# HP Switch Software OpenFlow Administrator's Guide K/KA/WB 15.14


## Abstract

This document describes the general steps and individual commands for enabling OpenFlow operation on HP Switches.

**Applicable Products**

HP Switch 2920 series
HP Switch 3500 series
HP Switch 3800 series
HP Switch 5400 series, v1 and v2 modules
HP Switch 6200 series
HP Switch 6600 series
HP Switch 8200 series, v1 and v2 modules



HP Part Number: 5998-4400
Published: October 2013
Edition: 2

**Acknowledgments**

This product contains OpenFlow protocol support functionality as provided in Open vSwitch code licensed under the Apache Software License v2.0 and available at http://openvswitch.org. The full text of the Apache Software License v2.0 is contained in the *OpenFlow License Information* document available at www.hp.com/Networking/support.

**Warranty**

For the software end user license agreement and the hardware limited warranty information for HP Networking products, visit www.hp.com/networking/support.

# Contents

# 1 Introduction

This document provides the following:

- General steps for OpenFlow configuration and administration
- OpenFlow command syntax descriptions, including show commands
- OpenFlow troubleshooting commands and debug actions

This document only covers the additional features and commands for administering OpenFlow on certain HP switches that use software version 15.10 or later, as described below:

| Release Version | Description |
|---|---|
| K/KA.15.10 | Added OpenFlow 1.0 support for the following switches:<br>• HP 3500, HP 3500yl<br>• HP 3800<br>• HP 5400 zl with v1 or v2 modules<br>• HP 6200 yl<br>• HP 6600<br>• HP 8200 zl with v1 and v2 modules |
| K/KA.15.11 | Added HP QoS Extensions to OpenFlow |
| K/KA.15.12 and WB.15.12 | Added OpenFlow support for the HP 2920 |
| K/KA/WB 15.14 | Added OpenFlow v1.3 support |

For more information about upgrading software, see the "Software Management" chapter in the *Management and Configuration Guide* for your HP switch.

## Conceptual overview

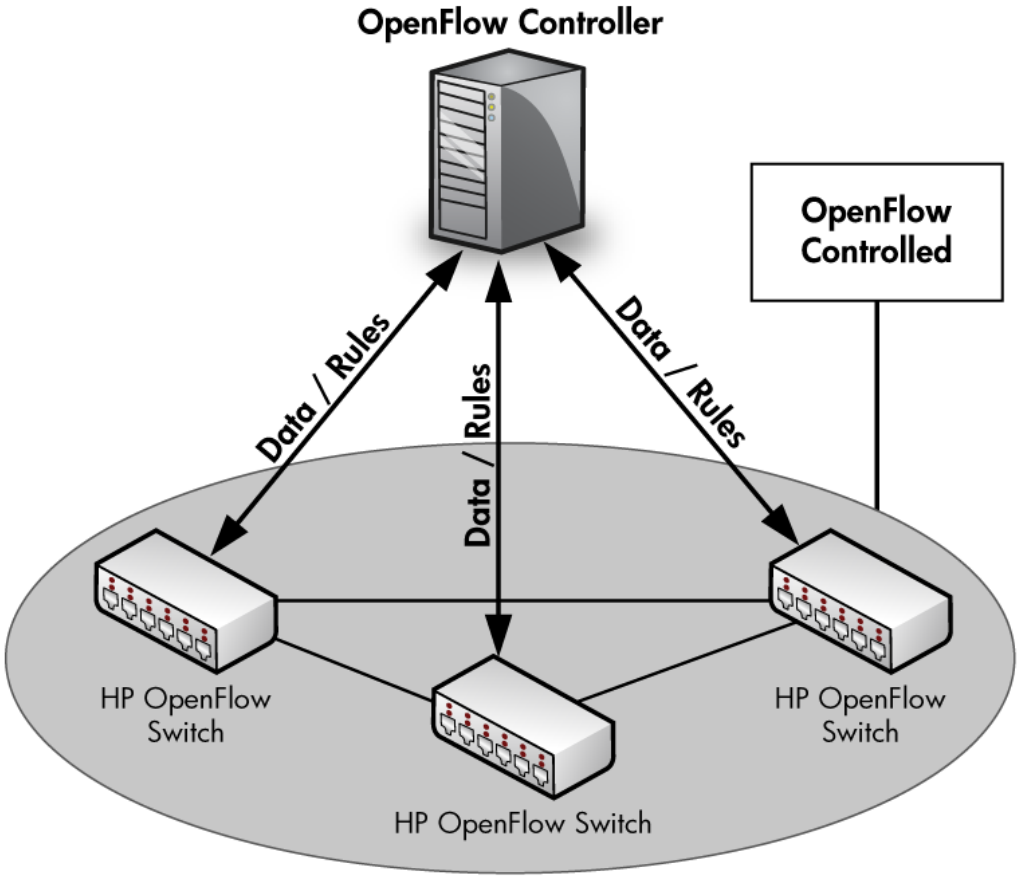
OpenFlow is a programmable open-standard network protocol that uses flexible matching rules to classify and manage network traffic into flows. OpenFlow defines a set of actions that network devices can take to manage these flows. An OpenFlow controller defines and communicates policies to specify traffic behavior on OpenFlow switches. OpenFlow separates the control plane (that decides how traffic must be forwarded) from the data plane (that implements how traffic is forwarded.)

OpenFlow is based on an Ethernet switch with internal flow-tables and a standardized interface to add and remove flow entries via an external controller.

OpenFlow is a software environment that allows for experimentation of networking protocols and traffic flows without interrupting the operation of production network. OpenFlow traffic can be separated from the rest of the traffic on the network per VLAN segregate, so that non-OpenFlow traffic is not impacted by OpenFlow.

OpenFlow implementation on HP Switches separates OpenFlow traffic and production traffic with OpenFlow instances. Traffic within an OpenFlow instance does not influence or degrade production traffic. OpenFlow configuration commands are applied per-instance.

Figure 1 OpenFlow Switches and controller




HP implementation complies with OpenFlow Switch Specification v1.0.0 (December 31, 2009.) HP updated the OpenFlow Switch Specification to version 1.3 (July 2013). For limitations on the HP implementation, see "Supported RFCs and standards" (page 12).

For more information see the Open Networking Foundation website at https://www.opennetworking.org/.

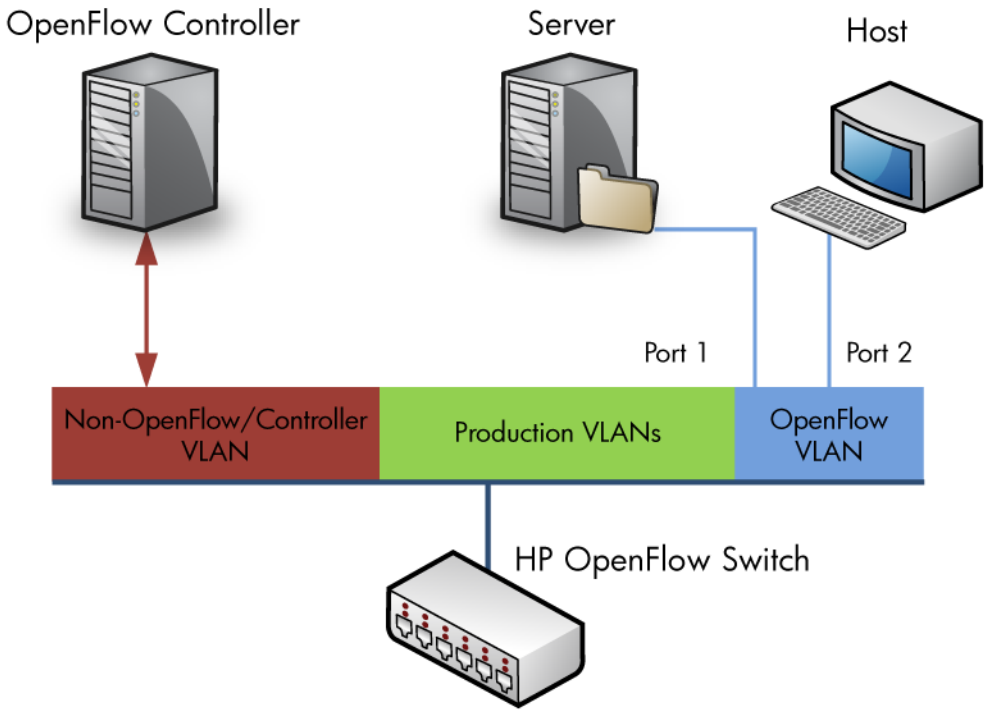
# OpenFlow architecture

OpenFlow can be configured to separate production traffic from OpenFlow traffic. OpenFlow traffic uses either the Virtualization or Aggregation Mode.

