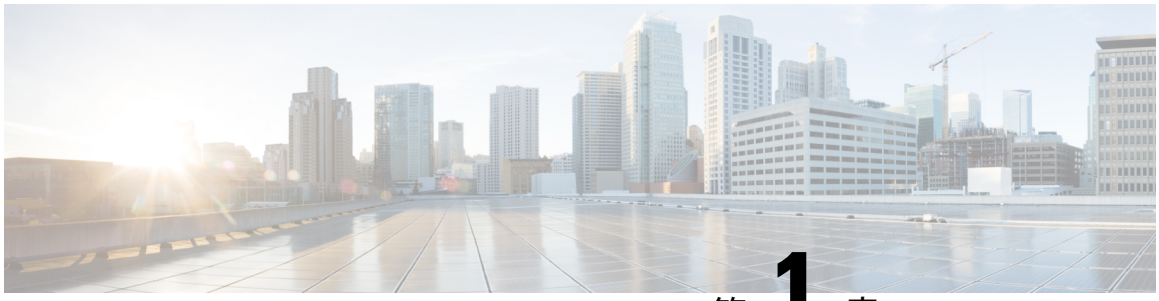
Cisco IOS リリース 15.2(7)Ex（Catalyst 1000 スイッチ）QoS コンフィギュレーション ガイド

初版：2019 年 12 月 25 日

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/

【注意】シスコ製品をご使用になる前に、安全上の注意（**www.cisco.com/jp/go/safety_warning/**）をご確認ください。本書は、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。また、契約等の記述については、弊社販売パートナー、または、弊社担当者にご確認ください。

目 次



第 1 章

自動 QoS の設定

自動 QoS の前提条件

標準 QoS または自動 QoS を設定する前に、次の事項を十分に理解しておく必要があります。

- 使用するアプリケーションのタイプおよびネットワークのトラフィック パターン
- トラフィックの特性およびネットワークのニーズ。バースト性の高いトラフィックかどうかの判別。音声およびビデオ ストリーム用の帯域幅確保の必要性
- ネットワークの帯域幅要件および速度
- ネットワーク上の輻輳発生箇所

自動 QoS の制約事項

次は、自動 QoS（auto-QoS）に対する制限です。

- ポート間でのポリシーマップ（ポリサー/マーキング）の TCAM（Ternary Content Addressable Memory）共有はサポートされていません。このため、自動 QoS/QoS ポリシーマップを適用できるインターフェイスの数は制限されています。
- セキュリティ アクセスコントロールリスト（ACL）とポリシーマップで使用される ACL の両方に同じ TCAM リージョンを使用する必要があります。

- **match ip dscp** コマンドを使用するポリシーマップは、IPv4 アドレスと IPv6 アドレスの両方に一致するため、スケール数は 16 クラスマップに制限されます。IPv4 用に 1 つの TCAM エントリが作成され、IPv6 用にも 1 つの TCAM エントリが作成されます。
- ASIC ごとに、8 つの TCP ポート比較演算子と 8 つの UDP ポート比較演算子がサポートされ、各 gt（より大きい）、lt（より小さい）、neq（等しくない）演算子は 1 つのポート比較演算子を使用し、各範囲演算子は 2 つのポート比較演算子を使用します。この組み合わせを使用するポリシーマップは、TCAM スケールおよびポリシーマップをアタッチできるインターフェイスの数に影響します。
- 次の制限事項が、IPv4 ACL ネットワーク インターフェイスに適用されます。
  - インターフェイスへのアクセスを制御する場合、名前付き ACL または番号付き ACL を使用できます。
  - レイヤ 3 インターフェイスに ACL が適用され、スイッチ上でルーティングがイネーブルになっていない場合は、SNMP、Telnet、Web トラフィックなど、CPU で処理されるパケットだけがフィルタ処理されます。
  - レイヤ 2 インターフェイスに ACL を適用する場合、ルーティングをイネーブルにする必要はありません。
- deny ACL は QoS ポリシーマップではサポートされません。

自動 QoS の設定に関する情報

このセクションでは、自動 QoS の設定について説明します。

自動 QoS の概要

自動 QoS 機能を使用して、QoS 機能の配置を容易にできます。自動 QoS は、ネットワーク設計を確認し、スイッチがさまざまなトラフィック フローに優先度を指定できるように QoS 設定をイネーブルにします。自動 QoS は、デフォルト（ディセーブル）の QoS 動作を使用せずに、出力キューを使用します。スイッチはパケットの内容やサイズに関係なく、各パケットにベストエフォート型のサービスを提供し、単一キューからパケットを送信します。

自動 QoS をイネーブルにすると、トラフィック タイプおよび入力パケット ラベルに基づいてトラフィックを自動的に分類します。スイッチは分類した結果を使用して適切な出力キューを選択します。

自動 QoS コマンドを使用して、次のシスコ デバイスに接続しているポートを識別できます。

- Cisco IP Phone
- Cisco SoftPhone アプリケーションを実行しているデバイス
- Cisco TelePresence
- Cisco IP Camera

- Cisco Digital Media Player

また、auto-QoS コマンドを使用してアップリンクを介して信頼のおけるトラフィックを受信するポートを指定します。自動 QoS は次の機能を実行します。

- 条件付きで信頼できるインターフェイスによる自動 QoS デバイスの有無の検出
- QoS 分類の設定
- 出力キューの設定

生成された自動 QoS 設定

デフォルトでは、自動 QoS はすべてのポートでディセーブルです。パケットは変更されません。つまり、パケットの CoS 値、DSCP 値、および IP precedence 値は変更されません。

インターフェイスの最初のポートで自動 QoS 機能をイネーブルにすると、次のようになります。

- 入力パケット ラベルを使用して、トラフィックの分類、パケット ラベルの割り当て、入力キューと出力キューの設定が行われます。
- QoS はグローバルにイネーブル（**mls qos** グローバル コンフィギュレーション コマンド）になり、他のグローバル コンフィギュレーション コマンドが自動的に生成されます。（例：グローバルな自動 QoS 設定 （10 ページ） を参照）。
- スイッチで信頼境界の機能がイネーブルになり、サポートされているデバイスを検出するために Cisco Discovery Protocol が使用されます。
- パケットがプロファイル内にあるかプロファイル外にあるかを判断するためにポリシングが使用され、パケット上のアクションが指定されます。

VOIP デバイスの詳細

以下のアクティビティは、これらの自動 QoS コマンドをポート上で実行する場合に発生します。

- Cisco IP Phone に接続されたネットワークエッジのポートで **auto qos voip cisco-phone** コマンドを入力すると、スイッチは信頼境界機能をイネーブルにします。パケットに 24、26、または 46 という DSCP 値がない場合、またはパケットがプロファイル外にある場合、スイッチは DSCP 値を 0 に変更します。Cisco IP Phone が存在しない場合、入力分類はパケットの QoS ラベルを信用しないように設定されます。ポリシングは、スイッチが信頼境界機能をイネーブルにする前に、ポリシーマップの分類に一致するトラフィックに適用されます。
- **auto qos voip cisco-softphone** インターフェイス コンフィギュレーション コマンドを、Cisco SoftPhone を稼働するデバイスに接続されたネットワークのエッジのポートに入力すると、スイッチはポリシングを使用して、パケットがプロファイル内にあるかプロファイル外にあるかを判断し、パケット上のアクションを指定します。パケットに 24、26、または 46

という DSCP 値がない場合、またはパケットがプロファイル外にある場合、スイッチは DSCP 値を 0 に変更します。

- ネットワーク内部に接続されたポート上で **auto qos voip trust** インターフェイス コンフィギュレーション コマンドを入力すると、非ルーテッド ポートの場合は入力パケット内の CoS 値、ルーテッドポートの場合は入力パケット内の DSCP 値が信頼されます（前提条件は、トラフィックがすでに他のエッジデバイスによって分類されていることです）。

表 *1:* トラフィック タイプ、パケット ラベル、およびキュー

	VoIP データ トラフィック	VoIP コントロール トラフィック	ルーティング プロトコル トラフィック	STP BPDU トラフィック	リアルタイム ビデオ トラフィック	その他すべてのトラフィック	
DSCP の値	46	24、26	48	56	34	–	
CoS 値	5	3	6	7	3	–	
CoS から出力キューへのマッピング	4、5（キュー 1）	2、3、6、7（キュー 2）			0（キュー 2）	2（キュー 3）	0、1（キュー 4）

- **auto qos voip cisco-phone**、**auto qos voip cisco-softphone** または **auto qos voip trust** インターフェイス コンフィギュレーション コマンドを使用して Auto-QoS をイネーブルにすると、スイッチはトラフィックタイプと入力パケットラベルに基づいて自動的に QoS 設定を生成し、例：グローバルな自動 QoS 設定 （10 ページ） に示されるコマンドをポートに適用します。

ビデオ、信頼、および分類用の拡張自動 QoS

自動 QoS は、ビデオをサポートするように拡張されました。ここでは、Cisco TelePresence System と Cisco IP Camera からのトラフィックを分類して信頼する自動設定が生成されます。

自動 QoS 設定の移行

レガシー自動 QoS から拡張自動 QoS への自動 QoS 設定の移行は、次の場合に発生します。

- スイッチが 12.2(55)SE イメージで起動されます。QoS はディセーブルです。

  インターフェイス上のいずれかのビデオまたは音声の信頼設定によって、拡張自動 QoS コマンドが自動的に生成されます。

- スイッチが QoS でイネーブルになっている場合（次のガイドラインが適用されます）。

- 音声デバイスで条件付き信頼用にインターフェイスを設定すると、レガシー自動QoS VoIP 設定だけが生成されます。
- ビデオ デバイスで条件付き信頼用にインターフェイスを設定すると、拡張自動 QoS VoIP 設定が生成されます。
- 新しいインターフェイスの自動QoSコマンドに基づいて分類または条件き信頼でインターフェイスを設定すると、拡張自動 QoS 設定が生成されます。

- **auto qos srnd4** グローバル コンフィギュレーション コマンドがイネーブルの際に、新しいデバイスを接続すると自動 QoS の移行が発生する場合。

（注） レガシー自動 QoS で以前に設定したインターフェイスが拡張自動 QoS に移行すると、新しいグローバル QoS コマンドに合わせて音声コマンドと設定が更新されます。

拡張自動 QoS からレガシー自動 QoS への自動 QoS 設定の移行が行われるのは、インターフェイスから既存の自動 QoS 設定をすべてディセーブルにした場合だけです。

自動 QoS 設定時の注意事項

自動 QoS を設定する前に、次の事項を確認してください。

- auto-QoS をイネーブルにした後、名前に *AutoQoS* を含むポリシー マップやを変更しないでください。ポリシーマップやを変更する必要がある場合、そのコピーを作成し、コピーしたポリシー マップやを変更します。生成したポリシー マップではなくこの新しいポリシー マップを使用するには、生成したポリシー マップをインターフェイスから削除し、新しいポリシー マップをインターフェイスに適用します。
- auto-QoS のデフォルトを利用するには、auto-QoS をイネーブルにしてから、その他の QoS コマンドを設定する必要があります。必要に応じて QoS 設定を微調整できますが、自動 QoS が完了した後にのみ調整することを推奨します。
- 自動 QoS は、スタティック アクセス、ダイナミックアクセス、音声 VLAN アクセス、およびトランク ポートでイネーブルにできます。
- デフォルトでは、CDP 機能はすべてのポート上でイネーブルです。自動 QoS が適切に動作するために、CDP をディセーブルにしないでください。

自動 QoS VoIP に関する考慮事項

自動 QoS VoIP を設定する前に、次の事項を確認してください。

- 自動 QoS は、非ルーテッド ポートおよびルーテッド ポートで Cisco IP Phone に VoIP のスイッチを設定します。また、自動 QoS は Cisco SoftPhone アプリケーションを稼働するデバイスの VoIP 用にスイッチを設定します。

（注） Cisco SoftPhone を稼働するデバイスが非ルーテッドポートまたはルーテッドポートに接続されている場合、スイッチはポート単位で Cisco SoftPhone アプリケーション 1 つのみをサポートします。

- ルーテッド ポートで Cisco IP Phone の自動 QoS をイネーブルにすると、スタティック IP アドレスを IP Phone に割り当てます。
- このリリースは、Cisco IP SoftPhone Version 1.3(3) 以降のみをサポートします。
- 接続される装置は Cisco Call Manager バージョン 4 以降を使用する必要があります。

拡張された自動 QoS に関する考慮事項

自動 QoS は、ビデオをサポートするように拡張されました。ここでは、Cisco TelePresence System と Cisco IP Camera からのトラフィックを分類して信頼する自動設定が生成されます。

拡張自動 QoS を設定する前に、次の事項を確認してください。

- **auto qos srnd4**グローバル コンフィギュレーション コマンドは、拡張自動 QoS 設定の結果として生成されます。

実行コンフィギュレーションでの自動 QoS の影響

自動 QoS がイネーブルになると、**auto qos** インターフェイス コンフィギュレーション コマンドおよび生成されたグローバル コンフィギュレーションが実行コンフィギュレーションに追加されます。

スイッチは、自動 QoS が生成したコマンドを、CLI から入力したように適用します。既存のユーザ設定では、生成されたコマンドの適用に失敗することがあります。また、生成されたコマンドで既存の設定が上書きされることもあります。これらのアクションが警告なしで発生する可能性があります。生成されたコマンドがすべて正常に適用された場合、上書きされなかったユーザ入力の設定は実行コンフィギュレーション内に残ります。上書きされたユーザ入力の設定は、現在の設定をメモリに保存せずに、スイッチをリロードすると復元できます。生成コマンドが適用されなかった場合、以前の実行コンフィギュレーションが復元されます。

自動 QoS の設定方法

このセクションでは、自動 QoS の設定方法について説明します。

自動 QoS のイネーブル化

QoS パフォーマンスを最適化するには、ネットワーク内のすべてのデバイスで自動 QoS をイネーブルにします。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**interface***interface-id*
	例：
	`Device(config)# interface gigabitethernet 1/0/1`
	Or
	`Device(config)# interface fastethernet 1/0/1`
		ビデオデバイスに接続されたポートか、またはネットワーク内部の別の信頼できるスイッチまたはルータに接続されたアップリンク ポートを指定し、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。

ステップ 3	次のいずれかを使用します。 • **auto qos voip** {**cisco-phone** | **cisco-softphone** | **trust**} • **auto qos video** {**cts** | **ip-camera** | **media-player**} • **auto qos classify** [**police**] • **auto qos trust** {**cos** | **dscp**} 例： `Device(config-if)# auto qos trust dscp`	VoIP 用の自動 QoS を有効にします。 • **cisco-phone**：ポートが Cisco IP 電話に接続されている場合、着信パケットの QoS ラベルは電話が検出された場合のみ信頼されます。 • **cisco-softphone**：ポートが Cisco SoftPhone 機能を実行するデバイスに接続されています。 • **trust**：アップリンクポートが信頼性のあるスイッチまたはルータに接続されていて、入力パケットの VoIP トラフィック分類が信頼されています。 ビデオ デバイス用の自動 QoS を有効にします。 • **cts**：Cisco Telepresence System に接続されているポート。 • **ip-camera**：Cisco ビデオ監視カメラに接続されているポート。 • **media-player**：CDP 対応 Cisco Digital Media Player に接続されているポート。

	コマンドまたはアクション	目的
		着信パケットの QoS ラベルが信頼されるのは、システムが検知される場合に限ります。 分類用の自動 QoS を有効にします。 • **police**：QoS ポリシーマップを定義し、それらをポートに適用してポリシングを設定します（ポートベースの QoS）。 信頼できるインターフェイス用の自動 QoS を有効にします。 • **cos**：サービスクラス。 • **dscp**：DiffServ コードポイント。 • <cr>：信頼インターフェイス。
ステップ 4	**exit** 例： `Device(config-if)# exit`	グローバル コンフィギュレーション モードに戻ります。

ステップ 5	**interface***interface-id*
	例：
	`Device(config)# interface gigabitethernet 1/0/1`
	Or
	`Device(config)# interface fastethernet 1/0/1`
		信頼できるスイッチまたはルータに接続されていると識別されたスイッチ ポートを指定し、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。

ステップ 6	**auto qos trust** 例： `Device(config-if)# auto qos trust`	ポートで自動 QoS を有効にし、そのポートが信頼できるルータまたはスイッチに接続されるように指定します。
ステップ 7	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show auto qos interface***interface-id* 例：	入力を確認します。

	コマンドまたはアクション	目的

	`Device(config)# show auto qos interface gigabitethernet 1/0/1`
	Or
	`Device(config)# show auto qos interface fastethernet 1/0/1`
		このコマンドは、自動 QoS がイネーブルであるインターフェイス上の自動QoSコマンドを表示します。自動 QoS 設定およびユーザの変更を表示するには、**show running-config** 特権 EXEC コマンドを使用します。

自動 QoS に関するトラブルシューティング

自動 QoS のトラブルシューティングを行うには、**debug auto qos** 特権 EXEC コマンドを使用します。詳細については、このリリースに対応するコマンドリファレンスにある **debug auto qos** コマンドを参照してください。

ポートで自動 QoS を無効にするには、**auto qos** インターフェイス コンフィギュレーション コマンドの **no** 形式（**no auto qos voip** など）を使用します。このポートに対して、auto-QoS によって生成されたインターフェイス コンフィギュレーション コマンドだけが削除されます。auto-QoS をイネーブルにした最後のポートで、**no auto qos voip** コマンドを入力すると、auto-QoS によって生成されたグローバル コンフィギュレーション コマンドが残っている場合でも、auto-QoS はディセーブルと見なされます（グローバル コンフィギュレーションによって影響を受ける他のポートでのトラフィックの中断を避けるため）。

