、DSCP、discard-class

WRED プレシデンス モード

WRED しきい値とドロップ曲線 （356 ページ）でドロップ曲線について説明しました。

WREDをイネーブルにすると、デフォルトではプレシデンスモードで動作し、8つの異なるドロップ曲線（有効なプレシデンス値ごとに1つずつ）が作成されます。デフォルトの最小しきい値は、プレシデンス値とともに大きくなります。そのため、プレシデンス0の場合はドロップがプレシデンス1より早く開始され、より積極的に行われます。また、プレシデンス1の場合もドロップがプレシデンス2より早く開始され、より積極的に行われ、以降同様に処理されます。曲線ごとに同じデフォルト最大しきい値とドロップ確率が設定されます。

パケットが着信すると、IP プレシデンス ビットによって、適切なドロップ確率を取得するために使用される曲線が決定されます。パケットが「IP」ではない場合は、プレシデンス 0 のドロップ曲線が使用されます。パケットがMPLSカプセル化されている場合は、EXP ビットがプレシデンス ビットとして扱われ、適切なドロップ曲線が決定されます。

次の例では、WRED がプレシデンス モードでイネーブル化されます。WRED がキューイング アクション（class-default を含む）を持つクラスに存在する必要があることに注意してください。

```
policy-map wred-precedence-example
  class bulk-data
     bandwidth remaining percent 30
     random-detect
     random-detect precedence-based
```

この例では、キューイング アクションとして **bandwidth remaining** コマンドを使用します。**random-detect** コマンドはWREDをクラスbulk-dataでイネーブルにします。また、**random-detect precedence-mode** コマンドは WRED にプレシデンス モードで動作するように指示します。

（注） WREDのデフォルトモードはプレシデンスベースであるため、**random-detect precedence-mode** コマンドはオプションです。

すべての QoS 機能と同様に、設定を確認するための主な手段は **show policy-map interface** コマンドです。

```
show policy-map int g1/0/0
 GigabitEthernet1/0/0

  Service-policy output: wred-precedence-example

   Class-map: bulk-data (match-all)
    6468334 packets, 9702501000 bytes
    5 minute offered rate 204108000 bps, drop rate 0000 bps
    Match: access-group name bulkdata
    Queueing
    queue limit 4166 packets
    (queue depth/total drops/no-buffer drops) 1308/0/0
    (pkts output/bytes output) 6468335/9702502500
    bandwidth remaining 30%
     Exp-weight-constant: 4 (1/16)
     Mean queue depth: 1308 packets
     class     Transmitted        Random drop    Tail drop      Minimum    Maximum
Mark
                 pkts/bytes         pkts/bytes     pkts/bytes     thresh     thresh
prob

     0             0/0             0/0            0/0           1041       2083
1/10
     1             0/0             0/0            0/0           1171       2083
1/10
     2             0/0             0/0            0/0           1301       2083
1/10
     3             0/0             0/0            0/0           1431       2083
1/10
     4       6468335/9702502500    0/0            0/0           1561       2083
1/10
     5             0/0             0/0            0/0           1691       2083
1/10
     6             0/0             0/0            0/0           1821       2083
1/10
     7             0/0             0/0            0/0           1951       2083
1/10
```

プレシデンス モードで作成される 8 つのドロップ曲線のそれぞれについて、統計と曲線設定値がどのように表示されるのかに注目してください。平均キュー深度が最小しきい値より小さいため、ランダム ドロップはレポートされません。

WRED DSCP モード

WRED を設定するための 2 つ目の選択肢は DSCP モードです。このモードでは、64 の一意の曲線が作成されます。

プレシデンス モードと同様に、IP 以外のトラフィックにはデフォルト（DSCP 0）の曲線が使用されます。MPLS トラフィックが着信すると、MPLS EXP ビットがプレシデンス値として扱われ、それに応じて曲線が選択されます（EXP 1 は DSCP CS 1 として扱われ、EXP 2 は CS 2 として扱われ、以降同様）。

次に、DSCP モードで WRED を設定する例を示します。

```
policy-map wred-dscp-example
  class bulk-data
```

```
    bandwidth remaining percent 30
    random-detect dscp-based
```