**Virtualization mode**

Virtualization mode allows production VLANs and VLANs that belong to OpenFlow instances to be configured on the switch. Each OpenFlow Instance is independent and has its own OpenFlow configuration and OpenFlow controller connection. A VLAN in virtualization mode must be a member of an OpenFlow instance.

Figure 2 Virtualization mode




### Aggregation mode

In Aggregation mode, all VLANs in the switch are part of an OpenFlow instance. The exception is the management VLAN and a VLAN that communicates to the controller. Similar to a lab environment the OpenFlow controller manages all the switching and routing for the switch.

**NOTE:** When Aggregation is configured, there is only OpenFlow traffic, no production traffic.

**Figure 3 Aggregation mode**

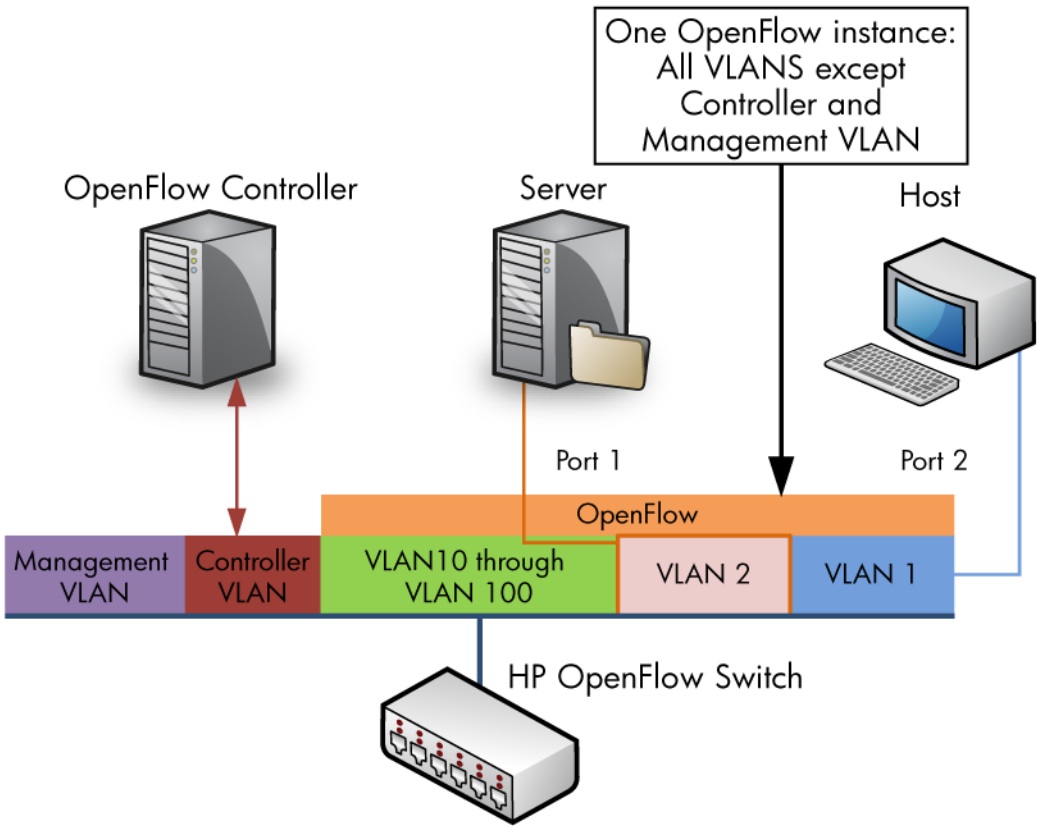
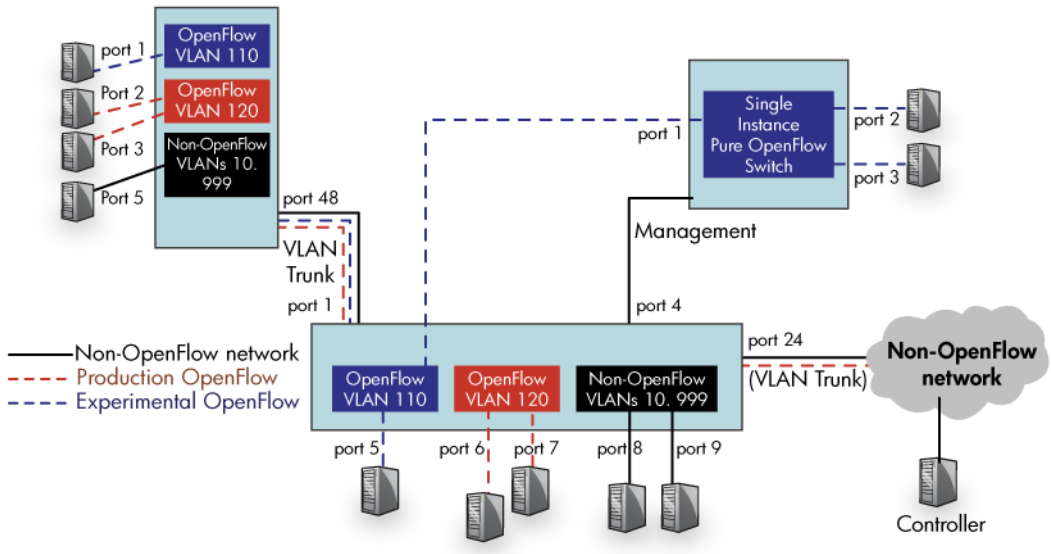



**Figure 4 Example network with production non-OpenFlow, production OpenFlow, and experimental OpenFlow**




# OpenFlow features and benefits

With the addition of OpenFlow Specification 1.3, the following features are supported:

- Multiple Flow tables
    - Pipeline processing
- OpenFlow physical, logical and reserve ports
- Version negotiation
- Group tables
- Auxiliary connections
- OpenFlow Extensible Match (OXM)
- Multiple controllers
- Support for IPv6 flows

OpenFlow switch side configuration enables the user to:

- Enable or disables OpenFlow
- Create OpenFlow instances and configures controller connections
- Display OpenFlow related configurations
- Availability of Config support to retain OpenFlow configuration across a reboot

OpenFlow supports high availability:

- The OpenFlow flow table is preserved across Management Module failover
- The OpenFlow configuration is synced from the AMM to the SMM

OpenFlow includes tools for limiting resources:

- Support for limiting the percentage of policy engine and IP control table resources used by OpenFlow
- Support for rate-limiting the amount of OpenFlow traffic sent to the CPU and from there to the controller
- Support for rate-limiting the amount of OpenFlow traffic that gets forwarded by the policy engine rules programmed by OpenFlow
- Support for hardware-only mode where only flows that can be programmed into hardware are accepted from the controller.

OpenFlow modes of operation:

- Support for hardware-only mode such where only flows that can be programmed into hardware are accepted from the controller.
- Support for active mode (default) where new flows are sent to the controller by the switch.
- Support for passive mode where new flows no longer are sent to the controller but are handled normally handled by the switch.

**IPv6 and OpenFlow**

Controlling IPv6 traffic using OpenFlow is supported beginning with OpenFlow Specification 1.3. For more information on configuring IPv6 traffic, see the *IPv6 Configuration Guide* for your switch.

# Administrative methods

This document provides the HP CLI commands for configuring and administering HP OpenFlow switches.

OpenFlow controllers include utilities for monitoring, administering, and troubleshooting OpenFlow switches. For example, the OpenFlow switch controller distribution includes the utility `ovs-ofctl`. The utility can show the current state of a switch that supports OpenFlow, including features, configuration and table entries. Other controllers have similar utilities; see the documentation for your controller for the complete command set.

# Supported RFCs and standards

HP implementation complies with OpenFlow Switch Specification to version 1.3 (July 2013) from the Open Networking Foundation, https://www.opennetworking.org/ with some differences.