自動 QoS の監視

表 *2:* 自動 *QoS* の監視用コマンド

コマンド	説明
show auto qos [**interface** [*interface-type*]]	最初の自動 QoS 設定を表示します。 **show auto qos** コマンド出力と **show running-config** コマンド出力を比較してユーザ定義の QoS 設定を比較できます。
show mls qos [**interface** | **maps**]	自動 QoS によって影響されるかもしれない QoS 設定に関する情報を表示します。
show mls qos interface [*interface-type* | **queueing** | **statistics**]	自動 QoS によって影響されるかもしれない QoS インターフェイス設定に関する情報を表示します。
show mls qos maps [**cos-output-q** | **dscp-mutation** | **dscp-output-q**]	自動 QoS によって影響されるかもしれない QoS マップ設定に関する情報を表示します。

コマンド	説明
show running-config	自動 QoS によって影響されるかもしれない QoS 設定に関する情報を表示します。 **show auto qos** コマンド出力と **show running-config** コマンド出力を比較してユーザ定義の QoS 設定を比較できます。

自動 QoS の構成例

次のセクションに自動 QoS の構成例を示します。

例：グローバルな自動 QoS 設定

次の表は、自動 QoS および拡張自動 QoS に対してスイッチによって自動的に生成されたコマンドを説明しています。

表 *3:* 生成された自動 *QoS* 設定

説明	自動的に生成されるコマンド **{voip}**

スイッチが、自動的に CoS 値を出力キューおよびしきい値 ID にマッピングします。	`Device(config)# no mls qos srr-queue output cos-map`
	`Device(config)# mls qos srr-queue output cos-map queue 1 threshold 1 4`
	`Device(config)# mls qos srr-queue output cos-map queue 2 threshold 1 2 6 7 6 7`
	`Device(config)# mls qos srr-queue output cos-map queue 2 threshold 2 3 4`
	`Device(config)# mls qos srr-queue output cos-map queue 3 threshold 2 0`
	`Device(config)# mls qos srr-queue output cos-map queue 4 threshold 2 1`

説明	自動的に生成されるコマンド{voip}
スイッチが、自動的に DSCP 値を出力キューおよびしきい値 ID にマッピングします。	Device(config)# **no mls qos srr-queue output dscp-map** Device(config)# **mls qos srr-queue output dscp-map queue 1 threshold 2 32 33 40 41 42 43 44 45** Device(config)# **mls qos srr-queue output dscp-map queue 1 threshold 2 46 47** Device(config)# **mls qos srr-queue output dscp-map queue 2 threshold 1 16 17 18 19 20 21 22 23** Device(config)# **mls qos srr-queue output dscp-map queue 2 threshold 1 26 27 28 29 30 31 34 35** Device(config)# **mls qos srr-queue output dscp-map queue 2 threshold 1 36 37 38 39** Device(config)# **mls qos srr-queue output dscp-map queue 2 threshold 2 24 48 49 50 51 52 53 54** Device(config)# **mls qos srr-queue output dscp-map queue 2 threshold 2 55 56 57 58 59 60 61 62** Device(config)# **mls qos srr-queue output dscp-map queue 2 threshold 2 63** Device(config)# **mls qos srr-queue output dscp-map queue 3 threshold 1 0 1 2 3 4 5 6 7** Device(config)# **mls qos srr-queue output dscp-map queue 4 threshold 1 8 9 10 11 12 13 14 15**

例：拡張されたビデオ、信頼、および分類デバイス用に自動 QoS で生成される設定

次の拡張自動 QoS コマンドを入力すると、スイッチは CoS/DSCP のマッピングを設定します（着信パケットの CoS 値を DSCP 値にマップします）。

- **auto qos video cts**
- **auto qos video ip-camera**
- **auto qos video media-player**
- **auto qos trust**
- **auto qos trust cos**
- **auto qos trust dscp**

auto qos classify コマンドを入力すると、スイッチが自動的にクラスマップおよびポリシーマップを作成します（以下を参照）。

```
Device(config)# policy-map AUTOQOS-SRND4-CLASSIFY-POLICY
Device(config-pmap)# class AUTOQOS_MULTIENHANCED_CONF_CLASS
Device(config-pmap-c)# set dscp af41
Device(config-pmap)# class AUTOQOS_BULK_DATA_CLASS
Device(config-pmap-c)# set dscp af11
Device(config-pmap)# class AUTOQOS_TRANSACTION_CLASS
Device(config-pmap-c)# set dscp af21
Device(config-pmap)# class AUTOQOS_SCAVANGER_CLASS
Device(config-pmap-c)# set dscp cs1
Device(config-pmap)# class AUTOQOS_SIGNALING_CLASS
Device(config-pmap-c)# set dscp cs3
Device(config-pmap)# class AUTOQOS_DEFAULT_CLASS
Device(config-pmap-c)# set dscp default
;
Device(config-if)# service-policy input AUTOQOS-SRND4-CLASSIFY-POLICY
```

auto qos classify police コマンドを入力すると、スイッチが自動的にクラス マップおよびポリシー マップを作成します（以下を参照）。

```
Device(config)# class-map match-all AUTOQOS_MULTIENHANCED_CONF_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-MULTIENHANCED-CONF
Device(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
Device(config)# class-map match-all AUTOQOS_TRANSACTION_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-TRANSACTIONAL-DATA
Device(config)# class-map match-all AUTOQOS_SIGNALING_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-SIGNALING
Device(config)# class-map match-all AUTOQOS_BULK_DATA_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-BULK-DATA
Device(config)# class-map match-all AUTOQOS_SCAVANGER_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-SCAVANGER
Device(config)# policy-map AUTOQOS-SRND4-CLASSIFY-POLICE-POLICY
Device(config-pmap)# class AUTOQOS_MULTIENHANCED_CONF_CLASS
Device(config-pmap-c)# set dscp af41
Device(config-pmap)# police 5000000 8000 exceed-action drop
Device(config-pmap)# class AUTOQOS_BULK_DATA_CLASS
Device(config-pmap-c)# set dscp af11
Device(config-pmap)# class AUTOQOS_TRANSACTION_CLASS
Device(config-pmap-c)# set dscp af21
Device(config-pmap)# class AUTOQOS_SCAVANGER_CLASS
Device(config-pmap-c)# set dscp cs1
Device(config-pmap)# police 10000000 8000 exceed-action drop
Device(config-pmap)# class AUTOQOS_SIGNALING_CLASS
Device(config-pmap-c)# set dscp cs3
Device(config-pmap)# police 32000 8000 exceed-action drop
Device(config-pmap)# class AUTOQOS_DEFAULT_CLASS
Device(config-pmap-c)# set dscp default
```

これは、**auto qos voip cisco-phone** コマンドの拡張コンフィギュレーションです。

```
Device(config)# class-map match-all AUTOQOS_VOIP_DATA_CLASS
Device(config-cmap)# match ip dscp ef
Device(config)# class-map match-all AUTOQOS_VOIP_VIDEO_CLASS
Device(config-cmap)# match ip dscp af41
Device(config)# class-map match-all AUTOQOS_VOIP_SIGNAL_CLASS
Device(config-cmap)# match ip dscp cs3
Device(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
Device(config)# policy-map AUTOQOS-SRND4-CISCOPHONE-POLICY
Device(config-pmap)# class AUTOQOS_VOIP_DATA_CLASS
```

```
Device(config-pmap-c)# set dscp ef
Device(config-pmap)# class AUTOQOS_VOIP_VIDEO_CLASS
Device(config-pmap-c)# set dscp af41
Device(config-pmap)# class AUTOQOS_VOIP_SIGNAL_CLASS
Device(config-pmap-c)# set dscp cs3
Device(config-pmap)# class AUTOQOS_DEFAULT_CLASS
Device(config-pmap-c)# set dscp default
Device(config-if)# service-policy input AUTOQOS-SRND4-CISCOPHONE-POLICY
```

これは、**auto qos voip cisco-softphone** コマンドの拡張コンフィギュレーションです。

```
Device(config)# class-map match-all AUTOQOS_VOIP_DATA_CLASS
Device(config-cmap)# match ip dscp ef
Device(config)# class-map match-all AUTOQOS_MULTIENHANCED_CONF_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-MULTIENHANCED-CONF
Device(config)# class-map match-all AUTOQOS_BULK_DATA_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-BULK-DATA
Device(config)# class-map match-all AUTOQOS_TRANSACTION_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-TRANSACTIONAL-DATA
Device(config)# class-map match-all AUTOQOS_SCAVANGER_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-SCAVANGER
Device(config)# class-map match-all AUTOQOS_SIGNALING_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-SIGNALING

Device(config)# policy-map AUTOQOS-SRND4-SOFTPHONE-POLICY
Device(config-pmap)# class AUTOQOS_VOIP_DATA_CLASS
Device(config-pmap-c)# set dscp ef
Device(config-pmap)# class AUTOQOS_MULTIENHANCED_CONF_CLASS
Device(config-pmap-c)# set dscp af41
Device(config-pmap)#police 5000000 8000 exceed-action drop
Device(config-pmap)#class AUTOQOS_BULK_DATA_CLASS
Device(config-pmap-c)# set dscp af11
Device(config-pmap)# class AUTOQOS_TRANSACTION_CLASS
Device(config-pmap-c)# set dscp af21
Device(config-pmap)# class AUTOQOS_SCAVANGER_CLASS
Device(config-pmap-c)# set dscp cs1
Device(config-pmap)# police 10000000 8000 exceed-action drop
Device(config-pmap)# class AUTOQOS_SIGNALING_CLASS
Device(config-pmap-c)# set dscp cs3
Device(config-pmap)# police 32000 8000 exceed-action drop
Device(config-pmap)# class AUTOQOS_DEFAULT_CLASS
Device(config-pmap-c)# set dscp default
;
Device(config-if)# service-policy input AUTOQOS-SRND4-SOFTPHONE-POLICY
```

自動 QoS に関する追加情報

関連資料

関連項目	マニュアル タイトル
この章で使用するコマンドの完全な構文および使用方法の詳細。	『*Cisco IOS Quality of Service Solutions Command Reference*』

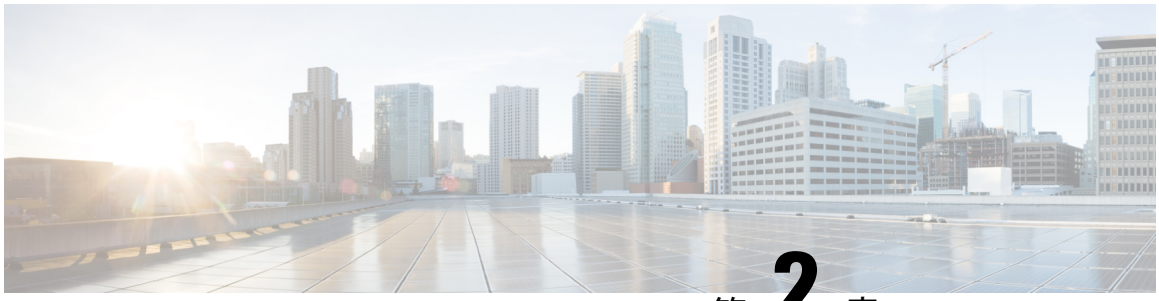
自動 QoS の機能履歴と情報

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 *4*: 自動 *QoS* の設定に関する機能情報

機能名	リリース	機能情報
自動 QoS	Cisco IOS Release 15.2(7)E1	この機能が導入されました。



第 2 章

QoS の設定

QoS の前提条件

標準 QoS を設定する前に、次の事項を十分に理解しておく必要があります。

- 使用するアプリケーションのタイプおよびネットワークのトラフィック パターン
- トラフィックの特性およびネットワークのニーズ。たとえば、ネットワークのトラフィックがバーストであるかどうか。音声およびビデオ ストリーム用の帯域幅確保の必要性
- ネットワークの帯域幅要件および速度
- ネットワーク上の輻輳発生箇所

QoS ACL の注意事項

アクセス コントロール リスト（ACL）を使用して QoS 設定する場合は、次のガイドラインに従ってください。

- IP フラグメントと設定されている IP 拡張 ACL を照合することによって、QoS を実施することはできません。IP フラグメントはベストエフォート型として送信されます。IP フラグメントは IP ヘッダーのフィールドで示されます。
- 1 つのクラス マップごとに、使用できる ACL は 1 つだけであり、使用できる **match** クラスマップ コンフィギュレーション コマンドは 1 つだけです。ACL には、フィールドとパケットの内容を照合する ACE を複数指定できます。

ポリシングの注意事項

- 入力ポートでは 1 つのパケットに適用できるポリサーは 1 つだけです。設定できるのは、平均レート パラメータおよび認定バースト パラメータだけです。
- QoS 対応として設定されているポートを介して受信したすべてのトラフィックは、そのポートに結合されたポリシーマップに基づいて分類、ポリシング、およびマーキングが行われます。QoS が設定されたトランク ポートでは、そのポートを通じて受信されるすべての VLAN 内トラフィックは、ポートに付加されたポリシー マップに従って分類、ポリシング、およびマーキングが行われます。
- スイッチ上で EtherChannel ポートが設定されている場合、EtherChannel を形成する個々の物理ポートに QoS の分類、ポリシング、マッピング、およびキューイングを設定する必要があります。また、QoS の設定を EtherChannel のすべてのポートで照合するかどうかを決定する必要があります。
- 既存の QoS ポリシーのポリシー マップを変更する必要がある場合は、最初にすべてのインターフェイスからポリシーマップを削除し、その後ポリシーマップを変更またはコピーします。変更が終了したら、変更したポリシーマップをインターフェイスに適用します。最初にすべてのインターフェイスからポリシー マップを削除しなかった場合、CPU 使用率が高くなり、コンソールが長期間停止する可能性があります。

一般的な QoS の注意事項

一般的な QoS の注意事項を次に示します。

- QoS を設定できるのは物理ポートだけです。VLAN レベルおよび EtherChannel ポートでは QoS はサポートされていません。
- スイッチで受信された制御トラフィック（スパニングツリー ブリッジ プロトコル データ ユニット（BPDU）やルーティング アップデート パケットなど）には、入力 QoS 処理がすべて行われます。
- キュー設定を変更すると、データが失われることがあります。したがって、トラフィックが最小のときに設定を変更するようにしてください。

QoS の制約事項

以下は、QoS の制約事項を示しています。

- 入力キューイングおよび集約ポリシーはサポートされていません。
- 32 個のクラス マップは、ASIC ごとにサポートされます。
- ポート間でのポリシーマップ（ポリサー/マーキング）の TCAM（Ternary Content Addressable Memory）共有はサポートされていません。このため、自動 QoS/QoS ポリシーマップを適用できるインターフェイスの数は制限されています。

- セキュリティ アクセスコントロールリスト（ACL）とポリシーマップで使用される ACL の両方に同じ TCAM リージョンを使用する必要があります。
- **match ip dscp** コマンドを使用するポリシーマップは、IPv4 アドレスと IPv6 アドレスの両方に一致するため、スケール数は16クラスマップに制限されます。IPv4用に1つのTCAMエントリが作成され、IPv6 用にも 1 つの TCAM エントリが作成されます。
- ASIC ごとに、8 つの TCP ポート比較演算子と 8 つの UDP ポート比較演算子がサポートされ、各 gt（より大きい）、lt（より小さい）、neq（等しくない）演算子は1つのポート比較演算子を使用し、各範囲演算子は2つのポート比較演算子を使用します。この組み合わせを使用するポリシーマップは、TCAMスケールおよびポリシーマップをアタッチできるインターフェイスの数に影響します。
- 次の制限事項が、IPv4 ACL ネットワーク インターフェイスに適用されます。
  - インターフェイスへのアクセスを制御する場合、名前付き ACL または番号付き ACL を使用できます。
  - レイヤ 3 インターフェイスに ACL が適用され、スイッチ上でルーティングがイネーブルになっていない場合は、SNMP、Telnet、Web トラフィックなど、CPU で処理されるパケットだけがフィルタ処理されます。
  - レイヤ 2 インターフェイスに ACL を適用する場合、ルーティングをイネーブルにする必要はありません。
- deny ACL は QoS ポリシーマップではサポートされません。
- ポリシング アクションの送信はサポートされていません。超過アクション ドロップのみサポートされます。
- QoS を設定できるのは物理ポートのみです。VLAN-based QoS はサポートされません。分類、キューイングおよびスケジューリングのような QoS が設定できます。また、ポートにポリシー マップも適用できます。物理ポートに QoS を設定した場合は、非階層型のポリシー マップをポートに適用します。

QoS の概要

QoS の実装

ネットワークは通常、ベストエフォート型の配信方式で動作します。したがって、すべてのトラフィックに等しいプライオリティが与えられ、適度なタイミングで配信される可能性はどのトラフィックでも同等です。輻輳が発生すると、すべてのトラフィックが等しくドロップされます。

