

# channel-protocol

To set the protocol that is used on an interface to manage channeling, use the **channel-protocol** command. To deselect the protocol, use the **no** form of this command.

**channel-protocol { lacp | pagp }**

**no channel-protocol**

Syntax Description	lacp	pagp
	Specifies LACP to manage channeling.	Specifies PAGP to manage channeling.

Defaults	pagp
----------	------

Command Modes	Interface configuration
---------------	-------------------------

Command History	Release	Modification
	12.2(18)ZY	Support for this command was introduced.

**Usage Guidelines**

You can also select the protocol using the **channel-group** command.

If the interface belongs to a channel, the **no** form of this command is rejected.

All ports in an EtherChannel must use the same protocol.

PAGP and LACP are not compatible; both ends of a channel must use the same protocol.

The **channel-protocol** command is performed on a channel-group basis and affects ports in the channel group that is being reconfigured only. You can use the **channel-protocol** command to restrict anyone from selecting a mode that is not applicable to the selected protocol.

Configure all ports in an EtherChannel to operate at the same speed and duplex mode (full duplex only for LACP mode). For a complete list of guidelines, refer to the “Configuring EtherChannel” section of the *Catalyst Supervisor Engine 32 PISA Cisco IOS Software Configuration Guide—Release 12.2ZY*.

**Examples** This example shows how to select LACP to manage channeling on the interface:

```
Router(config-if)# channel-protocol lacp
Router(config-if)#
```

Related Commands	Command	Description
	<b>channel-group</b>	Assigns and configures an EtherChannel interface to an EtherChannel group.
	<b>show etherchannel</b>	Displays the EtherChannel information for a channel.

# class-map

To access the QoS class map configuration mode to configure QoS class maps, use the **class-map** command. To delete a class map, use the **no** form of this command.

```
class-map name [match-all | match-any]
```

```
no class-map name [match-all | match-any]
```

## Syntax Description

<i>name</i>	Class map name.
<b>match-all</b>	(Optional) Matches all match criteria in the class map.
<b>match-any</b>	(Optional) Matches one or more match criteria.

## Defaults

When you do not specify the **match-all** or **match-any** keyword, the default is **match-all**.

## Command Modes

Global configuration

## Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

## Usage Guidelines

You apply the **class-map** command and its subcommands on a per-interface basis to define packet classification, marking, aggregate, and flow policing as part of a globally named service policy.

You can attach a service policy to an EtherChannel. Do not attach a service policy to a port that is a member of an EtherChannel.

After you are in QoS class map configuration mode, these configuration commands are available:

- **exit**—Used to exit from QoS class map configuration mode.
- **no**—Used to remove a match statement from a class map.
- **match**—Used to configure classification criteria. These optional **match** subcommands are available:
  - **access-group** {*acl-index* | *acl-name*}
  - **ip** {**dscp** | **precedence**} *value1 value2 ... value8*

These subcommands appear in the CLI help but are not supported on LAN interfaces or WAN interfaces on the OSMs:

- **input-interface** {{*interface interface-number*} | {**null** *number*} | {**vlan** *vlan-id*}}
- **protocol** *linktype*
- **destination-address** **mac** *mac-address*
- **source-address** **mac** *mac-address*

PFC QoS does not support these subcommands:

- **input-interface** *{{interface interface-number} | {null number} | {vlan vlan-id}}*
- **protocol** *linktype*
- **destination-address** *mac mac-address*
- **source-address** *mac mac-address*
- **qos-group** *group-value*

If you enter these subcommands, PFC QoS does not detect the unsupported keywords until you attach a policy map to an interface. When you try to attach the policy map to an interface, you get an error message. For additional information, refer to the *Catalyst Supervisor Engine 32 PISA Cisco IOS Software Configuration Guide—Release 12.2ZY* and the *Cisco IOS Release 12.2 Command Reference* publications.

After you have configured the class-map name and are in class-map configuration mode, you can enter the **match** subcommands. The syntax for these subcommands is as follows:

```
match {{{access-group acl-index} | acl-name} | [{ip dscp} | {precedence value}]}
```

See [Table 2-1](#) for a syntax description of the **match** subcommands.