Unsupported features:

- TABLE action.
- IN_PORT action for Hardware and Software tables. You cannot send out the packet on a port on which it arrived.
- The “enqueue” action.
- Handling of IP Fragments: OFPC_IP_REASM/OFPC_FRAG_REASM.
- The flow emergency cache implementation.
- Strip VLAN action is supported on both Policy Engine Table and Software Table.
- Some commands for port modification from a controller:
  - OFPPC_PORT_DOWN
  - OFPPC_NO_STP
  - OFPPC_NO_RECV
  - OFPPC_NO_RECV_STP
  - OFPPC_NO_FWD

**NOTE:** When the above commands are sent from the controller, an error message is returned to the controller:

```
OFPET_PORT_MOD_FAILED
```

Hardware differences between v1 & v2 Modules affect feature functionality, see “Flow classification on v1 and v2 modules” (page 89) for details.

# Interoperability

**Table 1 HP Switch features and interoperability with OpenFlow — by affect on feature or application**

| Affect | Feature |
|---|---|
| Feature can override OpenFlow [1] | 802.1X<br>MAC Auth<br>MAC Lockout<br>MAC Lockdown<br>Port Security<br>Web Auth |
| Feature can override OpenFlow [2] | ACLs – Port, VLAN, Router, IDM variants<br>IDM |
| Feature can override OpenFlow [3] | Rate Limiting |
| Feature can be configured if OpenFlow is used | Management VLAN<br>**NOTE:** Management VLAN feature can be configured but it cannot be part of an OpenFlow instance. |
| | Q-in-Q |

**Table 1 HP Switch features and interoperability with OpenFlow — by affect on feature or application** *(continued)*

| Affect | Feature |
|---|---|
| | Remote Mirror Endpoint |
| | Transparent Mode |
| Feature cannot be configured if OpenFlow is used [4] | Meshing |
| OpenFlow can override this feature [5] | DHCP Snooping |
| | DHCPv4 client |
| | DHCPv4 relay |
| | DHCPv6 client |
| | DNS |
| | Ping |
| | SNTP |
| | Telnet client and server |
| | TFTP |
| | TimeP |
| | Traceroute |
| | UDP broadcast forwarder |
| OpenFlow can override this feature [5] | BGP |
| | DHCPv6 relay |
| | Dynamic ARP Protection |
| | Dynamic IP Lockdown |
| | IGMP Proxy |
| | IGMPv2 |
| | IGMPv3 |
| | MLDv1 |
| | MLDv2 |
| | OSPFv2 |
| | OSPFv3 |
| | PIM-DM |
| | PIM-SM |
| | RIP |
| | Static Multicast Routes |
| | Static Routes |
| | Virus Throttling |
| | VRRP |
| OpenFlow does not affect this feature | Support existing L2, L3, security, HA, QoS functionalities |
| OpenFlow does not affect this feature [6] | Distributed Trunking |
| | GVRP |

### Table 1 HP Switch features and interoperability with OpenFlow — by affect on feature or application *(continued)*

| Affect | Feature |
|---|---|
| | LACP |
| | Loop Protect |
| | sFlow |
| | UDLD |
| OpenFlow does not affect this feature [7] | STP loop guard |
| | BPDU guard |
| | MSTP |
| | RSTP |
| | STP |
| | PVST |

[1] These authentication features still function in an OpenFlow instance and ports of an OpenFlow instance. The security features take a first look at the packet in VLS before sending the packets to OpenFlow.

[2] Any ACL entry that sets a drop bit in hardware (TCAM) would always win over the TCAM entry to copy OpenFlow traffic to the CPU such that packets on an OpenFlow instance could get dropped in hardware due to an ACL entry and OpenFlow would never be able to see those packets.

[3] Rate Limiting may be applied to limit OpenFlow traffic as well as other traffic. OpenFlow uses a form of rate-limiter to limit the OpenFlow traffic that gets to the CPU.

[4] Enabling meshing can break the distinction between OpenFlow VLANs and non-OpenFlow VLANs.

[5]
- The OpenFlow controller could set up a flow to match a protocol header and an action to drop the matching packets. This could lead to the protocol's packets never making it to the protocol handling code in the software data path causing the protocol to break on the OpenFlow instance.
- The OpenFlow controller could set up a flow to match a protocol header and a NORMAL action in software for the matching packets. In such a case, the protocol's packets are removed by OpenFlow in the software data path but reintroduced after examining the software flow table. Though this action may not break the protocol, it introduces an additional latency before the protocol code gets the protocol's packets.

[6] Protocol packets are not sent through the OpenFlow software data path.

[7] Port up or down events are sent to the controller to keep the controller aware of available ports on the switch. OpenFlow cannot override STP, RSTP, or MSTP decisions.

# Scalability

### Table 2 Switch modules scalability

| Switch/Modules | # flows v1.0 | # flows v1.3 |
|---|---|---|
| Compatible mode – "allow v1 modules" – A chassis where v1 as well as v2 modules are present may execute in this mode. Non-compatible mode – "no allow v1 modules" – A chassis that only has v2 modules may execute in this mode. | | |
| 8200/5400 v1 modules 3500 series 6600 series 6200 series | Total: 64 K | Total: 64 K |
| | Hardware: TCAM – 1.5K per slot | Hardware: Standard match mode TCAM – 1.5 K per slot |
| | Software: Total minus Hardware | Software: Total minus Hardware |
| 8200/5400 v2modules 3800 series 2920 series (Flow numbers will be lower for this series) | Total: 64 K<br>16 K for 2920 | Total: 64 K (may be higher)<br>16 K for 2920 |
| | Hardware: TCAM - 2000 per slot<br>TCAM – 500 per slot for 2920 | Hardware: Compatible mode Standard match mode TCAM – 1.5 K per slot |

**Table 2 Switch modules scalability** *(continued)*

| Switch/Modules | # flows v1.0 | # flows v1.3 |
|---|---|---|
| | | Non-compatible mode TCAM – 4K per slot |
| | Software: Total minus Hardware | Software: Total minus Hardware |

# 2 Configuring OpenFlow

## Configuration overview

1. Enable OpenFlow
2. Configure OpenFlow instances
3. Configure OpenFlow instance members
4. Set OpenFlow instance mode
5. Set Flow location
6. Configure software and hardware rate limiting
7. Configure listener ports
8. Configure controller IP and port
9. Configure policy engine resources

## Entering OpenFlow

### Entering OpenFlow context

You can use the `openflow` command options from configuration level by entering the word `openflow`, or from OpenFlow context level by typing the command option.

#### Syntax

`openflow`
Enters OpenFlow context

### Entering OpenFlow instance context

You can use the `instance` *`instance-name`* command from configuration level by beginning the command with `openflow`, or from OpenFlow instance context level by typing the command option.

#### Syntax

`openflow instance` *`instance-name`*
Enters OpenFlow instance context

*`instance-name`*
OpenFlow instance name

## Preparing for configuration

Plan your network including production and OpenFlow VLANs, OpenFlow instances, OpenFlow controller ports, listening ports, naming and numbering strategy.

Plan the number of VLANs configured for OpenFlow versus non-OpenFlow.

OpenFlow works on an instance only when OpenFlow is enabled on the instance as well as globally on the switch.

**NOTE:**

A maximum of 128 OpenFlow instances can be configured.
A maximum of 2048 VLANs are supported.

## Enabling or disabling OpenFlow

Enable or disable OpenFlow globally:

### Syntax

```
openflow [ enable | disable ]
[no] openflow enable
```

`enable`

Enables OpenFlow globally.

`disable`

Disables OpenFlow globally.

**NOTE:** Using `no openflow` without any additional parameters deletes **all** OpenFlow configurations. A warning message to confirm this command appears.

**NOTE:** OpenFlow instance parameters can only be changed with OpenFlow enabled. Instance parameters cannot be changed when instance is enabled. To enabling an instance use the following command.

```
openflow instance <instance name> enable
```

## Setting OpenFlow protocol version

### Syntax

```
openflow-instance-name # version 1.0|1.3 only
```

Default version:1.0

OpenFlow protocol version supported by the instance.