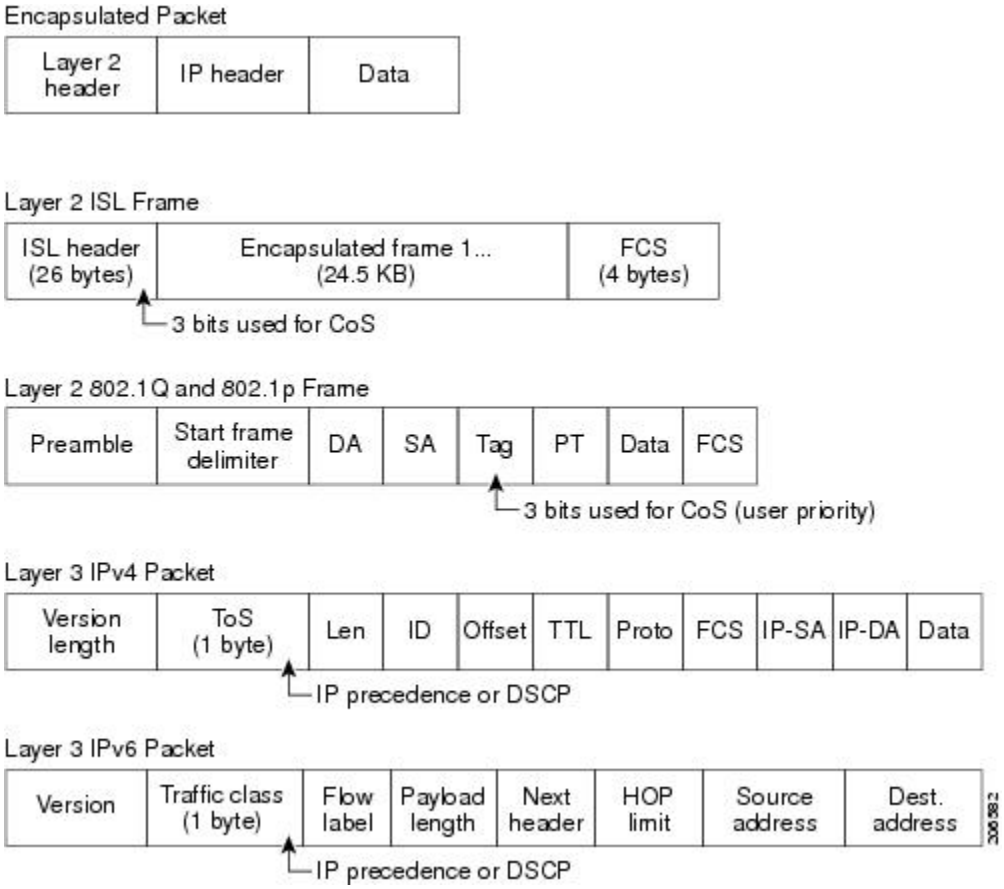
QoS 機能を設定すると、特定のネットワーク トラフィックを選択し、相対的な重要性に応じてそのトラフィックに優先度を指定し、輻輳管理および輻輳回避技術を使用して、優先処理を

実行できます。ネットワークに QoS を実装すると、ネットワーク パフォーマンスがさらに予測しやすくなり、帯域幅をより効率的に利用できるようになります。

QoS は、インターネット技術特別調査委員会（IETF）の規格である Differentiated Services（Diff-Serv）アーキテクチャに基づいて実装されます。このアーキテクチャでは、ネットワークに入るときに各パケットを分類することが規定されています。

図 *1:* フレームおよびパケットにおける *QoS* 分類レイヤ

次の図にレイヤ 2 フレームまたはレイヤ 3 パケットの特殊ビットを示します。



レイヤ 2 フレームのプライオリティ ビット

レイヤ 2 の ISL（スイッチ間リンク）フレーム ヘッダーには、下位 3 ビットで IEEE 802.1p サービス クラス（CoS）値を伝達する 1 バイトのユーザ フィールドがあります。レイヤ 2 ISL トランクとして設定されたポートでは、すべてのトラフィックが ISL フレームに収められます。

レイヤ 2 802.1Q フレーム ヘッダーには、2 バイトのタグ制御情報フィールドがあり、上位 3 ビット（ユーザ プライオリティ ビット）で CoS 値が伝達されます。レイヤ 2 802.1Q トランクとして設定されたポートでは、ネイティブ Virtual LAN（VLAN）のトラフィックを除くすべてのトラフィックが 802.1Q フレームに収められます。

他のフレーム タイプでレイヤ 2 CoS 値を伝達することはできません。

レイヤ 2 CoS 値の範囲は、0（ロー プライオリティ）～ 7（ハイ プライオリティ）です。

レイヤ 3 パケットのプライオリティ ビット

レイヤ 3 IP パケットは、0 ～ 63 の範囲を伝送できます。

分類を使用したエンドツーエンドの QoS ソリューション

インターネットにアクセスするすべてのスイッチおよびルータはクラス情報に基づいて、同じクラス情報が与えられているパケットは同じ扱いで転送を処理し、異なるクラス情報のパケットはそれぞれ異なる扱いをします。パケットのクラス情報は、設定されているポリシー、パケットの詳細な検証、またはその両方に基づいて、エンドホストが割り当てるか、または伝送中にスイッチまたはルータで割り当てることができます。パケットの詳細な検証は、コア スイッチおよびルータの負荷が重くならないように、ネットワークのエッジ付近で行います。

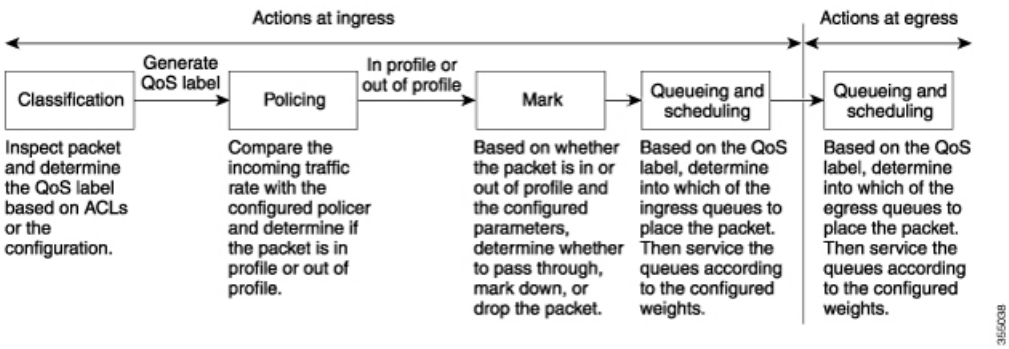
パス上のスイッチおよびルータは、クラス情報を使用して、個々のトラフィック クラスに割り当てるリソースの量を制限できます。Diff-Serv アーキテクチャでトラフィックを処理するときの、各デバイスの動作をホップ単位動作といいます。パス上のすべてのデバイスに一貫性のあるホップ単位動作をさせることによって、エンドツーエンドの QoS ソリューションを構築できます。

ネットワーク上で QoS を実装する作業は、インターネットワーキング デバイスが提供する QoS 機能、ネットワークのトラフィック タイプおよびパターン、さらには着信および発信トラフィックに求める制御のきめ細かさによって、簡単にも複雑にもなります。

QoS 基本モデル

QoS を実装するには、スイッチ上でパケットまたはフローを相互に区別し（分類）、パケットがスイッチを通過するときに所定の QoS を表すラベルを割り当て、設定されたリソース使用率制限にパケットを適合させ（ポリシングおよびマーキング）、リソース競合が発生する状況に応じて異なる処理（キューイングおよびスケジューリング）を行う必要があります。また、スイッチから送信されたトラフィックが特定のトラフィック プロファイルを満たすようにする必要もあります（シェーピング）。

図 *2 : QoS* 基本有線モデル



入力ポートでのアクション

入力ポートでのアクションには、トラフィックの分類、ポリシング、およびマーキングが含まれます。

- パケットと QoS ラベルを関連付けて、パケットごとに異なるパスを分類します。スイッチはパケット内の CoS または DSCP を QoS ラベルにマッピングして、トラフィックの種類を区別します。生成された QoS ラベルは、このパケットでこれ以降に実行されるすべての QoS アクションを識別します。
- ポリシングでは、着信トラフィックのレートを設定済みポリサーと比較して、パケットが適合か不適合かを判別します。ポリサーは、トラフィック フローで消費される帯域幅を制限します。その判別結果がマーカーに渡されます。
- マーキングでは、パケットが不適合の場合の対処法に関して、ポリサーおよび設定情報を検討し、パケットの扱い（パケットを変更しないで通過させるか、パケットの QoS ラベルをマークダウンするか、またはパケットをドロップするか）を決定します。

（注） キューイングおよびスケジューリングは、スイッチの出力でのみサポートされ、入力ではサポートされません。

出力ポートでのアクション

出力ポートでのアクションには、キューイングおよびスケジューリングがあります。

- 4 つの出力キューのどれを使用するかを選択する前に、キューイングでは、QoS パケット ラベルおよび対応する CoS 値を評価します。複数の入力ポートが 1 つの出力ポートに同時にデータを送信すると輻輳が発生することがあるため、WTD を使用してトラフィック クラスを区別し、QoS ラベルに基づいてパケットに別々のしきい値を適用します。しきい値を超過している場合、パケットはドロップされます。
- スケジューリングでは、設定されている SRR の共有重みまたはシェーピング重みに基づいて、4 つの出力キューを処理します。キューの 1 つ（キュー 1）は、他のキューの処理前に空になるまで処理される優先度キューにできます。

分類の概要

分類とは、パケットのフィールドを検証して、トラフィックの種類を区別するプロセスです。QoS がスイッチ上でグローバルにイネーブルになっている場合のみ、分類はイネーブルです。デフォルトでは、QoS はグローバルにディセーブルになっているため、分類は実行されません。

分類中に、スイッチは検索処理を実行し、パケットに QoS ラベルを割り当てます。QoS ラベルは、パケットに対して実行するすべての QoS アクション、およびパケットの送信元キューを識別します。

QoS ラベルは、パケット内の DSCP または CoS 値に基づいて、パケットに実行されるキューイングおよびスケジューリング アクションを決定します。QoS ラベルは信頼設定およびパケット タイプに従ってマッピングされます（分類フローチャートを参照）。

着信トラフィックの分類に、フレームまたはパケットのどのフィールドを使用するかは、ユーザ側で指定します。

Non-IP のトラフィック分類

次の表は、QoS 設定の非 IP トラフィックの分類オプションを示しています。

表 *5:* 非 *IP* トラフィックの分類

Non-IP のトラフィック分類	説明
CoS 値の信頼	着信フレーム内の CoS 値を信頼し（CoS を信頼するようにポートを設定）、設定可能な CoS/DSCP マップを使用してパケットの DSCP 値を生成します。 レイヤ 2 の ISL フレーム ヘッダーは、1 バイトのユーザ フィールドの下位 3 ビットで CoS 値を伝達します。 レイヤ 2 802.1Q フレームのヘッダーは、タグ制御情報フィールドの上位 3 ビットで CoS 値を伝達します。CoS 値の範囲は、0（ロー プライオリティ）～7（ハイ プライオリティ）です。
DSCP 値の信頼	着信フレームの DSCP 値を信頼します。これらの設定は、非 IP トラフィックの場合は無意味です。これらのいずれかの方法で設定されているポートに非 IP トラフィックが着信した場合は、CoS 値が割り当てられ、CoS/DSCP マップから内部 DSCP 値が生成されます。スイッチは内部 DSCP 値を使用して、トラフィックのプライオリティを表示する CoS 値を生成します。
設定されたレイヤ 2 の MAC ACL に基づいた分類	設定されたレイヤ 2 の MAC アクセス コントロール リスト（ACL）に基づいて分類を実行します。レイヤ 2 の MAC ACL は、MAC 送信元アドレス、MAC 宛先アドレス、およびその他のフィールドを調べることができます。ACL が設定されていない場合、パケットには DSCP および CoS 値として 0 が割り当てられ、トラフィックがベストエフォート型であることを意味します。ACL が設定されている場合は、ポリシーマップ アクションによって、着信フレームに割り当てられる DSCP または CoS 値が指定されます。

分類されたパケットは、ポリシングおよびマーキングの各段階に送られます。

IP のトラフィック分類

次の表は、QoS 設定の IP トラフィック分類オプションを示します。

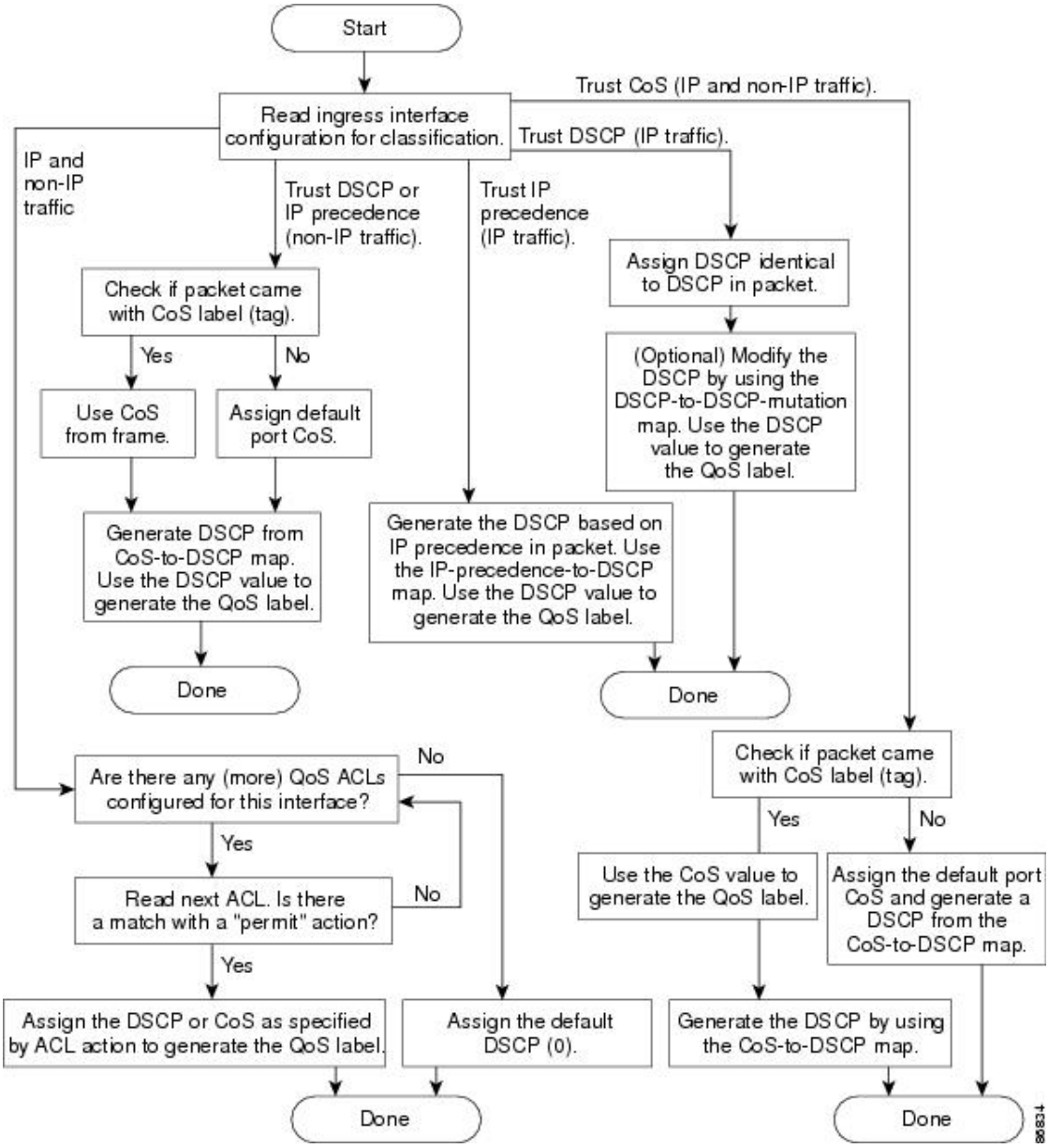
表 *6 : IP* のトラフィック分類

IP のトラフィック分類	説明
DSCP 値の信頼	着信パケットの DSCP 値を信頼し（DSCP を信頼するようにポートを設定し）、同じ DSCP 値をパケットに割り当てます。IETF は、1 バイトの ToS フィールドの上位 6 ビットを DSCP として定義しています。特定の DSCP 値が表すプライオリティは、設定可能です。DSCP 値の範囲は 0 ～ 63 です。 また IPv6 DSCP に基づいて IP トラフィックを分類することもできます。 2 つの QoS 管理ドメインの境界上にあるポートの場合は、設定可能な DSCP/DSCP 変換マップを使用して、DSCP を別の値に変更できます。
CoS 値の信頼	着信パケットに CoS 値がある場合には、その CoS 値を信頼し、CoS/DSCP マップを使用してパケットの DSCP 値を生成します。CoS 値が存在しない場合は、デフォルトのポート CoS 値を使用します。
IP 標準または拡張 ACL	設定された IP 標準 ACL または IP 拡張 ACL（IP ヘッダーの各フィールドを調べる）に基づいて、分類を実行します。ACL が設定されていない場合、パケットには DSCP および CoS 値として 0 が割り当てられ、トラフィックがベストエフォート型であることを意味します。ACL が設定されている場合は、ポリシーマップ アクションによって、着信フレームに割り当てられる DSCP または CoS 値が指定されます。
設定された CoS の上書き	着信パケットに設定された CoS を上書きし、デフォルトのポート CoS 値を適用します。IPv6 パケットの場合、DSCP 値は CoS/DSCP マップとポートのデフォルトの CoS を使用して書き換えられます。これは、IPv4 と IPv6 の両方のトラフィックに対して実行できます。

分類されたパケットは、ポリシングおよびマーキングの各段階に送られます。

分類フローチャート

図 *3 :* 分類フローチャート



アクセス コントロール リスト

IP 標準 ACL、IP 拡張 ACL、またはレイヤ 2 MAC ACL を使用すると、同じ特性を備えたパケット グループ（クラス）を定義できます。また IPv6 ACL に基づいて IP トラフィックを分類することもできます。

QoS のコンテキストでは、アクセス コントロール エントリ（ACE）の許可および拒否アクションの意味が、セキュリティ ACL の場合とは異なります。

- 許可アクションとの一致が検出されると（最初の一致の原則）、指定のQoS関連アクションが実行されます。
- 拒否アクションと一致した場合は、処理中の ACL がスキップされ、次の ACL が処理されます。