**Table 2-1** match Syntax Description

Optional Subcommand	Description
<b>access-group</b> <i>acl-index</i>   <i>acl-name</i>	Specifies the access list index or access list names; valid access list index values are from 1 to 2699.
<b>access-group</b> <i>acl-name</i>	Specifies the named access list.
<b>ip dscp</b> <i>value1 value2 ... value8</i>	Specifies the IP DSCP values to match; valid values are from 0 to 63. You can enter up to 8 DSCP values, and separate each value with one white space.
<b>ip precedence</b> <i>value1 value2 ... value8</i>	Specifies the IP precedence values to match; valid values are from 0 to 7. You can enter up to 8 precedence values, and separate each value with one white space.

### Examples

This example shows how to access the **class-map** commands and subcommands, configure a class map named **ipp5**, and enter a match statement for ip precedence 5:

```
Router(config)# class-map ipp5
Router(config-cmap)# match ip precedence 5
Router(config-cmap)#
```

This example shows how to configure the class map to match an already configured access list:

```
Router(config-cmap)# match access-group IPacl1
Router(config-cmap)#
```

### Related Commands

Command	Description
<a href="#">policy-map</a>	Accesses QoS policy-map configuration mode to configure the QoS policy map.
<a href="#">show class-map</a>	Displays class-map information.
<a href="#">show policy-map interface</a>	Displays the statistics and the configurations of the input and output policies that are attached to an interface.

## class-map type multicast-flows

To create multicast class maps and enter the multicast class map configuration mode, use the **class-map type multicast-flows** command. To delete a class map, use the **no** form of this command.

**class-map type multicast-flows** *name*

**no class-map type multicast-flows** *name*

<b>Syntax Description</b>	<i>name</i> Class-map name.
---------------------------	-----------------------------

<b>Defaults</b>	No class is specified.
-----------------	------------------------

<b>Command Modes</b>	Global configuration
----------------------	----------------------

<b>Command History</b>	<b>Release</b>	<b>Modification</b>
	12.2(18)ZY	Support for this command was introduced.

**Usage Guidelines** After you are in the multicast class-map configuration mode, these configuration commands are available:

- **exit**—Used to exit from multicast class-map configuration mode.
- **group**—Used to configure a multicast group range. The syntax for these subcommands is as follows:

**group** *group-addr* [**source** *addr* | **to** *addr*]

See [Table 2-2](#) for a syntax description of the **group** subcommands.

**Table 2-2** *group Syntax Description*

Subcommand	Description
<i>group-addr</i>	Multicast group address.
<b>source</b> <i>addr</i>	(Optional) Specifies the channel-source address.
<b>to</b> <i>addr</i>	(Optional) Specifies the multicast group range end address.

- **no**—Used to negate a command or set its defaults.

---

**Examples**

This example shows how to create a multicast class map:

```
Router(config)# class-map type multicast-flows static2  
Router(config-mcast-flows-cmap)#
```

This example shows how to configure a multicast group range:

```
Router(config-mcast-flows-cmap)# group 192.0.2.0 source 192.0.2.10  
Router(config-mcast-flows-cmap)#
```

## class (policy-map)

To specify the name of the class that has a policy that you want to create or change or to specify the default class (commonly known as the class-default class) before you configure its policy, use the **class** command in QoS policy-map configuration mode. To remove a class from the policy map, use the **no** form of this command.

```
class { class-name | class-default }
```

```
no class { class-name | class-default }
```

### Syntax Description

<i>class-name</i>	Name of the class to configure or modify the policy.
<b>class-default</b>	Specifies the default class.

### Defaults

No class is specified.

### Command Modes

QoS policy-map configuration

### Command History

Release	Modification
12.2(18)ZY	Support for this command was introduced.

### Usage Guidelines

You can use the **class** (policy-map) command to specify the class name of the policy that you want to create or change. You must first identify the policy map.

To identify the policy map (and enter the required QoS policy-map configuration mode), use the **policy-map** command before you use the **class** (policy-map) command. After you specify a policy map, you can configure the policy for new classes or modify the policy for any existing classes in that policy map.