This command lets you choose which version of OpenFlow the instance will use to negotiate with the controller. The command also allows for supported earlier versions of OpenFlow to be used in negotiation with the controller unless the option `only` is specified.

Default: version 1.0

# Configuring OpenFlow instances

Configures an OpenFlow instance.

**NOTE:**

- Configuration changes are not allowed when instance is enabled. Disable the instance and make instance specific configuration changes.
- When an OpenFlow instance is enabled, a policy-engine resource directs traffic on an OpenFlow VLAN to the OpenFlow module.
- For a named instance to be enabled, a listen port or a controller, and a member VLAN has to have been added to the instance.
- To enable an aggregate instance, a listen-port or a controller has to have been added to the instance.

For configuring Aggregation Mode, see "Aggregation mode" (page 37)

For configuring Virtualization Mode, see "Virtualization mode" (page 35)

### Syntax

```
openflow instance { instance-name | aggregate }[ enable |
disable ]
[no] openflow instance { instance-name | aggregate }
enable
```

The `no` form of the command deletes **all** OpenFlow configurations for the instance.

*`instance-name`*

Creates an OpenFlow instance.

Instance names can have a maximum length of 32 case-insensitive alphanumeric characters, numerals, and underscore.

`aggregate`

Creates an OpenFlow instance that includes all VLANs except the management VLAN and the OpenFlow controller VLANs. See "Aggregation mode" (page 37) for details on the use of this parameter.

`enable`

Enables the named OpenFlow instance or named aggregate.

`disable`

Disables the named OpenFlow instance or named aggregate.

## OpenFlow instance mode

OpenFlow can work either in *`active`* or *`passive`* mode.

Active mode

New packets of a flow that the switch is not aware of are sent to the OpenFlow controller.

Passive mode

There is one-way communication from the OpenFlow controller to the switch. Packets that do not match any flow in the flow table on the switch are not sent to the controller. Such packets of new flows are handled normally by the switch.

**NOTE:** This option is only available for an OpenFlow version 1.0 instance.

This command sets operation mode for an OpenFlow instance.

### Syntax

```
openflow instance { instance-name | aggregate }
mode { active | passive }
```

`instance` *`instance-name`*

Sets the mode for the named instance.

`aggregate`

Sets the mode for the aggregate instance.

`active`

New flows are redirected to the controller for the instance.

`passive`

New flows are not sent to the controller for the instance.

Default: active

## Configure OpenFlow instance members

Configures OpenFlow instance members.

- Only one VLAN can be added as a member of an OpenFlow instance.
- The same VLAN cannot be added as a member of multiple OpenFlow instances.
- The management VLAN cannot be added to an OpenFlow instance as a member VLAN.
- A Controller VLAN cannot be added to an OpenFlow instance as a member VLAN.

### Syntax

[no] openflow instance *instance-name*
member vlan *vlan-id*

*instance-name*

Add a member to this OpenFlow instance.

vlan *vlan-id*

Adds the VLAN to the named OpenFlow instance.

# Software flows

## Flow location

This command sets the location of flows for an instance or the aggregate. In hardware-only mode, flows are programmed only in hardware. The flows are located in hardware and software by default.

### Syntax

[no] openflow instance { *instance-name* | aggregate } flow-location
hardware-only

*instance-name*

Sets flow location for the named instance.

aggregate

Sets flow location for the aggregate instance.

hardware-only

Sets the location of flows to hardware only.

Default: Software and hardware.

**NOTE:** An error is returned to the controller if the flow cannot be added in hardware and the flow-location is set as hardware-only.

**NOTE:** Flows with an action to forward to multiple ports or all ports of a VLAN, such as flood, cannot be hardware accelerated. Such flows are handled in software. Changing flow location to hardware-only affects those flows.

For example, if a flow is added with action such as FLOOD, it can only go in software. This causes in a performance penalty or the flow not being programmed at all if running in hardware-only mode.

## Configuring number of software flow tables per instance

### Syntax

openflow-instance-name # software-flow-table *value*

Configures the number of software flow tables required for an instance.

Default: 1, Range: 1–4

## Provisioning ports which are not up

OpenFlow provides the ability to program flows for ports which are not up yet. Pre-provisioning provides port state notification to the controller as part of OFPT_PORT_STATUS message:

–OFPPS_LINK_DOWN - No physical link present

–OFPPS_BLOCKED - Port is blocked

–OFPPS_LIVE - Live for Fast Failover Group

# Auxiliary Connections

## Configuring auxiliary connections

### Syntax

`openflow # auxiliary-connection` *`index`* `port` *`port-number`* `type tcp|udp`

Creates an auxiliary connection with a unique index which later is associated with the instance controller main connection. Auxiliary connection use the same source IP address and the datapath ID as the main connection. The main connection auxiliary ID is set to zero, while the auxiliary connection ID is set to 1. Only one auxiliary connection is supported per main connection. and transport protocol options for auxiliary connections can be either TCP or UDP.

The packets supported on a auxiliary channel:

- OFPT_HELLO
- OFPT_ERROR
- OFPT_ECHO_REQUEST/ REPLY
- OFPT_FEATURES_REQUEST/REPLY
- OFPT_PACKET_IN
- OFPT_PACKET_OUT

The main use auxiliary connections is transactions related to OFPT_PACKET_IN/OFPT_PACKET_OUT.

### Options

index
: Unique identifier for an auxiliary connection.

port
: Protocol port on which the controller can be reached.

type
: Type of transport protocol to be used: TCP or UDP.

`[no]`
: Removes the auxiliary connection.

**NOTE:** Auxiliary connections are terminated when the main connection goes down or is closed by the user or when the instance/openflow is disabled.

TLS is not supported for Auxiliary connections.

### Example

```
HP-Stack-3800(openflow)# auxiliary-connection 1 port 6633 type tcp
HP-Stack-3800(openflow)# inst t1
HP-Stack-3800(of-inst-t1)# controller-id 1 auxiliary-connection 1

HP-E5406zl(config)# shopenflow instance t1
Configured OF Version    : 1.3
Negotiated OF Version    : NA
Instance Name            : t1
Admin. Status            : Enabled
Member List              : VLAN 2
Listen Port              : 6633
Oper. Status             : Down
```

```
Oper. Status Reason      : NA
Datapath ID              : 0000002320e877fe
Mode                     : Active
Flow Location            : Hardware and Software
No. of Hw Flows          : 0
No. of Sw Flows          : 0
Hw. Rate Limit           : 0 kbps
Sw. Rate Limit           : 100 pps
Conn. Interrupt Mode     : Fail-Secure
Maximum Backoff Interval : 60 seconds
Probe Interval           : 10 seconds
Hw. Table Miss Count     : NA
No. of Sw Flow Tables    : 1
Egress Only Ports        : None
Table Model              : Single Table

Controller Id Connection Status Connection State Secure Role
------------- ----------------- ---------------- ------ ------
1                Disconnected        Void               No     Equal

        Auxiliary                    Auxiliary      Auxiliary
 Controller Id Conn. index Auxiliary ID  Conn. Status  Conn. State  Type
 ------------- ----------- ------------  -------------  -----------  ----
1              1             1              Disconnected   Void           TCP


#HP-8206zl# show run
openflow
 controller-id 1 ip 20.0.0.2 controller-interface vlan 1
  auxiliary-connection 1 port 6633 type tcp
  instance "t1"
   listen-port
  member vlan2
  controller-id 1 auxiliary-connection 1
  version 1.3
  enable
   exit
 enable
  exit
```

# OpenFlow Controllers

## Configuring an OpenFlow controller

### Syntax

`openflow # controller [`*`controller-id`*`]`
[no] `controller-id` *`1-128`*

Range: 1–128

## Associating the auxiliary connection index

This is for a non-secure association with the controller id.

### Syntax

`openflow-instance-name # Auxiliary-connection 1`

Auxiliary connection index to be associated with the main connection

Range: 1–128

`auxiliary connection` *`1`*

Calls out only connection supported per main controller connection.

## Configuring instance controller secure association

### Syntax

[no] controller-id *controller-id secure* auxiliary-connection *aux-id*

This command:

- Configures instance controller association for a main controller connection.
- Secures the instance controller connection. Also available for OpenFlow version 1.0.
- Supports CA signed certificates. For CA signed certificates, same ROOT certificate is used to sign both controller and switch certificate.
- Supports mutual authentication.