（注） 拒否アクションは Cisco IOS リリース 3.7.4E 以降のリリースでサポートされます。

- 許可アクションとの一致が検出されないまま、すべての ACE の検証が終了した場合、そのパケットでは QoS 処理は実行されず、デバイスがベストエフォート型サービスを実行します。
- ポートに複数の ACL が設定されている場合に、許可アクションを含む最初の ACL とパケットの一致が見つかると、それ以降の検索処理は中止され、QoS処理が開始されます。

（注） アクセス リストを作成するときは、アクセス リストの末尾に暗黙の拒否ステートメントがデフォルトで存在し、それ以前のステートメントで一致が見つからなかったすべてのパケットに適用されることに注意してください。

ACL でトラフィック クラスを定義した後で、そのトラフィック クラスにポリシーを結合できます。ポリシーにはそれぞれにアクションを指定した複数のクラスを含めることができます。ポリシーには、特定の集約としてクラスを分類する（DSCP を割り当てるなど）コマンドまたはクラスのレート制限を実施するコマンドを含めることができます。このポリシーを特定のポートに結合すると、そのポートでポリシーが有効になります。

IP ACL を実装して IP トラフィックを分類する場合は、**access-list** グローバル コンフィギュレーション コマンドを使用します。レイヤ 2 MAC ACL を実装して非 IP トラフィックを分類する場合は、**mac access-list extended** グローバル コンフィギュレーション コマンドを使用します。

クラス マップおよびポリシー マップに基づく分類

クラス マップは、特定のトラフィック フロー（またはクラス）に名前を付けて、他のすべてのトラフィックと区別するためのメカニズムです。クラスマップでは、さらに細かく分類するために、特定のトラフィック フローと照合する条件を定義します。この条件には、ACL で定義されたアクセス グループとの照合、またはDSCP値の特定のリストとの照合を含めることができます。複数のトラフィック タイプを分類する場合は、別のクラス マップを作成し、異なる名前を使用できます。パケットをクラス マップ条件と照合した後で、ポリシー マップを使用してさらに分類します。

ポリシー マップでは、作用対象のトラフィック クラスを指定します。アクションとしては、トラフィック クラスの CoS 値または DSCP 値を信頼すること、トラフィック クラスの特定の DSCP 値の設定、またはトラフィックの帯域幅制限の指定およびトラフィックがアウト オブ プ

ロファイルであるときのアクションを含めることができます。ポリシーマップを効率的に機能させるには、ポートにポリシー マップを結合する必要があります。

クラスマップを作成するには、**class-map** グローバル コンフィギュレーション コマンドまたは **class** ポリシー マップ コンフィギュレーション コマンドを使用します。多数のポート間でマップを共有する場合には、**class-map** コマンドを使用する必要があります。**class-map** コマンドを入力すると、クラス マップ コンフィギュレーション モードが開始されます。このモードで、**match** クラス マップ コンフィギュレーション コマンドを使用して、トラフィックの一致条件を定義します。

class class-default ポリシーマップ 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 クラスを設定できます。分類されていないトラフィック（ポリシーマップで設定された他のトラフィック クラスで指定されているトラフィック）は、デフォルト トラフィックとして処理されます。

ポリシーマップは、**policy-map** グローバル コンフィギュレーション コマンドを使用して作成し、名前を付けます。このコマンドを入力すると、ポリシー マップ コンフィギュレーション モードが開始されます。このモードでは、**class** または **set** ポリシーマップ コンフィギュレーション コマンドおよびポリシーマップ クラス コンフィギュレーション コマンドを使用して、特定のトラフィッククラスに対して実行するアクションを指定します。

ポリシーマップには、ポリサー、トラフィックの帯域幅制限、および制限を超えた場合のアクションを定義する **police** ポリシーマップ クラス コンフィギュレーション コマンドを含めることもできます。

ポリシーマップを有効にするには、 **service-policy** インターフェイス コンフィギュレーション コマンドを使用してポートにマップを結合します。

ポリシングおよびマーキングの概要

パケットを分類し、DSCP または CoS に基づいて QoS ラベルを割り当てたあとで、ポリシングおよびマーキング プロセスを開始できます。

ポリシングには、トラフィックの帯域幅限度を指定するポリサーの作成が伴います。制限を超えるパケットは、「アウト オブ プロファイル」または「不適合」になります。各ポリサーはパケットごとに、パケットが適合か不適合かを判別し、パケットに対するアクションを指定します。これらのアクションはマーカーによって実行されます。パケットを変更しないで通過させるアクション、パケットをドロップするアクション、またはパケットに割り当てられた DSCP 値を変更（マークダウン）してパケットの通過を許可するアクションなどがあります。設定可能なポリシング済み DSCP マップを使用すると、パケットに新しい DSCP ベース QoS ラベルが設定されます。マークダウンされたパケットは、元の QoS ラベルと同じキューを使用して、フロー内のパケットの順番が崩れないようにします。

（注） すべてのトラフィックは、ブリッジングされるかルーティングされるかに関係なく、ポリサーの影響を受けます（ポリサーが設定されている場合）。その結果、ブリッジングされたパケットは、ポリシングまたはマーキングが行われたときにドロップされたり、DSCP または CoS フィールドが変更されたりすることがあります。

ポリシングは物理ポートに対して設定できます。ポリシーマップおよびポリシングアクションを設定した後で、**service-policy** インターフェイス コンフィギュレーション コマンドを使用して、ポリシーをポートに接続します。

物理ポートのポリシング

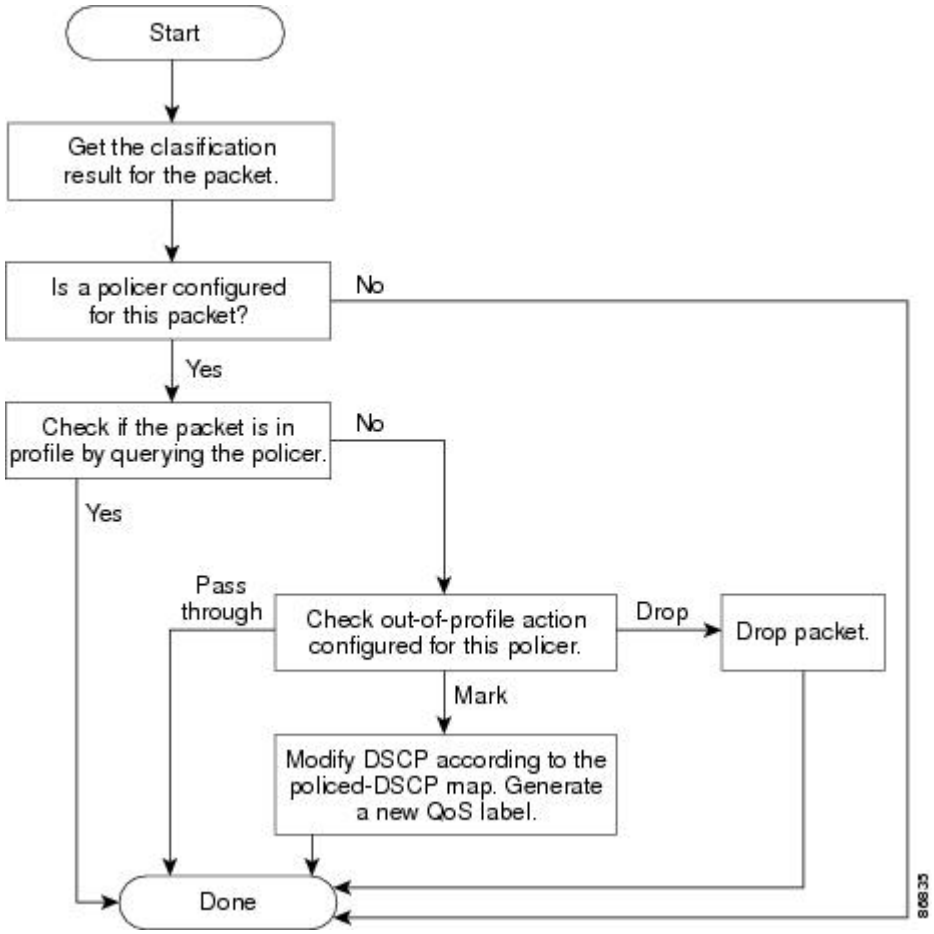
物理ポートのポリシー マップでは、個別のポリサーを作成できます。QoS では、一致する各トラフィック クラスに、ポリサー内で指定された帯域幅制限が個別に適用されます。このタイプのポリサーは、**police** ポリシーマップ クラス コンフィギュレーション コマンドを使用して、ポリシーマップ内で設定します。

ポリシングはトークンバケット アルゴリズムを使用します。各フレームがスイッチに着信すると、バケットにトークンが追加されます。バケットにはホールがあり、平均トラフィック レートとして指定されたレート（ビット/秒）で送信されます。バケットにトークンが追加されるたびに、スイッチは、バケット内に十分なスペースがあるかを確認します。十分なスペースがなければ、パケットは不適合とマーキングされ、指定されたポリサーアクション（ドロップまたはマークダウン）が実行されます。

バケットが満たされる速度は、バケット深度（burst-byte）、トークンが削除されるレート（rate-bps）、および平均レートを上回るバースト期間によって決まります。バケットのサイズによってバースト長に上限が設定され、バックツーバックで送信できるフレーム数が制限されます。バースト期間が短い場合、バケットはオーバーフローせず、トラフィック フローに何のアクションも実行されません。ただし、バースト期間が長く、レートが高い場合、バケットはオーバーフローし、そのバーストのフレームに対してポリシング アクションが実行されます。

バケットの深さ（バケットがオーバーフローするまでの許容最大バースト）を設定するには、**police** 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 burst-byte オプションを使用します。トークンがバケットから削除される速度（平均レート）を設定するには、**police** 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 *rate-bps* オプションを使用します。

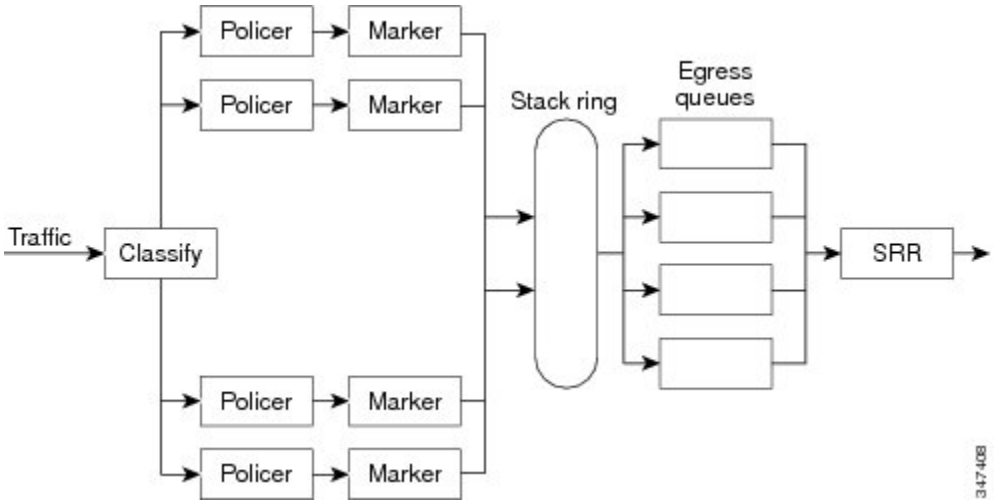
図 *4 :* 物理ポートのポリシングおよびマーキング フローチャート



キューイングおよびスケジューリングの概要

スイッチは、輻輳を防ぐために特定の場所にキューがあります。

図 *5:* スイッチの出力キューの位置



（注） スイッチはデフォルトで 4 つの出力キューをサポートしますが、合計 8 つの出力キューを有効にするオプションがあります。8 出力キューの設定はスタンドアロン スイッチでのみサポートされます。

重み付けテール ドロップ

出力キューは、重み付けテール ドロップ（WTD）と呼ばれるテール ドロップ輻輳回避メカニズムの拡張バージョンを使用します。WTD はキュー長を管理したり、トラフィック分類ごとにドロップ優先順位を設定したりするために実装されています。

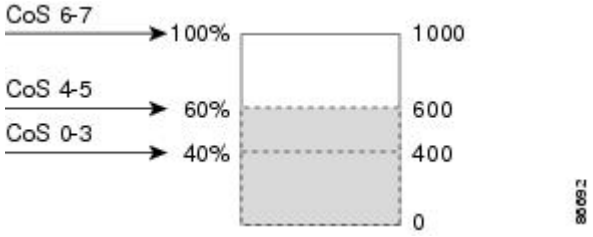
フレームが特定のキューにキューイングされると、WTD はフレームに割り当てられた QoS ラベルを使用して、それぞれ異なるしきい値を適用します。この QoS ラベルのしきい値を超えると（宛先キューの空きスペースがフレーム サイズより小さくなると）、フレームはドロップされます。

各キューには 3 つのしきい値があります。QoS ラベルは、3 つのしきい値のうちのどれがフレームの影響を受けるかを決定します。3 つのしきい値のうち、2 つは設定可能（明示的）で、1 つは設定不可能（暗示的）です。

図 *6: WTD* およびキューの動作

次の図は、サイズが 1000 フレームであるキューでの WTD の動作の例を示しています。ドロップ割合は次のように設定されています。40%（400 フレーム）、60%（600 フレーム）、および 100%（1000 フレーム）です。これらのパーセンテージは、40% しきい値の場合は最大 400 フレーム、60% しきい値の場合は最大 600 フレーム、100% しきい値の場合は最大 1000 フレーム

をキューイングできるという意味です。



この例では、CoS 値 6 および 7 は他の CoS 値よりも重要度が高く、100% ドロップしきい値に割り当てられます（キューフル ステート）。CoS 値 4 および 5 は 60% しきい値に、CoS 値 0 ～ 3 は 40% しきい値に割り当てられます。

600 個のフレームが格納されているキューに、新しいフレームが着信したとします。このフレームの CoS 値は 4 および 5 で、60% のしきい値が適用されます。このフレームがキューに追加されると、しきい値を超過するため、フレームは廃棄されます。

SRR のシェーピングおよび共有

出力キューでは、SRR を共有またはシェーピング用に設定できます。

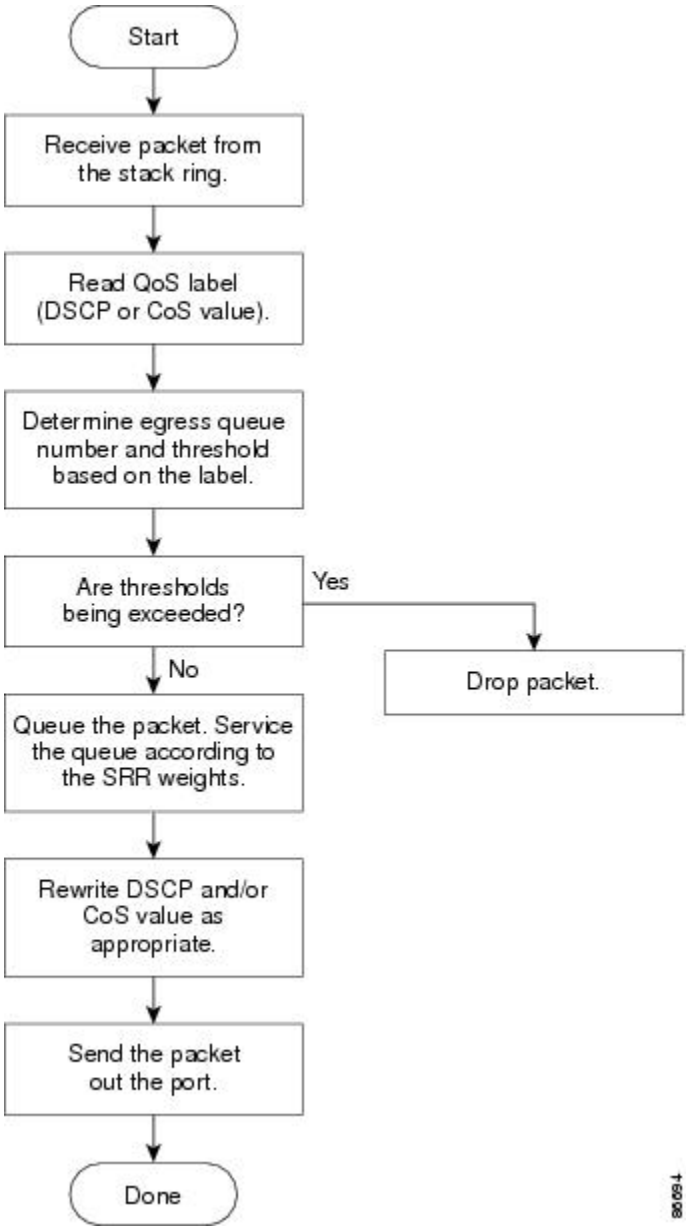
シェーピング モードでは、出力キューの帯域幅割合が保証され、この値にレートが制限されます。リンクがアイドルの場合でも、シェーピングされたトラフィックは割り当てられた帯域幅を超えて使用できません。シェーピングを使用すると、時間あたりのトラフィック フローがより均一になり、バースト トラフィックの最高時と最低時を削減します。シェーピングの場合は、各重みの絶対値を使用して、キューに使用可能な帯域幅が計算されます。

共有モードでは、設定された重みによりキュー間で帯域幅が共有されます。このレベルでは帯域幅は保証されていますが、このレベルに限定されていません。たとえば、特定のキューが空であり、リンクを共有する必要がない場合、残りのキューは未使用の帯域幅を使用して、共有できます。共有の場合、キューからパケットを取り出す頻度は重みの比率によって制御されます。重みの絶対値は関係ありません。シェーピングおよび共有は、インターフェイスごとに設定されます。各インターフェイスは、一意に設定できます。