To define the class characteristics, use the following guidelines:

- The class name that you specify in the policy map ties the characteristics for that class—that is, its policy—to the class map and its match criteria, as configured using the **class-map** command.
- When you configure a policy for a class, specify its bandwidth, and attach the policy map to an interface, CBWFQ determines if the bandwidth requirement of the class can be satisfied. If so, CBWFQ allocates a queue for the bandwidth requirement.
- When a class is removed, available bandwidth for the interface is incremented by the amount that was previously allocated to the class.
- The maximum number of classes that you can configure within a policy map is 64.

The **class-default** keywords are used to specify the predefined default class called class-default. The predefined default class called class-default is the class to which traffic is directed if that traffic does not match any of the match criteria in the configured class maps.

You can define a class policy to use either tail drop by using the **queue-limit** command or WRED by using the **random-detect** command. When using either tail drop or WRED, follow these guidelines:

- The **queue-limit** and **random-detect** commands cannot be used in the same class policy, but they can be used in two class policies in the same policy map.
- You can use the **bandwidth** command when either the **queue-limit** or the **random-detect** command is configured in a class policy. The **bandwidth** command specifies the amount of bandwidth allocated for the class.
- For the predefined default class, you can use the **fair-queue** (class-default) command. The **fair-queue** command specifies the number of dynamic queues for the default class. The **fair-queue** command can be used in the same class policy as either the **queue-limit** or **random-detect** command; it cannot be used with the **bandwidth** command.

## Examples

This example shows how to configure three class policies included in the policy map called policy1. Class1 specifies the policy for the traffic that matches access control list 136. Class2 specifies the policy for the traffic on interface ethernet101. The third class is the default class to which packets that do not satisfy configured match criteria are directed.

```
Router(config)# policy-map policy1
Router(config-pmap)# class-map class1
Router(config-pmap-c)# match access-group 136
Router(config-pmap)# class-map class2
Router(config-pmap-c)# match input-interface ethernet101
```

These examples show how to create the policy map that contains the policy specifications for class1, class2, and the default class:

```
Router(config)# policy-map policy1
Router(config-pmap)# class-map class1
Router(config-pmap-c)# bandwidth 2000
Router(config-pmap-c)# queue-limit 40

Router(config-pmap)# class class2
Router(config-pmap-c)# bandwidth 3000
Router(config-pmap-c)# random-detect
Router(config-pmap-c)# random-detect exponential-weighting-constant 10

Router(config-pmap)# class class-default
Router(config-pmap-c)# fair-queue 16
Router(config-pmap-c)# queue-limit 20
```



### Note

When the policy map containing these classes is attached to the interface to stipulate the service policy for that interface, available bandwidth is assessed, including all class policies and the Resource Reservation Protocol (RSVP), if configured.

This example shows how to configure the policy for the **class-default** default class included in the policy map called policy8. The **class-default** default class has 20 hashed queues for the traffic that does not meet the match criteria of the other classes that have policies that are defined by the policy map called policy8 and a weight factor of 14 that is used to calculate the average queue size. For congestion avoidance, WRED packet drop is used, not tail drop.

```
Router(config)# policy-map policy8
Router(config-pmap)# class class-default
Router(config-pmap-c)# fair-queue 20
Router(config-pmap-c)# random-detect exponential-weighting-constant 14
```

Related Commands	Command	Description
	<b>class-map</b>	Accesses the QoS class map configuration mode to configure QoS class maps.
	<b>fair-queue</b>	Specifies the number of dynamic queues to be reserved for use by the class-default class as part of the default class policy.
	<b>policy-map</b>	Accesses the QoS policy-map configuration mode to configure the QoS policy map.
	<b>queue-limit</b>	Specifies or modifies the maximum number of packets that the queue can hold for a class policy configured in a policy map.
	<b>random-detect (interface)</b>	Enables WRED or DWRED.
	<b>random-detect exponential-weighting-constant</b>	Configures the WRED and DWRED exponential weight factor for the average queue size calculation for the queue.
	<b>random-detect precedence</b>	Configures the WRED and DWRED parameters for a particular IP precedence.