Up to three controllers can be associated per OpenFlow instance.

secure

Initiates a TLS connection with the controller (TLS version 1.0 or greater.)

auxiliary-connection

Forms an auxiliary connection between the instance and the controller. Only one auxiliary connection is supported per main controller connection.

**NOTE:** This option is available for instances compatible with OpenFlow version 1.3.

[no]

Removes the auxiliary connection.

### Example

```
HP-3500yl-48G-PoEP(of-inst-t1)# show openflow instance t1
Configured OF Version    : 1.3
Negotiated OF Version    : 1.3
Instance Name            : t1
Admin. Status            : Enabled
Member List              : VLAN 3
Listen Port              : None
Oper. Status             : Up
Oper. Status Reason      : NA
Datapath ID              : 0003b499ba86bf80
Mode                     : Active
Flow Location            : Hardware and Software
No. of Hw Flows          : 0
No. of Sw Flows          : 0
Hw. Rate Limit           : 0 kbps
Sw. Rate Limit           : 100 pps
Conn. Interrupt Mode     : Fail-Secure
Maximum Backoff Interval : 60 seconds
Probe Interval           : 10 seconds
Hw. Table Miss Count     : NA
No. of Sw Flow Tables    : 1
Egress Only Ports        : None
Table Model              : Policy Engine and Software

Controller Id Connection Status  Connection State Secure Role
------------  -----------------  ----------------  ------  ------
 1               Connected           Active             Yes    Equal
```

## Adding a controller to an OpenFlow instance

### Syntax

`openflow-instance-name [#] controller-id [number]`

Adds a controller to an OpenFlow instance.

Range: 1–3 controllers can be associated per OpenFlow instance. Maximum of 3 controllers can be linked to an OpenFlow instance, the value of controller-id can be more than 3.

## Configuring IP Control Table Mode

Include IP control table in the OpenFlow packet processing pipeline. Default disabled.

### Syntax

`openflow # [no]`
`ip-control-table-mode`

Include IP control table in the OpenFlow packet processing pipeline. Default disabled.

## Configuring connection

A controller is identified by its IP address and a connection port. Each OpenFlow instance can have up to 3 controllers. OpenFlow controllers can be added or deleted using this command.

### Syntax

`openflow controller-id id ip ip-address [ port tcp-port]`
`controller-interface { vlan vlan-id | oobm }`
`[no]`
`openflow controller-id ID`

Up to 3 OpenFlow controller connections are supported per OpenFlow instance.

*id*

OpenFlow controller identification number.

The `no` removes the identified controller, if the controller is not in use by any OpenFlow instances.

Range: 1 — 128

*ip-address*

OpenFlow controller IP address.

*tcp-port*

Optional: Specify the interface through which to connect to a controller.

Default: port number 6633

Range: port numbers 1024 - 65536

`controller-interface`

The `[no]` form of the command with this parameter deletes the OpenFlow controller connection.

*vlan-id*

Connect to the OpenFlow controller through the identified VLAN.

**NOTE:** A VLAN that is a member of an OpenFlow instance cannot be added as an OpenFlow controller interface.

`oobm`

Connect to the OpenFlow controller through the OOBM interface. Only applicable for switches that have a separate out-of-band management (OOBM) port.

### Configuring IPv4 connections between an instance and controller

#### Syntax

`openflow# [no ]`
`openflow controller-id` *`id`* `ip`
`[ip-addr port` *`port-num`*`controller-interface-vlan` *`vlan-id`* `|`
*`oobm`*`]`

Configures the IPv4 or IPv6 connection between an instance and controller.

port

Specify the TCP port on which the controller can be reached.

Default port: 6633.

controller-interface-vlan

Specify the VLAN through which the controller connects.

oobm

Specify the oobm interface through which the controller connects.

[no]

Use the parameter [no] to remove an interface from a controller.

## Configure OpenFlow controller ports

An OpenFlow controller is configured globally under OpenFlow context and associated with an instance under instance context. See "Entering OpenFlow instance context" (page 16) for more information. OpenFlow controller traffic cannot be "in-band" or transit on a VLAN managed by OpenFlow and must transit on a VLAN not managed by OpenFlow.

OpenFlow controller traffic and OpenFlow traffic can transit on the same port, as long as they use different VLANs.

The VLAN chosen for OpenFlow controller traffic depends entirely on the IP address of the controller, and no specific configuration is needed. Thus the switch must have a proper IP configuration, and the controller address must be part of a subnet that is not on an OpenFlow VLAN.

For information on how to either manually assign an IP address to the switch or set it up to perform DHCP queries, see the *Configuring IP Addressing* chapter in the *Basic Operation Guide* for your HP switch.

Each OpenFlow instance can be controlled by up to three OpenFlow controllers and each generates OpenFlow commands and data flows between an OpenFlow switch and the controller.

### OpenFlow instance connection interruption mode

Use this to set switch behavior when the switch loses connection with the controller.

#### Syntax

`[no] openflow instance [`*`instance-name`*`] connection-interruption-mode { fail-secure | fail-standalone }`

`fail-secure`

If the switch loses connection with all controllers, packets and messages intended for the current controller are dropped. Flows continue to expire according to their time-outs.

Default: `fail-secure`

`fail-standalone`

If the switch loses connection with all controllers, packets and messages of new flows behave as a legacy switch or router would. Existing flows of this OpenFlow instance are removed.

## Associate OpenFlow instance with OpenFlow controller

Once the OpenFlow controller is set up, each instance must be associated to a controller.

### Syntax

`[no] openflow instance { instance-name | aggregate } controller-id controller-ID`

Up to three controllers can be specified per OpenFlow instance.

The `[no]` removes the identified controllers.

*`instance-name`*

Sets controller for the named instance.

`aggregate`

Sets controller for the aggregate instance.

*`controller-ID`*

OpenFlow controller ID to be associated with the instance; up to 3 controllers per instance.

**Example 1 Associating OpenFlow with multiple controllers**

To associate controllers 1, 5, and 100 to instance "test", use the following command:

```
HPswitch(config)# openflow instance test controller-id 1
#openflow instance test controller-id 5
#openflow instance test controller-id 100
```

**NOTE:** An OpenFlow controller is associated with an OpenFlow instance cannot be deleted.

## Setting maximum backoff interval for an instance

You can specify the maximum interval between two consecutive attempts to connect to a controller by an OpenFlow instance. The interval between two consecutive attempts increases exponentially until it reaches the specified value. All subsequent attempts use the specified value.

### Syntax

`openflow instance { instance-name | aggregate } max-backoff-interval secs`

*`instance-name`*

Sets the backoff interval for the named instance.

`aggregate`

Sets the backoff interval for the aggregate instance.

*`secs`*

Default: 60 seconds

Range: 1 — 3600 seconds

## Controller roles

Controller Roles is a mechanism which helps controllers synchronize handoff's in a scenario where multiple controllers are connected to the switch. A Controller is assigned one of the following roles:

- Equal
- Master
- Slave

Equal

This is the default role for a controller. The controller has full access to the switch and is equal to other controllers in the same role receiving all of the switch asynchronous messages (such as packet-in, flow-removed.) Controller-to-switch commands are sent and modified within this role.

Slave

A Slave controller only has the right to access to the switch in read-only mode. The controller cannot receive switch asynchronous messages except for Port-status. The controller is denied execution of the controller-to-switch commands: OFPT_PACKET_OUT, OFPT_FLOW_MOD, OFPT_GROUP_MOD, OFPT_PORT_MOD and OFPT_TABLE_MOD.

Master

The controller has full access to the switch. Only one controller can be the Master. When a controller role is changed to Master, the switch will automatically change all other controllers to Slave.