出力キューでのキューイングおよびスケジューリング

次の図は、スイッチの出力ポートのキューイングおよびスケジューリングのフローチャートを示しています。

図 7: スイッチの出力ポートのキューイングおよびスケジューリング フロー チャート



（注） 緊急キューがイネーブルの場合、SRRによって空になるまで処理されてから、他の3つのキューが処理されます。

出力緊急キュー

各ポートは、そのうち1つ（キュー1）を出力緊急キューにできる、4つの出力キューをサポートしています。これらのキューはキューセットに割り当てられます。スイッチに存在するすべ

てのトラフィックは、パケットに割り当てられた QoS ラベルに基づいて、これらの 4 つのキューのいずれかを通過し、しきい値の影響を受けます。

（注） 緊急キューがイネーブルの場合、SRR によって空になるまで処理されてから、他の 3 つのキューが処理されます。

キューおよび WTD しきい値

スイッチを通過する各パケットをキューおよびしきい値に割り当てることができます。

特に、出力キューには DSCP または CoS 値、しきい値 ID には DSCP または CoS 値をそれぞれマッピングします。**mls qos srr-queue output dscp-map queue** *queue-id* {*dscp1...dscp8* | **threshold** *threshold-id dscp1...dscp8*} または **mls qos srr-queue output cos-map queue** *queue-id* {*cos1...cos8* | **threshold** *threshold-id cos1...cos8*} グローバル コンフィギュレーション コマンドを使用します。DSCP 出力キューしきい値マップおよび CoS 出力キューしきい値マップを表示するには、**show mls qos maps** 特権 EXEC コマンドを使用します。

キューは WTD を使用して、トラフィック クラスごとに異なるドロップ割合をサポートします。各キューには 3 つのドロップしきい値があります。そのうちの 2 つは設定可能（明示的）な WTD しきい値で、もう 1 つはキューフル ステートに設定済みの設定不可能（暗示的）なしきい値です。しきい値 ID 1 および ID 2 用の 2 つの WTD しきい値割合を割り当てます。しきい値 ID 3 のドロップしきい値は、キューフル ステートに設定済みで、変更できません。

シェーピング モードまたは共有モード

共有重みまたはシェーピング重みをポートに割り当てるには、**srr-queue bandwidth share** *weight1 weight2 weight3 weight4* または **srr-queue bandwidth shape** *weight1 weight2 weight3 weight4* インターフェイス コンフィギュレーション コマンドを使用します。

バッファ割り当てと SRR 重み比率を組み合わせることにより、パケットがドロップされる前にバッファに格納して送信できるデータ量が制御されます。重みの比率は、SRR スケジューラが各キューからパケットを送信する頻度の比率です。

緊急キューがイネーブルでない限り、4 つのキューはすべて SRR に参加し、この場合、1 番めの帯域幅重みは無視されて比率計算に使用されません。緊急キューはプライオリティキューであり、他のキューのサービスが提供される前に空になるまでサービスを提供します。緊急キューを有効にするには、**priority-queue out** インターフェイス コンフィギュレーション コマンドを使用します。

（注） 出力キューのデフォルト設定は、ほとんどの状況に適しています。出力キューについて十分理解したうえで、この設定がユーザの QoS ソリューションを満たさないと判断した場合に限り、設定を変更してください。

パケットの変更

QoSを設定するには、パケットの分類、ポリシング、およびキューイングを行います。QoSを提供するプロセス中に次のパケットの変更が発生することがあります。

- IP パケットおよび非 IP パケットの分類では、受信パケットの DSCP または CoS に基づいて、パケットに QoS ラベルが割り当てられます。ただし、この段階ではパケットは変更されません。割り当てられたDSCPまたはCoS値の指定のみがパケットとともに伝達されます。
- ポリシング中は、IP および非 IP パケットに別の DSCP を割り当てることができます（これらのパケットが不適合で、ポリサーがマークダウンDSCPを指定している場合）。この場合も、パケット内のDSCPは変更されず、マークダウン値の指定がパケットとともに伝達されます。IPパケットの場合は、この後の段階でパケットが変更されます。非IPパケットの場合は、DSCP が CoS に変換され、キューイングおよびスケジューリングの決定に使用されます。
- フレームに割り当てられた QoS ラベル、および選択された変換マップに応じて、フレームの DSCP および CoS 値が書き換えられます。テーブル マップを設定しない場合、および着信フレームの DSCP を信頼するようにポートが設定されている場合、フレーム内の DSCP 値は変更されませんが、CoS は、DSCP/CoS マップに基づいて書き換えられます。着信フレームのCoSを信頼するようにポートが設定されていて、着信フレームがIPパケットの場合、フレーム内の CoS 値は変更されないで、CoS/DSCP マップに従って DSCP が変更されることがあります。

  入力変換が行われると、選択された新しいDSCP値に応じてDSCPが書き換えられます。ポリシー マップの設定アクションによっても、DSCP が書き換えられます。

標準 QoS のデフォルト設定

QoS はデフォルトではディセーブルになっています。

QoS が無効の場合は、パケットが変更されないため、信頼できるポートまたは信頼できないポートといった概念はありません。パケット内の CoS、DSCP 値は変更されません。

トラフィックはPass-Throughモードでスイッチングされます。パケットは書き換えられることなくスイッチングされ、ポリシングなしのベスト エフォートに分類されます。

mls qos グローバル コンフィギュレーション コマンドを使用してQoSを有効にし、その他のすべての QoS 設定がデフォルトである場合、トラフィックはポリシングを伴わないベストエフォート型として分類されます（DSCPおよびCoS値は0に設定されます）。ポリシー マップは設定されません。すべてのポート上のデフォルトポートの信頼性は、信頼性なし（untrusted）の状態です。

出力キューのデフォルト設定

次の表は、出力キューのデフォルト設定について説明しています。

次の表は、QoS がイネーブルの場合の各キュー セットに対するデフォルトの出力キューを示しています。すべてのポートはキューセット 1 にマッピングされます。ポートの帯域幅限度は 100% に設定され、レートは制限されません。SRR シェーピング重み（絶対）機能では、ゼロのシェーピング重みはキューが共有モードで動作していることを示しています。SRR 共有重み機能では、帯域幅の 4 分の 1 が各キューに割り当てられます。

表 *7:* 出力キューのデフォルト設定

機能	キュー 1	キュー 2	キュー 3	キュー 4
バッファ割り当て	25%	25%	25%	25%
WTD ドロップしきい値 1	80 %	80 %	80 %	80 %
WTD ドロップしきい値 2	80 %	80 %	80 %	80 %
予約済みしきい値	1000%	1000%	1000%	1000%
最大しきい値	400%	400%	400%	400%
SRR シェーピング重み（絶対）	25	0	0	0
SRR 共有重み	25	25	25	25

次の表は、QoS がイネーブルの場合のデフォルトの CoS 出力キューしきい値マップを示しています。

表 *8:* デフォルトの *CoS* 出力キューしきい値マップ

CoS 値	キュー ID-しきい値 ID
0、1	2 - 1
2、3	3 - 1
4	4 - 1
5	1 - 1
6、7	4 - 1

マッピング テーブルのデフォルト設定

デフォルトの DSCP/DSCP 変換マップは、着信 DSCP 値を同じ DSCP 値にマッピングするヌルマップです。

デフォルトのポリシング済み DSCP マップは、着信 DSCP 値を同じ DSCP 値にマッピングする（マークダウンしない）空のマップです。

QoS の設定方法

このセクションでは、QoS の設定方法について説明します。

QoS のグローバルなイネーブル化

デフォルトでは、QoS はスイッチ上でディセーブルに設定されています。

QoS をイネーブルにするために次の手順が必要です。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**mls qos** 例： `Device(config)# mls qos`	QoS をグローバルにイネーブルにします。 QoSは、次の関連トピックのセクションで説明されているデフォルト設定で動作します。 （注） QoS をディセーブルにするには、 **no mls qos** グローバル コンフィギュレーション コマンドを使用します。
ステップ 3	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。
ステップ 4	**show mls qos** 例： `Device# show mls qos`	QoS の設定を確認します。
ステップ 5	**copy running-config startup-config** 例：	（任意）コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	`Device# copy running-config startup-config`	

ポートの信頼状態による分類の設定

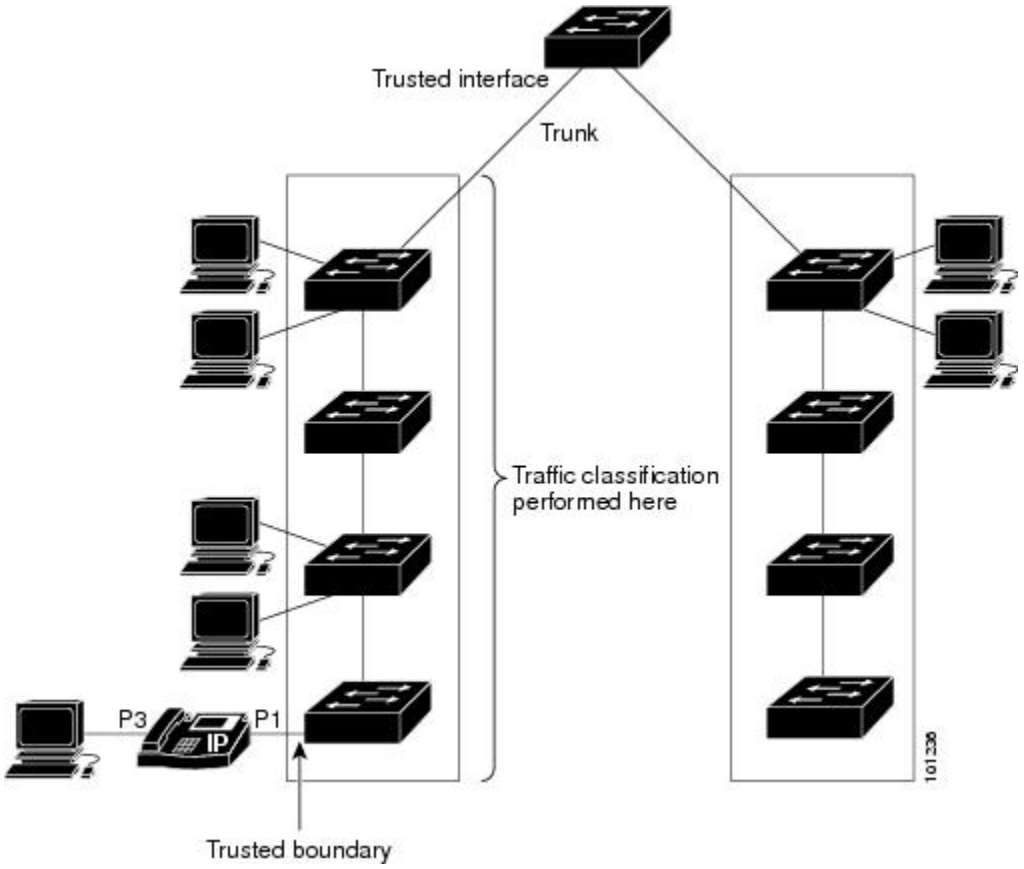
ここでは、ポートの信頼状態を使用して着信トラフィックを分類する方法について説明します。

（注） ネットワーク構成に応じて、このモジュールに示されている作業またはQoS ポリシーの設定（44 ページ）に記載されている作業を 1 つまたは複数実行する必要があります。

QoS ドメイン内のポートの信頼状態の設定

QoS ドメインに入るパケットは、QoS ドメインのエッジで分類されます。パケットがエッジで分類されると、QoS ドメイン内の各スイッチでパケットを分類する必要がないので、QoS ドメイン内のスイッチ ポートをいずれか 1 つの信頼状態に設定できます。

図 *8 : QoS* ドメイン内のポートの信頼状態



手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**interface** *interface-id*
	例：
	`Device(config)# interface gigabitethernet 1/0/1`
	`Or`
	`Device(config)# interface fastethernet 1/0/1`
		信頼するポートを指定し、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。指定できるインターフェイスは、物理ポートです。

ステップ 3	**mls qos trust** [**cos** | **dscp**] 例：	ポートの信頼状態を設定します。

	コマンドまたはアクション	目的
	`Device(config-if)# mls qos trust cos`	デフォルトでは、ポートは trusted ではありません。キーワードを指定しなかった場合、**dscp** がデフォルトになります。 キーワードの意味は次のとおりです。 • **cos**：パケットの CoS 値を使用して入力パケットを分類します。タグのない IP パケットの場合、ポートのデフォルトの CoS 値が使用されます。デフォルトのポート CoS 値は 0 です。 • **dscp**：パケットの DSCP 値を使用して入力パケットを分類します。非 IP パケットでは、パケットがタグ付きの場合、パケットの CoS 値が使用されます。パケットがタグなしの場合は、デフォルトのポート CoS が使用されます。スイッチは、内部で CoS/DSCP マップを使用して CoS 値を DSCP 値にマッピングします。 ポートを trusted 以外のステートに戻すには、**no mls qos trust** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show mls qos interface** 例： `Device# show mls qos interface`	入力を確認します。

インターフェイスの CoS 値の設定

QoS は、trusted ポートおよび untrusted ポートで受信したタグなしフレームに、**mls qos cos** インターフェイス コンフィギュレーション コマンドで指定された CoS 値を割り当てます。

ポートのデフォルト CoS 値を定義する場合、またはポート上のすべての着信パケットにデフォルト CoS 値を割り当てる場合には、特権 EXEC モードから次の手順を実行します。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**interface** *interface-id*
	例：
	`Device(config)# interface gigabitethernet 1/0/2`
	Or
	`Device(config)# interface fastethernet 1/0/2`
		設定するポートを指定し、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。
		有効なインターフェイスには、物理ポートが含まれます。

ステップ 3	**mls qos cos** {*default-cos* | **override**} 例： `Device(config-if)# mls qos cos override`	ポートのデフォルトの CoS 値を設定します。 • *default-cos* には、ポートに割り当てるデフォルトの CoS 値を指定します。パケットがタグなしの場合、デフォルトの CoS 値がパケットの CoS 値になります。指定できる CoS 範囲は 0 ～ 7 です。デフォルトは 0 です。 • 着信パケットにすでに設定されている信頼状態を変更し、すべての着信パケットにデフォルトのポート CoS 値を適用する場合は、**override** キーワードを使用します。デフォルトでは、CoS の上書きはディセーブルに設定されています。 特定のポートに届くすべての着信パケットに、他のポートからのパケットより高い、または低いプライオリティを与える場合には、**override** キーワードを使用します。ポートがすでに DSCP または CoS を信頼するように設定されている場合でも、設定済みの信頼状態がこのコマンドによって上書き変更され、すべての着信 CoS 値に、このコマンドで設定されたデフォルトの CoS 値が割り当てられます。着信パケットがタ

	コマンドまたはアクション	目的
		グ付きの場合、入力ポートで、ポートのデフォルト CoS を使用してパケットの CoS 値が変更されます。 （注） デフォルトの設定に戻す場合は、 **no mls qos cos** {*default-cos* | **override**} インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show mls qos interface** 例： `Device# show mls qos interface`	入力を確認します。

ポート セキュリティを確保するための信頼境界の設定

一般的なネットワークでは、Cisco IP 電話をスイッチポートに接続して、電話の背後からデータパケットを生成するデバイスをカスケードします。Cisco IP Phone では、音声パケット CoS レベルをハイ プライオリティ（CoS = 5）にマーキングし、データ パケットをロー プライオリティ（CoS = 0）にマーキングすることで、共有データ リンクを通して音声品質を保証しています。電話からスイッチに送信されたトラフィックは通常 802.1Q ヘッダーを使用するタグでマーキングされています。ヘッダーには VLAN 情報およびパケットのプライオリティになる CoS の 3 ビット フィールドが含まれています。

ほとんどの Cisco IP Phone 設定では、電話からスイッチへ送信されるトラフィックは、音声トラフィックがネットワーク内の他のタイプのトラフィックに対して適切にプライオリティ付けがされていることを保証するように信頼されています。**mls qos trust cos** インターフェイス コンフィギュレーション コマンドを使用して、ポートで受信されるすべてのトラフィックの CoS ラベルを信頼するように、電話が接続されているスイッチポートを設定します。**mls qos trust dscp** インターフェイス コンフィギュレーション コマンドを使用して、電話が接続されているルーテッドポートを設定し、ポートで受信されるすべてのトラフィックの DSCP ラベルを信頼するように、

信頼設定により、ユーザが電話をバイパスして PC を直接デバイスに接続する場合に、ハイプライオリティキューの誤使用を避けるため信頼境界機能も使用できます。信頼境界機能を使用しないと、（信頼性のある CoS 設定により）PC が生成した CoS ラベルがデバイスで信頼されてしまいます。それに対して、信頼境界機能は CDP を使用してスイッチ ポートにある Cisco

IP Phone（Cisco IP Phone 7910、7935、7940、および 7960）の存在を検出します。電話が検出されない場合、信頼境界機能が高優先順位キューの誤使用を避けるためにスイッチポートの信頼設定をディセーブルにします。信頼境界機能は、PC および Cisco IP Phone がデバイスに接続されているハブに接続されている場合は機能しないことに注意してください。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**cdp run** 例： `Device(config)# cdp run`	CDP をグローバルにイネーブルにします。デフォルトでは、CDP がイネーブルに設定されています。

ステップ 3	**interface** *interface-id*
	例：
	`Device(config)# interface gigabitethernet 1/0/1`
	Or
	`Device(config)# interface fastethernet 1/0/1`
		Cisco IP Phone に接続するポートを指定し、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。
		有効なインターフェイスには、物理ポートが含まれます。