この場合も、**show policy-map interface** コマンドによって設定を確認できます。

```
show policy-map int
 GigabitEthernet1/0/0

  Service-policy output: wred-dscp-example

    Class-map: bulk-data (match-all)
      5655668 packets, 8483502000 bytes
      5 minute offered rate 204245000 bps, drop rate 0000 bps
      Match: access-group name bulkdata
      Queueing
      queue limit 4166 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 5655669/8483503500
      bandwidth remaining 30%
       Exp-weight-constant: 4 (1/16)
       Mean queue depth: 1 packets
       dscp   Transmitted      Random drop   Tail drop    Minimum   Maximum   Mark
              pkts/bytes       pkts/bytes  pkts/bytes    thresh     thresh    prob
       af11 1205734/1808601000     0/0         0/0         1821      2083      1/10
       cs4  5270109/7905163500     0/0         0/0         1561      2083      1/10
```

2 つの DSCP 値（af11 と cs4）の統計およびドロップ曲線情報だけが表示されることに注意してください。DSCP モードでは、64 の一意のドロップ曲線が設定され、IOS はそのすべての統計を保持します。ただし、実際にトラフィックが確認されたドロップ曲線の情報だけが表示されます。この例では、DSCP af11 および cs4 のトラフィックだけが検出されたため、それが表示されています。

WRED discard-class

discard-class は、qos-group と非常によく似た概念の内部マーキングです。出力時に WRED ドロップ曲線を選択ためにマークを使用するのと同様に、入力時に（出力時にではなく）discard-class をマークすることができます。

場合によっては、パケットでのプレシデンスまたはDSCPマーキングを、出力インターフェイスでの分類に利用できないことがあります。例としては、入力インターフェイスで IP パケットを受信し、出力インターフェイスでMPLSカプセル化パケットを転送するMPLSカプセル化ルータがあります。

DSCP はより少数の EXP 値（DiffServ フィールドの 6 ビットと MPLS ヘッダーの 3 ビット フィールド）にマッピングする必要があるため、ある程度の精度が失われます。af11 を契約内バルク データに使用し、af12 を契約外バルク データに使用するとします。出力インターフェイスでは、DSCP の可視性は失われます。そのため、af11 と af12 は、高い確率で同じ EXP にマップされます。ここでは、出力インターフェイスで af12 よりも af11 を優先的に処理する方法について説明します。

WRED discard-class を使用すると、これを実現できます。これを行うには、次のサンプル ポリシーのように、入力インターフェイスで discard-class をマークする必要があります。

```
policy-map mark-in-contract
  class bulk-data
```

```
police cir 50000000 pir 100000000
conform-action set-dscp-transmit af11
conform-action set-mpls-exp-imposition-transmit 1
conform-action set-discard-class-transmit 2
exceed-action set-dscp-transmit af12
exceed-action set-mpls-exp-imposition-transmit 1
exceed-action set-discard-class-transmit 1
violate-action drop
```

このポリシーでは、CIR に準拠するトラフィックは契約内としてマークされます。

```
conform-action set-dscp-transmit af11
conform-action set-mpls-exp-imposition-transmit 1
conform-action set-discard-class-transmit 2            ****
```

CIR と PIR の間のトラフィックは、契約外としてマークされます。

```
exceed-action set-dscp-transmit af12
exceed-action set-mpls-exp-imposition-transmit 1
exceed-action set-discard-class-transmit 1             ****
```

違反トラフィックはドロップされます。

適合トラフィックと超過トラフィックに対して同じ EXP 値がどのように設定されるのかに注意してください。それらはすべてバルク データ トラフィックであり、MPLS ネットワークで同じ Per-Hop-Behavior を使用します。ただし、契約内トラフィックと契約外トラフィックの場合は、個別の discard-class もマークされます（アスタリスクを参照）。これを出力インターフェイスで使用することにより、優先的な処理が実現されます。