### Syntax

To query the switch:

```
show openflow instance [instance-name]
```

### Example

```
HP-3500yl-48G-PoEP(of-inst-t1)# show openflow instance t1
Configured OF Version    : 1.3
Negotiated OF Version: 1.3
Instance Name            : t1
Admin. Status            : Enabled
Member List              : VLAN 3
Listen Port              : None
Oper. Status             : Up
Oper. Status Reason      : NA
Datapath ID              : 0003b499ba86bf80
Mode                     : Active
Flow Location            : Hardware and Software
No. of Hw Flows          : 0
No. of Sw Flows          : 0
Hw. Rate Limit           : 0 kbps
Sw. Rate Limit           : 100 pps
Conn. Interrupt Mode     : Fail-Secure
Maximum Backoff Interval : 60 seconds
Probe Interval           : 10 seconds
Hw. Table Miss Count     : NA
No. of Sw Flow Tables    : 1
Egress Only Ports        : None
Table Model              : Policy Engine and Software

Controller Id  Connection Status  Connection State  Secure  Role
 ------------- -----------------  ----------------  ------  ------

 1              Connected           Active              No
 Slave
```

```
2                Connected            Active               No
Master
```

## Controller role change

When a controller's role is changed, the following messages can display:

OFPT_ROLE_REQUEST

Message from controller to change or query its role.

OFPT_ROLE_REPLY

Message sent in response to the OFPT_ROLE_REQUEST,  it returns the current Roleof the controller.

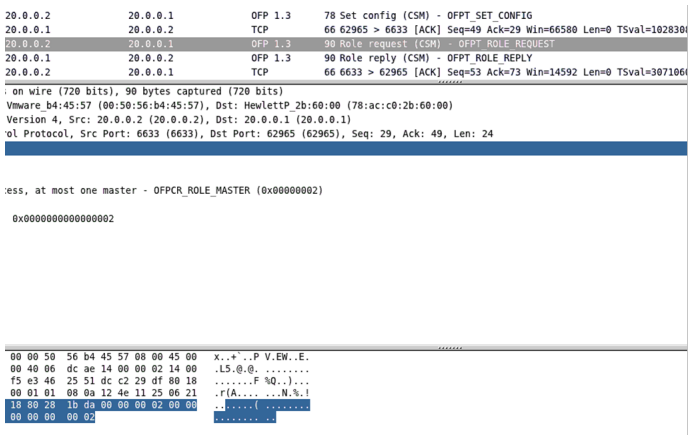
OFPT_SET_ASYNC

A controller, through this message can configure what asynchronous message it wants to receive.

OFPT_GET_ASYNC

Controller uses this message to retrieve the asynchronous configuration set using the OFPT_SET_ASYNC message.

**Figure 5 OFPT_ROLE_REQUEST message**




**NOTE:** On failover/connection interruption, once connection is reestablished, each controller connection is set as OFPCR_ROLE_EQUAL, which controller can query and change if required

# Port modification

Port modification is used to change the characteristics of a port for an instance on the switch via the controller. The controller sends an OFP_PORT_MOD message to the switch which instructs the port on how to modify and change chracteristics. See "Port configuration" (page 35) and "Port error messages" (page 62).

The following command checks the state of the highlighted bits for the current values of the port configuration for all ports of an instance.

## Syntax

```
show openflow instance t1 port-statistics
```

## Example

```
HP-Stack-3800(of-inst-t1)# show openflowinstance t1 port-statistics
Number of Ports :1
Port 1/1: Up
```

```
Status
Admin. Status: Enabled   Flood: Enabled
Receive: Enabled      Forward: Enabled
Packet_in: Disabled
Statistics
  Collisions       : 0
  Rx Packets       : 0                        TxPackets   : 47
   Rx Bytes         : 0                         TxBytes     : 10718
  Rx Dropped       : 0                        TxDropped   : 0
   Rx Errors        : 0                         TxErrors    : 0
   Frame Errors     : 0
  CRC Errors       : 0
  Overrun Errors   : 0
```

## Example v1.0

```
Wireshark Capture of a sample Port-Mod message for a 1.0 instance

OpenFlow Protocol
  Header
   Version: 0x01
   Type: Port Mod (CSM) (15)
   Length: 32
   Transaction ID: 4
  Port Modification
   Port #: 5
    MAC Address: HewlettP_02:2c:bb (84:34:97:02:2c:bb)
Port ConfigFlags
.... .... .... .... .... .... .... ...0 = Port is administratively down: No (0)
.... .... .... .... .... .... .... ..0. = Disable 802.1D spanning tree on port: No (0)
.... .... .... .... .... .... .... .0.. = Drop non-802.1D packets received on port: No (0)
.... .... .... .... .... .... .... 0... = Drop received 802.1D STP packets: No (0)
.... .... .... .... .... .... ...1 .... = Do not include this port when flooding: Yes (1)
.... .... .... .... .... .... ..0. .... = Drop packets forwarded to port: No (0)
.... .... .... .... .... .... .0.. .... = Do not send packet-in msgs for port: No (0)
Port Config Mask
.... .... .... .... .... .... .... ...0 = Port is administratively down: No (0)
.... .... .... .... .... .... .... ..0. = Disable 802.1D spanning tree on port: No (0)
.... .... .... .... .... .... .... .0.. = Drop non-802.1D packets received on port: No (0)
.... .... .... .... .... .... .... 0... = Drop received 802.1D STP packets: No (0)
.... .... .... .... .... .... ...1 .... = Do not include this port when flooding: Yes (1)
.... .... .... .... .... .... ..0. .... = Drop packets forwarded to port: No (0)
.... .... .... .... .... .... .0.. .... = Do not send packet-in msgs for port: No (0)
Port Advertise Flags
.... .... .... .... .... .... .... ...0 = 10 Mb half-duplex rate support: No (0)
.... .... .... .... .... .... .... ..0. = 10 Mb full-duplex rate support: No (0)
.... .... .... .... .... .... .... .0.. = 100 Mb half-duplex rate support: No (0)
.... .... .... .... .... .... .... 0... = 100 Mb full-duplex rate support: No (0)
... .... .... .... .... .... ...0 .... = 1 Gb half-duplex rate support: No (0)
.... .... .... .... .... .... ..0. .... = 1 Gb full-duplex rate support: No (0)
.... .... .... .... .... .... .0.. .... = 10 Gb full-duplex rate support: No (0)
.... .... .... .... .... .... 0... .... = Copper medium support: No (0)
.... .... .... .... .... ...0 .... .... = Fiber medium support: No (0)
.... .... .... .... .... ..0. .... .... = Auto-negotiation support: No (0)
 .... .... .... .... .... .0.. .... .... = Pause support: No (0)
.... .... .... .... .... 0... .... .... = Asymmetric pause support: No (0)
Pad: 0
Pad: 0
Pad: 0
Pad: 0
```

## Example v1.3

```
Wireshark Capture of a sample Port-Mod message for a 1.3 instance
OpenFlow Protocol
  Header
   Version: 0x04
   Type: Port Mod (CSM) (16)
    Length: 40
   Transaction ID: 4043243760
   Port Modification
   Port #: 2
   Pad: 0
    Pad: 0
    Pad: 0
```

```
    Pad: 0
  MAC Address: HewlettP_02:2c:be (84:34:97:02:2c:be)
    Pad: 0
   Pad: 0
   Port ConfigFlags
.... .... .... .... .... .... .... ...0 = Port is administratively down: No (0)
.... .... .... .... .... .... .... ..0. = Disable 802.1D spanning tree on port: No (0)
.... .... .... .... .... .... .... .0.. = Drop non-802.1D packets received on port: No (0)
.... .... .... .... .... .... .... 0... = Drop received 802.1D STP packets: No (0)
.... .... .... .... .... .... ...0 .... = Do not include this port when flooding: No (0)
.... .... .... .... .... .... ..0. .... = Drop packets forwarded to port: No (0)
.... .... .... .... .... .... .1.. .... = Do not send packet-in msgs for port: Yes (1)
   Port Config Mask
.... .... .... .... .... .... .... ...0 = Port is administratively down: No (0)
.... .... .... .... .... .... .... ..0. = Disable 802.1D spanning tree on port: No (0)
.... .... .... .... .... .... .... .0.. = Drop non-802.1D packets received on port: No (0)
.... .... .... .... .... .... .... 0... = Drop received 802.1D STP packets: No (0)
.... .... .... .... .... .... ...0 .... = Do not include this port when flooding: No (0)
.... .... .... .... .... .... ..0. .... = Drop packets forwarded to port: No (0)
.... .... .... .... .... .... .1.. .... = Do not send packet-in msgs for port: Yes (1)
  Port Advertise Flags
.... .... .... .... .... .... .... ...0 = 10 Mb half-duplex rate support: No (0)
.... .... .... .... .... .... .... ..0. = 10 Mb full-duplex rate support: No (0)
.... .... .... .... .... .... .... .0.. = 100 Mb half-duplex rate support: No (0)
.... .... .... .... .... .... .... 0... = 100 Mb full-duplex rate support: No (0)
.... .... .... .... .... .... ...0 .... = 1 Gb half-duplex rate support: No (0)
.... .... .... .... .... .... ..0. .... = 1 Gb full-duplex rate support: No (0)
.... .... .... .... .... .... .0.. .... = 10 Gb full-duplex rate support: No (0)
.... .... .... .... .... .... 0... .... = Copper medium support: No (0)
.... .... .... .... .... ...0 .... .... = Fiber medium support: No (0)
.... .... .... .... .... ..0. .... .... = Auto-negotiation support: No (0)
.... .... .... .... .... .0.. .... .... = Pause support: No (0)
.... .... .... .... .... 0... .... .... = Asymmetric pause support: No (0)
   Pad: 0
   Pad: 0
  Pad: 0
  Pad: 0
```