ステップ 4	**cdp enable** 例： `Device(config-if)# cdp enable`	ポート上で CDP をイネーブルにします。デフォルトでは、CDP がイネーブルに設定されています。
ステップ 5	次のいずれかを使用します。 • **mls qos trust cos** • **mls qos trust dscp** 例： `Device(config-if)# mls qos trust cos`	Cisco IP Phone から受信したトラフィックの CoS 値を信頼するようにスイッチポートを設定します。 または Cisco IP Phone から受信したトラフィックの DSCP 値を信頼するようにルーテッド ポートを設定します。 デフォルトでは、ポートは trusted ではありません。
ステップ 6	**mls qos trust device cisco-phone** 例：	Cisco IP Phone が信頼できるデバイスであることを指定します。

	コマンドまたはアクション	目的
	`Device(config-if)# mls qos trust device cisco-phone`	信頼境界機能と自動 QoS（**auto qos voip** インターフェイス コンフィギュレーション コマンド）を同時にイネーブルにはできません。両者は相互に排他的です。 （注） 信頼境界機能をディセーブルにするには、 **no mls qos trust device** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 7	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 8	**show mls qos interface** 例： `Device# show mls qos interface`	入力を確認します。

DSCP トランスペアレント モードの有効化

スイッチは透過的な DSCP 機能をサポートします。この機能は発信パケットの DSCP フィールドのみに作用します。デフォルトでは、DSCP 透過性はディセーブルです。スイッチでは着信パケットの DSCP フィールドが変更され、発信パケットの DSCP フィールドは、ポートの信頼設定、ポリシング、マーキングを含めて Quality of Service（QoS）に基づきます。

no mls qos rewrite ip dscp コマンドを使用して DSCP 透過が有効になっている場合、スイッチは着信パケットの DSCP フィールドは変更せず、送信パケットの DSCP フィールドも着信パケットのものと同じになります。

透過的な DSCP 設定にかかわらず、スイッチはパケット内部の DSCP 値を変更し、トラフィックのプライオリティを提示する CoS 値を生成します。また、スイッチは内部 DSCP 値を使用して、出力キューおよびしきい値を選択します。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**mls qos** 例： `Device(config)# mls qos`	QoS をグローバルにイネーブルにします。
ステップ 3	**no mls qos rewrite ip dscp** 例： `Device(config)# no mls qos rewrite ip dscp`	DSCP 透過性を有効にします。スイッチが IP パケットの DSCP フィールドを変更しないよう設定されます。
ステップ 4	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。

ステップ 5	**show mls qos interface** [*interface-id*]
	例：
	`Device(config)# show mls qos interface gigabitethernet 1/0/1`
	`Or`
	`Device(config)# show mls qos interface fastethernet 1/0/1`
		入力を確認します。

DSCP 透過モード

DSCP 透過を無効にして、トラストの設定または ACL に基づいて DSCP 値を変更するようにスイッチを設定するには、**mls qos rewrite ip dscp** グローバル コンフィギュレーション コマンドを使用します。

no mls qos グローバル コンフィギュレーション コマンドを使用して QoS を無効にする場合、CoS 値と DSCP 値は変更されません（デフォルトの QoS 設定）。

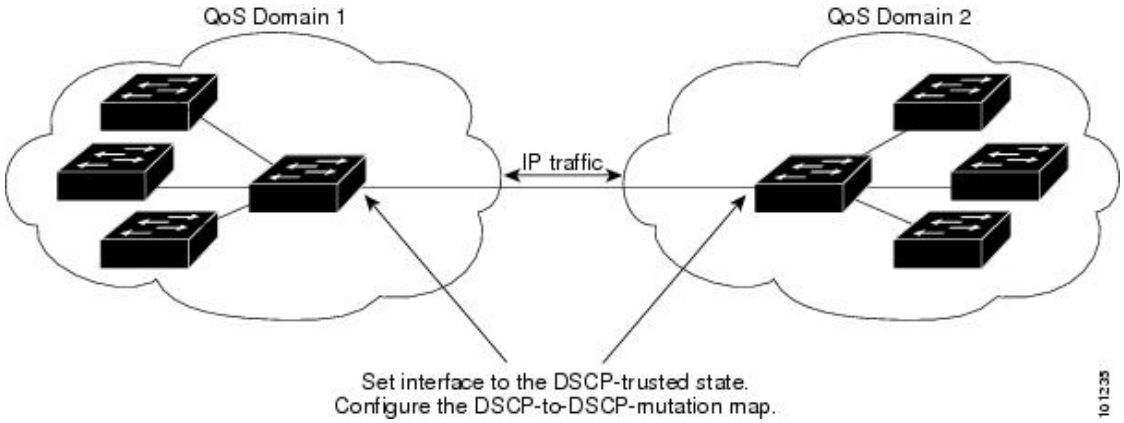
no mls qos rewrite ip dscp グローバル コンフィギュレーション コマンドを入力して DSCP 透過を有効にしてから、**mls qos trust** [**cos** | **dscp**] インターフェイス コンフィギュレーション コマンドを入力する場合、DSCP 透過は引き続き有効となります。

（注） DSCP 透過はデフォルトでは無効になっています。

別の QoS ドメインとの境界ポートでの DSCP 信頼状態の設定

2 つの異なる QoS ドメインを管理しているときに、その QoS ドメイン間の IP トラフィックに QoS 機能を実装する場合は、ドメインの境界に位置するデバイスポートを DSCP trusted ステートに設定できます。受信ポートでは DSCP trusted 値をそのまま使用し、QoS の分類手順が省略されます。2 つのドメインで異なる DSCP 値が使用されている場合は、他のドメイン内での定義に一致するように一連の DSCP 値を変換する DSCP/DSCP 変換マップを設定できます。

図 *9*: 別の *QoS* ドメインとのポート境界での *DSCP* 信頼ステート



ポート上に DSCP trusted ステートを設定して、DSCP/DSCP 変換マップを変更するには、特権 EXEC モードで次の手順を実行します。両方の QoS ドメインに一貫した方法でマッピングするには、両方のドメイン内のポート上で次の手順を実行する必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**mls qos map dscp-mutation** *in-dscp* **to** *out-dscp* 例： `Device(config)# mls qos map dscp-mutation 10 11 12 13 to 30`	DSCP/DSCP 変換マップを変更します。 デフォルトの DSCP/DSCP 変換マップは、着信 DSCP 値を同じ DSCP 値にマッピングするヌル マップです。 • *in-dscp* には、最大 8 つの DSCP 値をスペースで区切って入力します。さらに、**to** キーワードを入力します。 • *out-dscp* には、1 つの DSCP 値を入力します。

	コマンドまたはアクション	目的
		DSCP の範囲は 0 ～ 63 です。

ステップ 3	**interface** *interface-id*
	例：
	`Device(config)# interface gigabitethernet 1/0/2`
	Or
	`Device(config)# interface fastethernet 1/0/2`
		信頼するポートを指定し、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。
		有効なインターフェイスには、物理ポートが含まれます。

ステップ 4	**mls qos trust dscp** 例： `Device(config-if)# mls qos trust dscp`	DSCP trusted ポートとして入力ポートを設定します。デフォルトでは、ポートは trusted ではありません。 （注） ポートを trusted 以外のステートに戻すには、**no mls qos trust** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 5	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show mls qos maps dscp-mutation** 例： `Device# show mls qos maps dscp-mutation`	入力を確認します。 （注） ポートを trusted 以外のステートに戻すには、 **no mls qos trust interface** コンフィギュレーション コマンドを使用します。デフォルトの DSCP/DSCP 変換マップ値に戻すには、 **no mls qos map dscp-mutation** *dscp-mutation-name* グローバル コンフィギュレーション コマンドを使用します。

QoS ポリシーの設定

QoS ポリシーを設定するには、次のタスクが必要です。

- トラフィックのクラスへの分類

- 各トラフィック クラスに適用するポリシーの設定
- ポートへのポリシーの付加

ここでは、トラフィックを分類、ポリシング、マーキングする方法について説明します。ネットワーク設定に応じて、この項のモジュールの 1 つ以上を実行します。

ACL を使用したトラフィックの分類

IPv4 標準 ACLS、IPv4 拡張 ACL または IPv6 ACL を使用して IP トラフィックを分類できます。

非 IP トラフィックの分類はレイヤ 2 MAC ACL でできます。

IPv4 トラフィック用の IP 標準 ACL の作成

始める前に

この作業を実行する前に、QoS 設定のために使用するアクセス リストを決定します。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**access-list** *access-list-number* {**permit**} *source* [*source-wildcard*] 例： `Device(config)# access-list 1 permit 192.2.255.0 1.1.1.255`	IP 標準 ACL を作成し、必要な回数だけコマンドを繰り返します。 • *access-list-number* には、アクセス リスト番号を入力します。有効範囲は 1 ～ 99 および 1300 ～ 1999 です。 • **permit** キーワードを使用すると、条件が一致した場合に特定のトラフィックタイプを許可します。 • *source* には、パケットの送信元となるネットワークまたはホストを指定します。**any** キーワードは 0.0.0.0 255.255.255.255 の省略形として使用できます。 •（任意）*source-wildcard* には、source に適用されるワイルドカードビットをドット付き 10 進表記で入

	コマンドまたはアクション	目的
		力します。無視するビット位置には 1 を設定します。 アクセス リストを作成するときは、アクセスリストの末尾に暗黙の拒否ステートメントがデフォルトで存在し、それ以前のステートメントで一致が見つからなかったすべてのパケットに適用されることに注意してください。 （注） アクセスリストを削除するには、**no access-list** *access-list-number* グローバル コンフィギュレーション コマンドを入力します。
ステップ 3	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。
ステップ 4	**show access-lists** 例： `Device# show access-lists`	入力を確認します。
ステップ 5	**copy running-config startup-config** 例： `Device# copy-running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

IPv4 トラフィック用の IP 拡張 ACL の作成

始める前に

この作業を実行する前に、QoS 設定のために使用するアクセス リストを決定します。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**access-list** *access-list-number* **permit** *protocol source source-wildcard destination destination-wildcard* 例： `Device(config)# access-list 100 permit ip any any dscp 32`	IP 拡張 ACL を作成し、必要な回数だけコマンドを繰り返します。 • *access-list-number* には、アクセス リスト番号を入力します。有効範囲は 100 ～ 199 および 2000 ～ 2699 です。 • **permit** キーワードを使用すると、条件が一致した場合に特定のトラフィックタイプを許可します。 • *protocol* には、IP プロトコルの名前または番号を入力します。疑問符（?）を使用すると、使用できるプロトコル キーワードのリストが表示されます。 • *source* には、パケットの送信元となるネットワークまたはホストを指定します。ネットワークまたはホストを指定するには、ドット付き 10 進表記を使用したり、*source* 0.0.0.0 *source-wildcard* 255.255.255.255 の短縮形として **any** キーワードを使用したり、*source* 0.0.0.0 を表す **host** キーワードを使用します。 • *source-wildcard* では、無視するビット位置に 1 を入力することによって、ワイルドカード ビットを指定します。ワイルドカードを指定するには、ドット付き 10 進表記を使用したり、*source* 0.0.0.0 *source-wildcard* 255.255.255.255 の短縮形として **any** キーワードを使用したり、*source* 0.0.0.0 を表す **host** キーワードを使用します。 • *destination* には、パケットの宛先となるネットワークまたはホストを指定します。*destination* および *destination-wildcard* には、*source* お

	コマンドまたはアクション	目的
		および *source-wildcard* での説明と同じオプションを使用できます。 アクセス リストを作成する際は、アクセス リストの末尾に暗黙の deny ステートメントがデフォルトで存在し、ACL の終わりに到達するまで一致が見つからなかったすべてのパケットに適用されることに注意してください。 （注） アクセスリストを削除するには、**no access-list** *access-list-number* グローバル コンフィギュレーション コマンドを入力します。
ステップ 3	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。
ステップ 4	**show access-lists** 例： `Device# show access-lists`	入力を確認します。
ステップ 5	**copy running-config startup-config** 例： `Device# copy-running-config startup-config`	（任意）コンフィギュレーション ファイルに設定を保存します。

IPv6 トラフィック用の IPv6 ACL の作成

始める前に

この作業を実行する前に、QoS 設定のために使用するアクセス リストを決定します。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**ipv6 access-list** *access-list-name* 例： `Device(config)# ipv6 access-list ipv6_Name_ACL`	IPv6 ACL を定義し、IPv6 アクセス リスト コンフィギュレーション モードを開始します。 アクセス リスト名にはスペースまたは引用符を含めることはできません。また、数字で開始することもできません。 （注） アクセスリストを削除するには、**no ipv6 access-list** *access-list-number* グローバル コンフィギュレーション コマンドを入力します。
ステップ 3	{**permit**} *protocol* {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix/ prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] 例： `Device(config-ipv6-acl)# permit ip host 10::1 host 11::2 host`	条件が一致したら、**permit** を入力してパケットを許可します。次に、条件について説明します。 *protocol* には、インターネットプロトコルの名前または番号を入力します。**ahp**、**esp**、**icmp**、**ipv6**、**pcp**、**stcp**、**tcp**、**udp**、または IPv6 プロトコル番号を表す 0 ～ 255 の整数を使用できます。 • *source-ipv6-prefix/prefix-length* または *destination-ipv6-prefix/ prefix-length* は、許可条件を設定する送信元または宛先 IPv6 ネットワークあるいはネットワーク クラスで、コロン区切りの 16 ビット値を使用した 16 進形式で指定します（RFC 2373 を参照）。 • IPv6 プレフィックス ::/0 の短縮形として、**any** を入力します。 • **host** *source-ipv6-address* または *destination-ipv6-address* には、許可条件を設定する送信元または宛先 IPv6 ホストアドレスを入力します。アドレスはコロン区切りの 16 ビット値を使用した 16 進形式で指定します。

	コマンドまたはアクション	目的
		• （任意）*operator* には、指定のプロトコルの送信元ポートまたは宛先ポートを比較するオペランドを指定します。オペランドには、**lt**（より小さい）、**gt**（より大きい）、**eq**（等しい）、**neq**（等しくない）、および **range**（包含範囲）があります。 *source-ipv6-prefix/prefix-length* 引数のあとの operator は、送信元ポートに一致する必要があります。*destination-ipv6- prefix/prefix-length* 引数のあとの operator は、宛先ポートに一致する必要があります。 • （任意）*port-number* は、0 ～ 65535 の 10 進数または TCP あるいは UDP ポートの名前です。TCP ポート名を使用できるのは、TCP のフィルタリング時だけです。UDP ポート名を使用できるのは、UDP のフィルタリング時だけです。 • （任意）**dscp** *value* を入力して、各 IPv6 パケットヘッダーの Traffic Class フィールド内のトラフィック クラス値と DiffServ コードポイント値を照合します。指定できる範囲は 0 ～ 63 です。
ステップ 4	**end** 例： `Device(config-ipv6-acl)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show ipv6 access-list** 例： `Device# show ipv6 access-list`	アクセス リストの設定を確認します。
ステップ 6	**copy running-config startup-config** 例：	（任意）コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	`Device# copy-running-config startup-config`	

非 IP トラフィック用のレイヤ 2 MAC ACL の作成

始める前に

この作業を実行する前に、レイヤ 2 の MAC アクセス リストが QoS 設定に必要であることを決定します。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**mac access-list extended** *name* 例： `Device(config)# mac access-list extended maclist1`	リストの名前を指定することによって、レイヤ 2 MAC ACL を作成します。 このコマンドを入力すると、拡張 MAC ACL コンフィギュレーション モードに切り替わります。 （注） アクセスリストを削除するには、 **no mac access-list extended** *access-list-name* グローバル コンフィギュレーション コマンドを入力します。

ステップ 3	{**permit** | **deny**} { **host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*]
	例：
	`Device(config-ext-mac1) # permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0`
	`Device(config-ext-mac1) # permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp`
		条件が一致した場合に許可または拒否するトラフィック タイプを指定します。必要な回数だけコマンドを入力します。
		• *src-MAC-addr* には、パケットの送信元となるホストの MAC アドレスを指定します。MAC アドレスを指定するには、16 進表記（H.H.H）を使用したり、*source* 0.0.0、*source-wildcard* ffff.ffff.ffff の短縮形として **any** キーワードを使用したり、*source* 0.0.0 を表す **host** キーワードを使用します。

	コマンドまたはアクション	目的
		• *mask* では、無視するビット位置に 1 を入力することによって、ワイルドカード ビットを指定します。 • *dst-MAC-addr* には、パケットの宛先となるホストの MAC アドレスを指定します。MAC アドレスを指定するには、16 進表記（H.H.H）を使用したり、*source* 0.0.0、*source-wildcard* ffff.ffff.ffff の短縮形として **any** キーワードを使用したり、*source* 0.0.0 を表す **host** キーワードを使用します。 •（任意）*type mask* には、Ethernet II または SNAP でカプセル化されたパケットの Ethertype 番号を指定して、パケットのプロトコルを識別します。*type* の範囲は 0 ～ 65535 です。通常は 16 進数で指定します。*mask* では、一致をテストする前に Ethertype に適用される *don't care* ビットを入力します。 アクセス リストを作成するときは、アクセスリストの末尾に暗黙の拒否ステートメントがデフォルトで存在し、それ以前のステートメントで一致が見つからなかったすべてのパケットに適用されることに注意してください。
ステップ 4	**end** 例： `Device(config-ext-mac1)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show access-lists** [*access-list-number* | *access-list-name*] 例： `Device# show access-lists`	入力を確認します。
ステップ 6	**copy running-config startup-config** 例：	（任意）コンフィギュレーション ファイルに設定を保存します。