出力インターフェイスで、次のように、WRED を discard-class ベース モードで設定します。

```
policy-map wred-discard-class-example
  class bulk-data
    bandwidth remaining percent 30
    random-detect discard-class-based
```

show policy-map interface コマンドの出力は、次のようなものになります。

```
show policy-map int g1/0/0
 GigabitEthernet1/0/0

  Service-policy output: wred-discard-class-example

    Class-map: bulk-data (match-all)
      1500 packets, 1040000 bytes
      5 minute offered rate 51955000 bps, drop rate 0000 bps
      Match: access-group name bulkdata
      Queueing
      queue limit 4166 packets
      (queue depth/total drops/no-buffer drops) 943/0/0
      (pkts output/bytes output) 1500/1040000
      bandwidth remaining 30%
        Exp-weight-constant: 4 (1/16)
        Mean queue depth: 943 packets
        discard-class  Transmitted  Random drop   Tail drop   Minimum   Maximum  Mark
                       pkts/bytes   pkts/bytes    pkts/bytes  thresh    thresh   prob
```

```
0                0/0             0/0             0/0             1041        2083        1/10
1              500/4000          0/0             0/0             1171        2083        1/10
2             1000/1000000       0/0             0/0             1301        2083        1/10
3                0/0             0/0             0/0             1431        2083        1/10
4                0/0             0/0             0/0             1561        2083        1/10
5                0/0             0/0             0/0             1691        2083        1/10
6                0/0             0/0             0/0             1821        2083        1/10
7                0/0             0/0             0/0             1951        2083        1/10
```

出力を見ると、WRED を discard-class モードで動作させると 8 つのドロップ曲線が作成されることが分かります。上記の設定では、契約内トラフィックは discard-class 2 でマークされ、契約外トラフィックは discard-class 2 でマークされます。

また、discard-class 1 の WRED 曲線の方が最小しきい値が低いことも分かります。つまり、輻輳の発生時には、契約外トラフィックのドロップが契約内トラフィックより早く開始され、より積極的に行われます。

明示的に設定された discard-class を持たない（discard-class が明示的に設定されていない）トラフィックは discard-class 0 と見なされます。

コマンド リファレンス：random detect

random-detect *options* コマンドを使用すると、WRED の動作をイネーブルにして制御することができます。次のように、さまざまなオプションを適用できます。

WRED のイネーブル化：次のいずれかを使用します。

random-detect

WRED をプレシデンス モードでイネーブルにします。

random-detect precedence-based

WRED をプレシデンス モードでイネーブルにします。

random-detect dscp-based

WRED を DSCP モードでイネーブルにします。

random-detect discard-class-based

WRED を discard-class モードでイネーブルにします。

WRED ドロップ曲線の調整：次のいずれかを使用します。

random-detect precedence *value min-threshold max-threshold drop-probability*

特定のプレシデンス値のドロップ曲線を修正します。

random-detect dscp *value min-threshold max-threshold drop-probability*

特定の DSCP 値のドロップ曲線を修正します。

random-detect precedence *value min-threshold max-threshold drop-probability*

特定の discard-class 値のドロップ曲線を修正します。min-threshold と max-threshold は、パケット数（デフォルト）、バイト数、または時間で設定できます。単位としてバイト数ま

たは時間を使用するには、まず、**queue-limit** コマンドを使用してキューをそのモード用に設定する必要があります。

WRED 指数加重定数の変更

random-detect exponential-weighting-constant *value*

明示的輻輳通知のサポートののイネーブル化

random-detect ecn

使用方法：

random-detect コマンドは、**bandwidth** コマンド、**bandwidth remaining** コマンド、または **shape** コマンドで設定されたキューイング クラスで使用できます。これには暗黙の帯域幅残存値を持つ class-default が含まれます。

ASR 1000 シリーズ アグリゲーション サービス ルータには、スケジューリング階層の親レベルまたは親の親レベルのキューがありません。そのため、**random-detect** コマンドは、子キューイング ポリシーを含むクラスではサポートされません。

WRED最小しきい値および最大しきい値のデフォルト値は、クラスのキュー制限に比例し、そのためキューの予想サービス レートに比例します。変更がそのクラスのアプリケーションに与える影響を十分に理解していない場合は、WRED ドロップ曲線を修正しないでください。