### Example

Send a Port-Mod command to the switch using dpctl utility.

```
root@openflow-ubuntu-10:/home/openflow# dpctltcp:10.20.30.50:6633 port-desc

... {no="6", hw_addr="00:1b:3f:cf:76:fa", name="A6", config="0x0", state="0x1", curr="0x0", adv="0x0", supp="0x0",
 peer="0x0", curr_spd="100000000kbps", max_spd="100000000kbps"} ...

root@openflow-ubuntu-10:/home/openflow# dpctltcp:10.20.30.50:6633 port-mod
port=6,addr=00:1b:3f:cf:76:fa,conf=0x40,mask=0x40

SENDING:
port_mod{port="6", hwaddr="00:1b:3f:cf:76:fa", config="0x00000040", mask="0x40", adv="0x0"}
OK

root@openflow-ubuntu-10:/home/openflow# dpctltcp:10.20.30.50:6633 port-desc
... {no="6", hw_addr="00:1b:3f:cf:76:fa", name="A6", config="0x40", state="0x1", curr="0x0", adv="0x0", supp="0x0",
 peer="0x0", curr_spd="100000000kbps", max_spd="100000000kbps"} ...
```

## Default Port Configurations

The configuration of an egress-only ports for an instance have the OFPPC_NO_FLOOD and OFPPC_NO_PACKET_IN bits set as the default. Packets matching flood action are not sent from these ports and any flow miss of incoming packets on this port will not produce a “Packet-In” message to the controller. The configuration of a member port for an instance by default will not have any of the bits set in the configuration field (configfield) of OFP_PORTSTRUCTURE.

## OpenFlow Version information

Port Modification of OFP_PORT_CONFIG bit OFPPC_NO_FLOOD and OFPPC_NO_PACKET_IN are allowed via the Port Mod message in Openflow 1.0. Modification ofOFP_PORT_CONFIG bit OFPPC_NO_PACKET_IN via the Port Mod message are allowed in OpenFlow v1.3.

### Not Allowed

- Only OFPPC_NO_FLOOD and OFPPC_NO_PACKET_IN bits modifications are supported by the controller.
- The Port Mod message is rejected if modification of OFP_PORT_CONFIG is requested.
- When the Egress-Only Mode for Openflow is enabled on the switch the Port-Mod message for an egress-only port (for that instance) will produce an error message.
- If a port is not exclusive to the Openflow Member VLAN, a Port Mod error message will display.
- Packets that match a flood action rule can not be sent out of global egress-only ports. This is true even if the port is a member of the Openflow instance and is part of the global egress-only pool. (Both v1.0 and v1.3 instances).

## Pre-Provisioning

Pre-provisioning in OpenFlow sends a port status change message to controller instead of deleting the down or blocked ports. The down or blocked ports are not deleted from an OpenFlow instance but messaged with an appropriate enumset (LINK_DOWN/BLOCKED/ LIVE.) This enables the controller to see 'down / blocked' ports and it can pre-provision flows which deals with ports that are down.

OpenFlow v1.3 provides following enums for port state notification as part of OFPT_PORT_STATUS message:

- OFPPS_LINK_DOWN - No physical link present
- OFPPS_BLOCKED - Port is blocked
- OFPPS_LIVE - Live for Fast Failover Group

OpenFlow v1.0 works differently. Whenever a port goes down or gets blocked, controller receives a message—PORT_DELETE.

### Example

```
HP-Stack-3800(openflow)# show vlan 3
Status and Counters - VLAN Information - VLAN 3
VLAN ID : 3
Name : VLAN3
Status : Port-based
Voice : No
Jumbo : No

Port Information  Mode       Unknown VLAN  Status
----------------  --------  ------------  ----------
1/1                  Untagged  Learn           Up
1/2                  Untagged  Learn           Up
1/4                 Untagged  Learn          Down
1/5                  Untagged  Learn           Down
```

```
openflow@openflow-ubuntu-02:~$ dpctl tcp:20.0.0.1:6633 port-desc
SENDING:
stat_req{type="port-desc", flags="0x0"}

RECEIVED:
stat_repl{type="port-desc", flags="0x0"{no="4", hw_addr="08:2e:5f:69:6e:7c", name="1/4", config="0x0", state="0x1",
 curr="0x0", adv="0x0", supp="0x0", peer="0x0", curr_spd="0kbps", max_spd="0kbps"}, {no="2",
hw_addr="08:2e:5f:69:6e:7e", name="1/2", config="0x0", state="0x4", curr="0x220", adv="0x0", supp="0x22f",
peer="0x0", curr_spd="3567587328kbps", max_spd="3567587328kbps"}, {no="5", hw_addr="08:2e:5f:69:6e:7b", name="1/5",
 config="0x0", state="0x1", curr="0x0", adv="0x0", supp="0x0", peer="0x0", curr_spd="0kbps", max_spd="0kbps"},
 {no="local", hw_addr="08:2e:5f:69:6e:65", name="local", config="0x0", state="0x0", curr="0x0", adv="0x0",
supp="0x0", peer="0x0", curr_spd="0kbps", max_spd="0kbps"}, {no="1", hw_addr="08:2e:5f:69:6e:7f", name="1/1",
config="0x0", state="0x4", curr="0x220", adv="0x0", supp="0x22f", peer="0x0", curr_spd="3567587328kbps",
max_spd="3567587328kbps"}}}
```

# Configuring egress only ports

This CLI command enables or disables support for advertising egress-only ports to the controller. Ports that are members of non-OpenFlow VLANs are egress-only ports. A controller can add a flow with an egress-only port as an output port to allow traffic to be forwarded from an OpenFlow instance to a non-OpenFlow VLAN. All instance member ports and egress-only ports are exposed as instance ports to the controller.

## Syntax

```
Openflow # egress-only-ports
```

egress-only-ports

Enable or disable support for advertising egress-only ports to the controller.

Ports that are members of non-OpenFlow VLANs are egress-only ports. A controller can add a flow with an egress-only port as an output port to enable traffic to be forwarded from an OpenFlow instance to a non-OpenFlow VLAN.

**NOTE:** Egress-only ports cannot be used as an "in-port" in any flow by a controller. If this is attempted an error message will be displayed similar to `FLOW_MOD_FAILED`.

## Example

```
Openflow # egress-only-ports
Configured OF Version: 1.0
Negotiated OF Version: 1.0
Instance Name: test
Admin. Status: Enabled
Member List: VLAN 3
Listen Port: None
Oper. Status: Up\
Oper. Status Reason: NA
DatapathID: 00032c4138c98500
Mode: Active
Flow Location: Hardware and Software
No. of Hw Flows: 0
No. of Sw Flows: 0
Hw. Rate Limit: 0 kbps
Sw. Rate Limit: 100 pps
Conn. Interrupt Mode: Fail-Secure
 Maximum BackoffInterval : 60 seconds
Probe Interval: 10 seconds
Hw. Table Miss Count: 0
No. of Sw Flow Tables: NA
Egress Only Ports: A1,A3-A24,F1-F22

Table Model: Single Table

Controller id  Connection Status  Connection State  Secure  Role
-------------- -----------------  ----------------  ------  ------
1      Disconnected    Backoff     No   Equal
```

# Enable or disable support for advertising egress-only ports to the controller

## Syntax

```
openflow-instance-name # egress-only-ports
[no] egress-only-ports
```

# Software and hardware rate limiting

You can specify resource limits used by an OpenFlow instance. Each OpenFlow instance has completely independent rate-limiters that can be set separately.