	コマンドまたはアクション	目的
	`Device# copy-running-config startup-config`	

クラス マップによるトラフィックの分類

個々のトラフィックフロー（またはクラス）を他のすべてのトラフィックから分離して名前を付けるには、**class-map** グローバル コンフィギュレーション コマンドを使用します。クラス マップでは、さらに細かく分類するために、特定のトラフィック フローと照合する条件を定義します。match ステートメントには、ACL、DSCP 値などの条件を指定できます。一致条件は、クラス マップ コンフィギュレーション モードの中で match ステートメントを 1 つ入力することによって定義します。

（注） **class** ポリシーマップ コンフィギュレーション コマンドを使用することによって、ポリシーマップの作成時にクラスマップを作成することもできます。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	次のいずれかを使用します。 • **access-list** *access-list-number* {**permit**} *source* [*source-wildcard*] • **access-list** *access-list-number* {**permit**} *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] • **ipv6 access-list** *access-list-name* {**permit**} *protocol* {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix/ prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] • **mac access-list extended** *name* {**permit** | **deny**} { **host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*]	必要な回数だけコマンドを繰り返し、IP 標準または IP 拡張 ACL、IP トラフィック用の IPv6 ACL、または非 IP トラフィック用のレイヤ 2 MAC ACL を作成します。 アクセス リストを作成するときは、アクセスリストの末尾に暗黙の拒否ステートメントがデフォルトで存在し、それ以前のステートメントで一致が見つからなかったすべてのパケットに適用されることに注意してください。

	コマンドまたはアクション	目的
	例： `Device(config)# access-list 103 permit ip any any dscp 10`	
ステップ 3	**class-map** [**match-all**] *class-map-name* 例： `Device(config)# class-map class1`	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。 デフォルトでは、クラス マップは定義されていません。 • （任意）このクラスマップ配下のすべての一致ステートメントの論理 AND を実行するには、**match-all** キーワードを使用します。この場合は、クラス マップ内のすべての一致条件と一致する必要があります。 • *class-map-name* には、クラス マップ名を指定します。 （注） 既存のクラスマップを削除するには、**no class-map** [**match-all**] *class-map-name* グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**match** { **access-group** *acl-index-or-name* | **ip dscp** *dscp-list*} 例： `Device(config-cmap)# match ip dscp 10 11 12`	トラフィックを分類するための一致条件を定義します。 デフォルトでは、一致条件は定義されていません。 クラス マップごとにサポートされる一致条件は 1 つだけです。また、クラス マップごとにサポートされる ACL は 1 つだけです。 • **access-group** *acl-index-or-name* には、ステップ 2 で作成した ACL の番号または名前を指定します。 • IPv6 トラフィックを **match access-group** コマンドでフィルタリ

	コマンドまたはアクション	目的
		ングするには、ステップ 2 の手順で IPv6 ACL を作成します。 • **ip dscp** *dscp-list* には、着信パケットと照合する IP DSCP 値を 8 つまで入力します。各値はスペースで区切ります。指定できる範囲は 0 ～ 63 です。 （注） 一致条件を削除するには、**no match** {**access-group** *acl-index-or-name* | **ip dscp**} クラスマップ コンフィギュレーション コマンドを使用します。
ステップ 5	**end** 例： `Device(config-cmap)# end`	特権 EXEC モードに戻ります。
ステップ 6	**show class-map** 例： `Device# show class-map`	入力を確認します。

ポリシーマップによる物理ポートのトラフィックの分類、ポリシング、およびマーキング

作用対象となるトラフィック クラスを指定するポリシー マップを、物理ポート上に設定できます。トラフィッククラスの CoS または DSCP 値を信頼するアクション、トラフィッククラスに特定の DSCP 値を設定するアクション、および一致する各トラフィッククラスにトラフィック帯域幅限度を指定するアクション（ポリサー）や、トラフィックが不適合な場合の対処法を指定するアクション（マーキング）などを指定できます。

ポリシー マップには、次の特性もあります。

- 1 つのポリシーマップに、それぞれ異なる一致条件とポリサーを指定した複数のクラス ステートメントを指定できます。
- ポリシー マップには、事前に定義されたデフォルトのトラフィック クラスを含めることができます。デフォルトのトラフィック クラスはマップの末尾に明示的に配置されます。
- 1 つのポートから受信されたトラフィック タイプごとに、別々のポリシー マップ クラスを設定できます。

物理ポートでポリシー マップを設定する場合には、次の注意事項に従ってください。

- 入力ポートごとに付加できるポリシー マップは、1 つだけです。
- グローバル コンフィギュレーション モードでは、mls qos を有効にする必要があります。
- **set ip dscp** コマンドを入力または使用した場合、デバイスはこのコマンドをその構成内で **set dscp** に変更します。
- ポリシーマップとポート信頼状態は、両方とも物理インターフェイス上で有効にすることができます。ポリシー マップは、ポート信頼状態の前に適用されます。
- **class class-default** ポリシーマップ 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のトラフィッククラスを設定すると、未分類トラフィック（トラフィッククラスで指定された一致基準に一致しないトラフィック）はデフォルトのトラフィッククラス（class-default）として処理されます。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**class-map** *class-map-name* 例： `Device(config)# class-map ipclass1`	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。 デフォルトでは、クラスマップは定義されていません。*class-map-name* には、クラス マップ名を指定します。
ステップ 3	**policy-map** *policy-map-name* 例： `Device(config-cmap)# policy-map flowit`	ポリシー マップ名を入力することによってポリシーマップを作成し、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。 デフォルトでは、ポリシーマップは定義されていません。 ポリシーマップのデフォルトの動作では、パケットが IP パケットの場合は DSCP が 0 に、パケットがタグ付きの場合は CoS が 0 に設定されます。ポリシングは実行されません。

	コマンドまたはアクション	目的
		（注） 既存のポリシーマップを削除するには、**no policy-map** *policy-map-name* グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**class** [*class-map-name* | **class-default**] 例： `Device(config-pmap)# class ipclass1`	トラフィックの分類を定義し、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。 デフォルトでは、ポリシーマップ クラス マップは定義されていません。 すでに **class-map** グローバル コンフィギュレーション コマンドを使用してトラフィッククラスが定義されている場合は、このコマンドで *class-map-name* にその名前を指定します。 **class-default** トラフィッククラスは定義済みで、どのポリシーにも追加できます。このトラフィック クラスは、常にポリシー マップの最後に配置されます。暗黙の **match any** が **class-default** クラスに含まれている場合、他のトラフィッククラスと一致しないパケットはすべて **class-default** と一致します。 （注） 既存のクラスマップを削除するには、 **no class** *class-map-name* ポリシーマップ コンフィギュレーション コマンドを使用します。
ステップ 5	**set dscp** *new-dscp* 例： `Device(config-pmap-c)# set dscp 45`	パケットに新しい値を設定することによって、IP トラフィックを分類します。 • **dscp** *new-dscp* には、分類されたトラフィックに割り当てる新しい DSCP 値を入力します。指定できる範囲は 0 ～ 63 です。
ステップ 6	**police** *rate-bps burst-byte* [**exceed-action** {**drop**}] 例：	分類したトラフィックにポリサーを定義します。

	コマンドまたはアクション	目的
	`Device(config-pmap-c)# police 100000 80000 exceed-action drop`	デフォルトでは、ポリサーは定義されていません。 • *rate-bps* には、平均トラフィック レートをビット/秒（bps）で指定します。指定できる範囲は 8000 ～ 10000000000 です • *burst-byte* には、標準バースト サイズをバイト数で指定します。 • 8000 <= *rate-bps* < 102300000 の場合、*burst-byte* の範囲は 8000 ～ 65535 になります。 • 102300000 <= *rate-bps* < 1023000000 の場合、*burst-byte* の範囲は 8000 ～ 524280 になります。 • 1023000000 <= *rate-bps* <= 10Gig の場合、*burst-byte* の範囲は 8000 ～ 1000000 になります。 • （任意）レートを超過した場合に実行するアクションを指定します。パケットをドロップするには、**exceed-action drop** キーワードを使用します。 （注） 既存のポリサーを削除するには、**no police** *rate-bps burst-byte* [**exceed-action drop**] ポリシーマップ コンフィギュレーション コマンドを使用します。
ステップ 7	**exit** 例： `Device(config-pmap-c)# exit`	ポリシーマップ コンフィギュレーション モードに戻ります。
ステップ 8	**exit** 例：	グローバル コンフィギュレーション モードに戻ります。

	コマンドまたはアクション	目的
	`Device(config-pmap)# exit`	

ステップ 9	**interface** *interface-id*
	例：
	`Device(config)# interface gigabitethernet 1/0/1`
	Or
	`Device(config)# interface fastethernet 1/0/1`
		ポリシーマップを適用するポートを指定し、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。
		有効なインターフェイスには、物理ポートが含まれます。

ステップ 10	**service-policy input** *policy-map-name* 例： `Device(config-if)# service-policy input flowit`	ポリシーマップ名を指定し、入力ポートに適用します。 サポートされるポリシーマップは、入力ポートに 1 つだけです。 （注） ポリシーマップとポートの関連付けを解除するには、 **no service-policy***input policy-map-name* インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 11	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 12	**show policy-map** [*policy-map-name* [**class** *class-map-name*]] 例： `Device# show policy-map`	入力を確認します。

出力キューの特性の設定

ネットワークおよび QoS ソリューションの複雑さに応じて、次のモジュールで示す作業をすべて実行しなければならない場合があります。次の特性を決定する必要があります。

- DSCP 値または CoS 値によって各キューおよびしきい値 ID にマッピングされるパケット
- キューセット（ポートごとの 4 つの出力キュー）に適用されるドロップしきい値の割合、およびトラフィック タイプに必要なメモリの確保量および最大メモリ

- ポートの帯域幅に関するレート制限の必要性
- 出力キューの処理頻度、および使用する技術（シェーピング、共有、または両方）

設定時の注意事項

緊急キューがイネーブルにされているとき、または SRR の重みに基づいて出力キューのサービスが提供されるときには、次の注意事項に従ってください。

- 出力緊急キューがイネーブルにされている場合は、キュー 1 に対して SRR のシェーピングおよび共有された重みが無効にされます。
- 出力緊急キューがディセーブルにされており、SRR のシェーピングおよび共有された重みが設定されている場合は、キュー 1 に対して shaped モードは shared モードを無効にし、SRR はこのキューに shaped モードでサービスを提供します。
- 出力緊急キューがディセーブルで、SRR シェーピング重みが設定されていない場合、SRR はこのキューを共有モードで処理します。

出力キューおよびしきい値 ID への DSCP または CoS 値のマッピング

トラフィックにプライオリティを設定するには、特定の DSCP または CoS を持つパケットを特定のキューに格納し、より低いプライオリティを持つパケットがドロップされるようにキューのしきい値を調整します。キューのデフォルトの数は 4 です。**mls qos srr-queue output queues 8** コマンドを使用して 8 まで増やすことができます。

（注） 出力キューのデフォルト設定は、ほとんどの状況に適しています。デフォルト設定の変更が必要となるのは、出力キューについて完全に理解している場合、およびデフォルトの設定がご使用の QoS ソリューションを満たしていない場合だけです。

DSCP または CoS 値を出力キューおよび ID にマッピングするには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	次のいずれかを使用します。 • **mls qos srr-queue output dscp-map queue** *queue-id* **threshold** *threshold-id dscp1...dscp8*	DSCP または CoS 値を出力キューおよびしきい値 ID にマッピングします。

	コマンドまたはアクション	目的
	• **mls qos srr-queue output cos-map queue** *queue-id* **threshold** *threshold-id cos1...cos8* 例： `Device(config)# mls qos srr-queue output dscp-map queue 1 threshold 2 10 11`	デフォルトでは、DSCP 値 0 ～ 15 はキュー 2 およびしきい値 1 に、DSCP 値 16 ～ 31 はキュー 3 およびしきい値 1 に、DSCP 値 32 ～ 39 および 48 ～ 63 はキュー 4 およびしきい値 1 に、DSCP 値 40 ～ 47 はキュー 1 およびしきい値 1 にマッピングされます。 デフォルトでは、CoS 値 0 および 1 はキュー 2 およびしきい値 1 に、CoS 値 2 および 3 はキュー 3 およびしきい値 1 に、CoS 値 4、6、および 7 はキュー 4 およびしきい値 1 に、CoS 値 5 はキュー 1 およびしきい値 1 にマッピングされます。 • *queue-id* で指定できる範囲は 1 ～ 4 です。 • *threshold-id* で指定できる範囲は 1 ～ 2 です。しきい値 3 のドロップしきい値（%）は事前に定義されています。パーセンテージはキューがいっぱいの状態に対して設定されます。 • *dscp1...dscp8* には、各値をスペースで区切って、最大 8 の値を入力します。指定できる範囲は 0 ～ 63 です。 • *cos1...cos8* には、各値をスペースで区切って、最大 8 の値を入力します。指定できる範囲は 0 ～ 7 です。 （注） デフォルトの DSCP 出力キューしきい値マップまたはデフォルトの CoS 出力キューしきい値マップに戻すには、**no mls qos srr-queue output dscp-map** または **no mls qos srr-queue output cos-map** グローバル コンフィギュレーション コマンドを使用します。

	コマンドまたはアクション	目的
ステップ 3	**mls qos srr-queue output cos-map queue** *queue-id* **threshold** *threshold-id cos1...cos8* 例： `Device(config)# mls qos srr-queue output cos-map queue 3 threshold 1 2 3`	CoS 値を出力キューおよびしきい値 ID にマッピングします。 デフォルトでは、CoS 値 0 および 1 はキュー 2 およびしきい値 1 に、CoS 値 2 および 3 はキュー 3 およびしきい値 1 に、CoS 値 4、6、および 7 はキュー 4 およびしきい値 1 に、CoS 値 5 はキュー 1 およびしきい値 1 にマッピングされます。 • *queue-id* で指定できる範囲は 1 ～ 4 です。 • *threshold-id* で指定できる範囲は 1 ～ 2 です。しきい値 3 のドロップしきい値（%）は事前に定義されています。パーセンテージはキューがいっぱいの状態に対して設定されます。 • *cos1...cos8* には、各値をスペースで区切って、最大 8 の値を入力します。指定できる範囲は 0 ～ 7 です。 （注） デフォルトの CoS 出力キューしきい値マップを返すには、**no mls qos srr-queue output cos-map** グローバル コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show mls qos maps** 例： `Device# show mls qos maps`	入力を確認します。 DSCP 出力キューしきい値マップは、表形式で表示されます。d1 列は DSCP 値の最上位桁、d2 行は DSCP 値の最下位桁を示します。d1 および d2 値の交点がキュー ID およびしきい値 ID です。たとえば、キュー 2 およびしきい値 1（02-01）のようになります。

	コマンドまたはアクション	目的
		CoS出力キューしきい値マップでは、先頭行に CoS 値、2 番めの行に対応するキュー ID およびしきい値 ID が示されます。たとえば、キュー2およびしきい値 2（2-2）のようになります。 デフォルトのDSCP出力キューしきい値マップまたはデフォルトの CoS 出力キューしきい値マップに戻すには、**no mls qos srr-queue output dscp-map** または **no mls qos srr-queue output cos-map** グローバル コンフィギュレーション コマンドを使用します。

出力キューでの SRR シェーピング重みの設定

各キューに割り当てられる使用可能な帯域幅の量を指定できます。重みの比率は、SRR スケジューラが各キューからパケットを送信する頻度の比率です。

出力キューにシェーピング重み、共有重み、またはその両方を設定できます。バースト性のあるトラフィックをスムーズにする、または長期にわたって出力をスムーズにする場合に、シェーピングを使用します。

ポートにマッピングされた4つの出力キューにシェーピング重みを割り当てて、帯域幅のシェーピングをイネーブルにするには、特権EXECモードで次の手順を実行します。この手順は任意です。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**interface** *interface-id*
	例：
	`Device(config)# interface gigabitethernet 1/0/1`
	Or
	`Device(config)# interface fastethernet 1/0/1`
		発信トラフィックのポートを指定し、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。

	コマンドまたはアクション	目的
ステップ 3	**srr-queue bandwidth shape** *weight1 weight2 weight3 weight4* 例： `Device(config-if)# srr-queue bandwidth shape 8 0 0 0`	出力キューにSRR重みを割り当てます。デフォルトでは、weight1 は 25、weight2、weight3、および weight4 は 0 に設定されています。これらのキューは共有モードです。 *weight1 weight2 weight3 weight4* には、シェーピングされるポートの割合を制御する重みを入力します。このキューのシェーピング帯域幅は、インバース比率（1/weight）によって制御されます。各値はスペースで区切ります。指定できる範囲は 0 ～ 65535 です。 重み 0 を設定した場合は、対応するキューが共有モードで動作します。**srr-queue bandwidth shape** コマンドで指定された重みは無視され、**srr-queue bandwidth share** インターフェイス コンフィギュレーション コマンドで設定されたキューの重みが有効になります。シェーピングおよび共有の両方に対して同じキューセットのキューを設定した場合は、必ず番号が最も小さいキューにシェーピングを設定してください。 シェーピング モードは、共有モードを無効にします。 デフォルトの設定に戻す場合は、 **no srr-queue bandwidth shape** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。
ステップ 5	**show mls qos interface** *interface-id* **queueing** 例： `Device(config)# show mls qos interface gigabitethernet 1/0/1 queuing`	入力を確認します。

	コマンドまたはアクション	目的
	Or Device(config)# **show mls qos interface fastethernet 1/0/1 queuing**	

出力キューでの SRR 共有重みの設定

共有モードでは、設定された重みによりキュー間で帯域幅が共有されます。このレベルでは帯域幅は保証されていますが、このレベルに限定されていません。たとえば、特定のキューが空であり、リンクを共有する必要がない場合、残りのキューは未使用の帯域幅を使用して、共有ができます。共有の場合、キューからパケットを取り出す頻度は重みの比率によって制御されます。重みの絶対値は関係ありません。

（注） 出力キューのデフォルト設定は、ほとんどの状況に適しています。出力キューについて十分理解したうえで、この設定がユーザのQoSソリューションを満たさないと判断した場合に限り、設定を変更してください。

ポートにマッピングされた4つの出力キューに共有重みを割り当てて、帯域幅の共有をイネーブルにするには、特権 EXEC モードで次の手順を実行します。この手順は任意です。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**interface** *interface-id* 例： Device(config)# **interface gigabitethernet 1/0/1** Or Device(config)# **interface fastethernet 1/0/1**	発信トラフィックのポートを指定し、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。
ステップ 3	**srr-queue bandwidth share** *weight1 weight2 weight3 weight4* 例： Device(config-id)# **srr-queue bandwidth share 1 2 3 4**	出力キューにSRR重みを割り当てます。デフォルトでは、4つの重みがすべて25です（各キューに帯域幅の1/4が割り当てられています）。 *weight1 weight2 weight3 weight4* には、SRR スケジューラがパケットを送信する頻度の比率を制御する重みを入力しま

	コマンドまたはアクション	目的
		す。各値はスペースで区切ります。指定できる範囲は 1 ～ 255 です。 デフォルトの設定に戻す場合は、 **no srr-queue bandwidth share** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例： `Device(config-id)# end`	特権 EXEC モードに戻ります。

ステップ 5	**show mls qos interface** *interface-id* **queueing**
	例：
	`Device(config)# show mls qos interface gigabitethernet 1/0/1 queuing`
	Or
	`Device(config)# show mls qos interface fastethernet 1/0/1 queuing`
		入力を確認します。

出力緊急キューの設定

出力緊急キューにパケットを入れることにより、特定のパケットのプライオリティを他のすべてのパケットより高く設定できます。SRR は、このキューが空になるまで処理してから他のキューを処理します。

出力緊急キューをイネーブルにするには、特権EXECモードで次の手順を実行します。この手順は任意です。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**mls qos** 例： `Device(config)# mls qos`	スイッチのQoSをイネーブルにします。

	コマンドまたはアクション	目的

ステップ 3	**interface** *interface-id*
	例：
	`Device(config)# interface gigabitethernet 1/0/1`
	Or
	`Device(config)# interface fastethernet 1/0/1`
		出力ポートを指定し、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。

ステップ 4	**priority-queue out** 例： `Device(config-if)# priority-queue out`	デフォルトでディセーブルに設定されている出力緊急キューをイネーブルにします。 このコマンドを設定すると、SRR に参加するキューは 1 つ少なくなるため、SRR 重みおよびキュー サイズの比率が影響を受けます。これは、**srr-queue bandwidth shape** または **srr-queue bandwidth share** コマンド内の *weight1* が無視される（比率計算に使用されない）ことを意味します。 （注） 出力緊急キューをディセーブルにするには、**no priority-queue out** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 5	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。

出力インターフェイスの帯域幅の制限

出力ポートの帯域幅は制限できます。たとえば、カスタマーが高速リンクの一部しか費用を負担しない場合は、帯域幅をその量に制限できます。

（注） 出力キューのデフォルト設定は、ほとんどの状況に適しています。出力キューについて十分理解したうえで、この設定がユーザの QoS ソリューションを満たさないと判断した場合に限り、設定を変更してください。

出力ポートの帯域幅を制限するには、特権EXECモードで次の手順を実行します。この手順は任意です。

手順

	コマンドまたはアクション	目的
ステップ 1	**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 2	**interface** *interface-id*
	例：
	`Device(config)# interface gigabitethernet 1/0/1`
	Or
	`Device(config)# interface fastethernet 1/0/1`
		レートを制限するポートを指定し、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。

ステップ 3	**srr-queue bandwidth limit** *weight1* 例： `Device(config-if)# srr-queue bandwidth limit 80`	ポートの上限となるポート速度の割合を指定します。指定できる範囲は 10 ～ 90 です。 デフォルトでは、ポートのレートは制限されず、100% に設定されています。 （注） デフォルトの設定に戻す場合は、**no srr-queue bandwidth limit** インターフェイス コンフィギュレーション コマンドを使用します。
ステップ 4	**end** 例： `Device(config-if)# end`	特権 EXEC モードに戻ります。

ステップ 5	**show mls qos interface** [*interface-id*] **queueing**
	例：
	`Device(config)# show mls qos interface gigabitethernet 1/0/1 queuing`
	Or
	`Device(config)# show mls qos interface fastethernet 1/0/1 queuing`
		入力を確認します。

標準 QoS のモニタリング

表 *9:* スイッチ上で標準 *QoS* をモニタリングするためのコマンド

コマンド	説明
show mls qos	グローバル QoS コンフィギュレーション情報を表示します。
show mls qos interface [*interface-id*] [**policers** | **queueing** | **statistics**]	バッファ割り当て、ポリサーが設定されているポート、キューイング方式、入出力統計情報など、ポート レベルの QoS 情報が表示されます。
show mls qos maps [**cos-output-q** | **dscp-mutation**]	QoS のマッピング情報を表示します。
show running-config | include rewrite	DSCP 透過性設定を表示します。

QoS の設定例

例：DSCP 信頼状態へのポートの設定および DSCP/DSCP 変換マップの変更

次に、ポートが DSCP を信頼する状態に設定し、着信した DSCP 値 10 ～ 13 が DSCP 値 30 にマッピングされるように DSCP/DSCP 変換マップを変更する例を示します。

```
Device(config)# mls qos map dscp-mutation
10 11 12 13 to 30
Device(config)# interface gigabitethernet 1/0/2
Device(config-if)# mls qos trust dscp
Device(config-if)# end
```

例：ACL によるトラフィックの分類

次に、指定された 3 つのネットワーク上のホストだけにアクセスを許可する例を示します。ネットワーク アドレスのホスト部分にワイルドカード ビットが適用されます。アクセス リストのステートメントと一致しない送信元アドレスのホストはすべて拒否されます。

```
Device(config)# access-list 1 permit 192.5.255.0 0.0.0.255
Device(config)# access-list 1 permit 128.88.0.0 0.0.255.255
Device(config)# access-list 1 permit 36.0.0.0 0.0.0.255
! (Note: all other access implicitly denied)
```

次に、任意の送信元から、DSCP 値が 32 に設定されている任意の宛先への IP トラフィックを許可する ACL を作成する例を示します。

```
Device(config)# access-list 100 permit ip any any dscp 32
```

次に、10.1.1.1 の送信元ホストから 10.1.1.2 の宛先ホストへの IP トラフィック（precedence 値は 5）を許可する ACL を作成する例を示します。

```
Device(config)# access-list 100 permit ip host 10.1.1.1 host 10.1.1.2 precedence 5
```

次に、任意の送信元からアドレス 224.0.0.2 の宛先グループへの PIM トラフィック（DSCP 値は 32）を許可する ACL を作成する例を示します。

```
Device(config)# access-list 102 permit pim any 224.0.0.2 dscp 32
```

次に、任意の送信元から、DSCP 値が 32 に設定されている任意の宛先への IPv6 トラフィックを許可する ACL を作成する例を示します。

```
Device(config)# ipv6 access-list 100 permit ip any any dscp 32
```

次に、10.1.1.1 の送信元ホストから 10.1.1.2 の宛先ホストへの IPv6 トラフィック（precedence 値は 5）を許可する ACL を作成する例を示します。

```
Device(config)# ipv6 access-list ipv6_Name_ACL permit ip host 10::1 host 10.1.1.2
precedence 5
```

次に、2 つの許可（permit）ステートメントを指定したレイヤ 2 の MAC ACL を作成する例を示します。最初のステートメントでは、MAC アドレスが 0001.0000.0001 であるホストから、MAC アドレスが 0002.0000.0001 であるホストへのトラフィックが許可されます。2 番めのステートメントでは、MAC アドレスが 0001.0000.0002 であるホストから、MAC アドレスが 0002.0000.0002 であるホストへの、Ethertype が XNS-IDP のトラフィックのみが許可されます。

```
Device(config)# mac access-list extended maclist1
Device(config-ext-macl)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
Device(config-ext-macl)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
! (Note: all other access implicitly denied)
```

例：クラス マップによるトラフィックの分類

次に、*class1* というクラス マップの設定例を示します。*class1* にはアクセス リスト 103 という一致条件が 1 つ設定されています。このクラスマップによって、任意のホストから任意の宛先へのトラフィック（DSCP 値は 10）が許可されます。

```
Device(config)# access-list 103 permit ip any any dscp 10
Device(config)# class-map class1
Device(config-cmap)# match access-group 103
```

```
Device(config-cmap)# end
Device#
```

次に、DSCP 値が 10、11、および 12 である着信トラフィックと照合する、*class2* という名前のクラス マップを作成する例を示します。

```
Device(config)# class-map class2
Device(config-cmap)# match ip dscp 10 11 12
Device(config-cmap)# end
Device#
```

次に、IP DSCP および IPv6 と照合するクラス マップを設定する例を示します。

```
Device(config)# Class-map cm-1
Device(config-cmap)# match ip dscp 10
Device(config-cmap)# exit
Device(config)# Class-map cm-2
Device(config-cmap)# match ip dscp 20
Device(config-cmap)# exit
Device(config)# Policy-map pm1
Device(config-pmap)# class cm-1
Device(config-pmap-c)# set dscp 4
Device(config-pmap-c)# exit
Device(config-pmap)# class cm-2
Device(config-pmap-c)# set dscp 6
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# interface G1/0/1
Device(config-if)# service-policy input pm1
```

次に、IPv4 トラフィックと IPv6 トラフィックの両方に適用するクラス マップを設定する例を示します。

```
Device(config)# ip access-list 101 permit ip any any
Device(config)# ipv6 access-list ipv6-any permit ip any any
Device(config)# Class-map cm-1
Device(config-cmap)# match access-group 101
Device(config-cmap)# exit
Device(config)# class-map cm-2
Device(config-cmap)# match access-group name ipv6-any
Device(config-cmap)# exit
Device(config)# Policy-map pm1
Device(config-pmap)# class cm-1
Device(config-pmap-c)# set dscp 4
Device(config-pmap-c)# exit
Device(config-pmap)# class cm-2
Device(config-pmap-c)# set dscp 6
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# interface G0/1
Device(config-if)# switch mode access
Device(config-if)# service-policy input pm1
```

例：ポリシーマップを使用した物理ポートのトラフィックの分類、ポリシング、およびマーキング

次に、ポリシー マップを作成し、入力ポートに結合する例を示します。この設定では、IP 標準ACLでネットワーク10.1.0.0からのトラフィックを許可します。この分類にトラフィックが一致した場合、着信パケットの DSCP 値が信頼されます。一致したトラフィックが平均トラフィック レート（48000 bps）、および標準バースト サイズ（8000 バイト）を超過している場合は、（ポリシング済み DSCP マップに基づいて）DSCP はマークダウンされて、送信されます。

```
Device(config)# access-list 1 permit 10.1.0.0 0.0.255.255
Device(config)# class-map ipclass1
Device(config-cmap)# match access-group 1
Device(config-cmap)# exit
Device(config)# policy-map flow1t
Device(config-pmap)# class ipclass1
Device(config-pmap-c)# set dscp cs1
Device(config-pmap-c)# police 1000000 8000 exceed-action drop
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# interface gigabitethernet 2/0/1
Device(config-if)# service-policy input flow1t
```

次に、2 つの許可ステートメントを指定してレイヤ 2 MAC ACL を作成し、入力ポートに結合する例を示します。最初の許可ステートメントでは、MAC アドレスが 0001.0000.0001 であるホストから、MACアドレスが0002.0000.0001であるホストへのトラフィックが許可されます。2 番めの許可ステートメントでは、MAC アドレスが 0001.0000.0002 であるホストから、MAC アドレスが 0002.0000.0002 であるホストへの、Ethertype が XNS-IDP のトラフィックのみが許可されます。

```
Device(config)# mac access-list extended maclist1
Device(config-ext-mac)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
Device(config-ext-mac)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
Device(config-ext-mac)# exit
Device(config)# mac access-list extended maclist2
Device(config-ext-mac)# permit 0001.0000.0003 0.0.0 0002.0000.0003 0.0.0
Device(config-ext-mac)# permit 0001.0000.0004 0.0.0 0002.0000.0004 0.0.0 aarp
Device(config-ext-mac)# exit
Device(config)# class-map macclass1
Device(config-cmap)# match access-group maclist1
Device(config-cmap)# exit
Device(config)# policy-map macpolicy1
Device(config-pmap)# class macclass1
Device(config-pmap-c)# set dscp 63
Device(config-pmap-c)# exit
Device(config-pmap)# class macclass2 maclist2
Device(config-pmap-c)# set dscp 45
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# mls qos trust cos
Device(config-if)# service-policy input macpolicy1
```

次に、分類されていないトラフィックに適用されるデフォルト クラスを使用して、IPv4 と IPv6 の両方のトラフィックに適用されるクラス マップを作成する例を示します。

```
Device(config)# ip access-list 101 permit ip any any
Device(config)# ipv6 access-list ipv6-any permit ip any any
Device(config)# class-map cm-1
Device(config-cmap)# match access-group 101
Device(config-cmap)# exit
Device(config)# class-map cm-2
Device(config-cmap)# match access-group name ipv6-any
Device(config-cmap)# exit
Device(config)# policy-map pm1
Device(config-pmap)# class cm-1
Device(config-pmap-c)# set dscp 4
Device(config-pmap-c)# exit
Device(config-pmap)# class cm-2
Device(config-pmap-c)# set dscp 6
Device(config-pmap-c)# exit
Device(config-pmap)# class class-default
Device(config-pmap-c)# set dscp 10
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# interface G0/1
Device(config-if)# switch mode access
Device(config-if)# service-policy input pm1
```

例：DSCP/DSCP 変換マップの設定

次の例では、DSCP/DSCP 変換マップを定義する方法を示します。明示的に設定されていないすべてのエントリは変更されません（空のマップで指定された値のままです）。

```
Device(config)# mls qos map dscp-mutation 1 2 3 4 5 6 7 to 0
Device(config)# mls qos map dscp-mutation 8 9 10 11 12 13 to 10
Device(config)# mls qos map dscp-mutation 20 21 22 to 20
Device(config)# mls qos map dscp-mutation 30 31 32 33 34 to 30
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# mls qos trust dscp
Device(config-if)# end
Device# show mls qos maps dscp-mutation
Dscp-dscp mutation map:
   mutation1:
     d1 :  d2 0  1  2  3  4  5  6  7  8  9
     ---------------------------------------
      0 :    00 00 00 00 00 00 00 00 10 10
      1 :    10 10 10 10 14 15 16 17 18 19
      2 :    20 20 20 23 24 25 26 27 28 29
      3 :    30 30 30 30 30 35 36 37 38 39
      4 :    40 41 42 43 44 45 46 47 48 49
      5 :    50 51 52 53 54 55 56 57 58 59
      6 :    60 61 62 63
```

（注） 上記の DSCP/DSCP 変換マップでは、変換される値が表形式で示されています。d1 列は元の DSCP の最上位桁、d2 行は元の DSCP の最下位桁を示します。d1 と d2 の交点の値が、変換される値です。たとえば、DSCP 値が 12 の場合、対応する変換される値は 10 です。

例：出力キューの特性の設定

次に、キュー 1 に帯域幅のシェーピングを設定する例を示します。キュー 2、3、4 の重み比が 0 に設定されているので、これらのキューは共有モードで動作します。キュー 1 の帯域幅の重みは 1/8（12.5%）です。

```
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# srr-queue bandwidth shape 8 0 0 0
```

次の例では、出力ポートで稼働する SRR スケジューラの重み比を設定する方法を示します。4 つのキューが使用され、共有モードで各キューに割り当てられる帯域幅の比率は、キュー 1、2、3、および 4 に対して 1/(1 + 2 + 3 + 4)、2/(1 + 2 + 3 + 4)、3/(1 + 2 + 3 + 4)、および 4/(1 + 2 + 3 + 4) になります（それぞれ、10、20、30、および 40%）。キュー 4 はキュー 1 の帯域幅の 4 倍、キュー 2 の帯域幅の 2 倍、キュー 3 の帯域幅の 1 と 1/3 倍であることを示します。

```
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# srr-queue bandwidth share 1 2 3 4
```

次の例では、SRR の重みが設定されている場合、出力緊急キューをイネーブルにする方法を示します。出力緊急キューは、設定された SRR ウェイトを上書きします。

```
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# srr-queue bandwidth shape 25 0 0 0
Device(config-if)# srr-queue bandwidth share 30 20 25 25
Device(config-if)# priority-queue out
Device(config-if)# end
```

次に、ポートの帯域幅を 80% に制限する例を示します。

```
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# srr-queue bandwidth limit 80
```

このコマンドを 80% に設定すると、ポートは該当期間の 20% はアイドルになります。回線レートは接続速度の 80%（800 Mbps）に低下します。ただし、ハードウェアはライン レートを 6% 単位で調整しているため、この値は厳密ではありません。

QoS の機能情報

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能情報
Cisco IOS Release 15.2(7)E1	QoS	ネットワークに QoS を実装すると、ネットワーク パフォーマンスがさらに予測しやすくなり、帯域幅をより効率的に利用できるようになります。

Cisco Feature Navigator を使用すると、プラットフォームおよびソフトウェアイメージのサポート情報を検索できます。Cisco Feature Navigator には、http://www.cisco.com/go/cfn からアクセスします。