## Syntax

```
openflow instance { instance-name | aggregate }
limit { hardware-rate kbps | software-rate pps }
```

*instance-name*

Set software and hardware rate limiting for the named instance.

aggregate

Set software and hardware rate limiting for the aggregate instance.

*kbps*

Limit the bandwidth that can be utilized by an OpenFlow instance.

Default: 0 kbps
Range 0 — 10,000,000 kbps

*pps*

Configure the OpenFlow instance packet rate limit.

Limits the number of packets per second per module that this instance can send to the software path.

Default: 100 pps
Range: 1 — 10,000 pps

**NOTE:** Increasing the software rate limit increases CPU consumption and may impact system performance.

If the software rate limit is specified beyond 1000 pps the warning below appears:

```
Increasing the software rate limit would increase CPU
consumption and may impact the system performance.
```

# Configuring listener ports

Configures an OpenFlow port to listen for incoming connections from an OpenFlow controller.

## Syntax

```
[no] openflow instance { instance-name | aggregate }
listen-port [tcp-port] [oobm]
```

*instance-name*

Sets the `listen—port` for the named instance.

aggregate

Sets the `listen—port` for the aggregate instance.

*tcp-port*

Specify the port to listen on.

Default: Port number 6633

Range: Port number 1024 - 65536

oobm

Configure to listen through the OOBM port. Only applicable for switches that have a separate out-of-band management (OOBM) port.

# Configuring IP control table

## Syntax

```
OpenFlow # limit policy-engine-usage | ip-ctrl-table-usuage
| multiport-filter-usage max-percentage
```

policy-engine-usage

Maximum percentage of policy engine resources used by OpenFlow.

ip-cntrl-table-uage

Maximum percentage of IP control table resources used by OpenFlow.

multiport-filter-usage

Maximum percentage of the multiport-filer resources used by OpenFlow.

You can limit the OpenFlow percentage of policy engine resources, ip control table usage and multiport filter usage so that it does not impact other functions that use the same resources.

The limit can only be set when OpenFlow is disabled globally.

*Max-Percent*

Specifying 0% allocates resources for OpenFlow.

By default, the OpenFlow policy engine resource usage is set at 50% to avoid oversubscribing resources and impacting performance. In addition to OpenFlow, the policy engine resource can be used by Access Control Lists, Quality of Service, Identity Driven Management, Virus Throttling, Mirroring, Policy Based Routing, and other features.

**NOTE:** The maximum percentage is not a guaranteed percentage but a maximum allowed limit.

Using the default 50% resource usage setting, the 8200zl and 5400zl switches with v1 zl modules, and the 3500/3500yl, 6200yl, and 6600 switches can support approximately 1000 OpenFlow rules in hardware while the 8200zl and 5400zl switches with v2 zl modules and the 3800 switches can support up to 1000 OpenFlow rules in hardware. Additional rules beyond the hardware limit are processed by software.

To increase the number of flows beyond the default 50% setting, use the above OpenFlow limit policy-engine-usage command. Since several features, including OpenFlow, ACL, and QoS, also can use policy engine resources, ensure that any new setting you configure does not exceed total policy engine resources. For example, if all policy engine resources are in use, OpenFlow rules will no longer be added in hardware and the switch will deny attempts to configure ACLs with the CLI. To determine resource usage on your switch, see "Viewing OpenFlow resources" (page 46) and the appendix titled "Monitoring Resources" in the latest *Management and Configuration Guide* for your switch.

Default: 50%

Range: 0 — 100%

**NOTE:** Resource usage can only be set when OpenFlow is disabled.

## Example

```
openflow# limit multiport-filter-usage [1-100]
0-100:  Maximum percentage of Multiport filters used by OpenFlow.

HP-3500yl-24G-PoEP# show openflow multiport-filter-limit
```

```
Total Multiport Filters: 2037
                  Filters       Filters        Filters
Features       Allocated    Used            Free
------------ ---------- -----------  -----------
OpenFlow       1024             0                1024
```

## Hardware statistics refresh rate

### Syntax

```
openflow-instance-name [#]hardware statistics|refresh rate
policy-engine-table <value>
```

Refresh rate for policy engine table statistics.

Default: 0–3600

## Backing up your configuration (optional)

Dual management module in E8200 platforms is supported. Flow configuration is synchronized across management modules and flow table preserved during switchover.

See *Chassis Redundancy* in the *Management and Configuration Guide* for your switch.

## Configuring OpenFlow VLANs

Configure all OpenFlow VLANs and production VLANs; verify reachability.

**NOTE:** For information on configuring and verifying VLANs, see the *Advanced Traffic Management Guide* for your switch.

## Configuring and verifying routing

For information on configuring and verifying routing, see the *Multicast and Routing Guide* for your switch.

## Configuring physical and logical ports

For information on configuring and verifying ports, see the *Management and Configuration Guide* for your switch.

## Configuring OpenFlow

OpenFlow traffic can be separated from the rest of the traffic on the network per VLAN, so that non-OpenFlow traffic will not be impacted by OpenFlow.

**NOTE:** If multiple commands to the same TCP port are received from multiple controllers, the last command takes priority.

**OpenFlow can be configured for Virtualization or Aggregation mode.**

Virtualization mode

Each OpenFlow instance is independent and has its own OpenFlow configuration and OpenFlow controller connection. Some VLANs are designated as members of OpenFlow instances while other VLANs are not. VLANs that are not members of OpenFlow instances could be thought of as VLANs carrying production traffic.

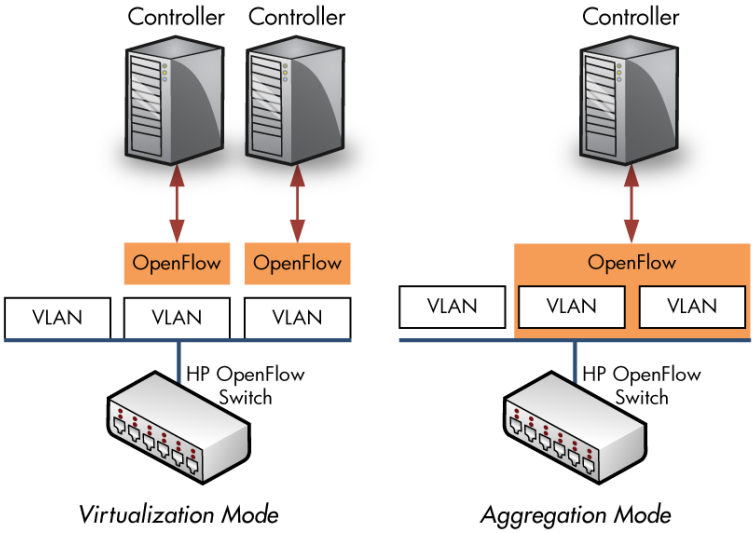
Aggregation mode

Provides a single OpenFlow instance that includes all of VLANs configured on the switch except the VLAN(s) that connect to the controller(s) and the management VLAN on the switch.

Production traffic is not allowed.

**NOTE:** It is not possible to mix aggregation and virtualization modes of operation.

**Figure 6 Supported VLAN modes**




**NOTE:** In Aggregation Mode, OpenFlow manages all VLANs except controller VLAN or MGMT VLAN.

## Virtualization mode

With Virtualization mode, some VLANs can be designated as members of OpenFlow instances. Each OpenFlow instance is independent and has its own OpenFlow configuration and OpenFlow controller connection. Each OpenFlow instance can have one member VLAN.

Normal operation requires the presence of at least one VLAN not managed by OpenFlow. The non-OpenFlow VLANs are used to run the OpenFlow controller connections. Non-OpenFlow VLAN(s) can also be used for any traffic that is not supposed to be managed by OpenFlow, referred to as production traffic.

Typical networks use VLAN 1 as the management VLAN. A management VLAN cannot be a member of an OpenFlow instance. A management VLAN can be configured as a controller VLAN.

VLANs are not shared among OpenFlow instances. In addition to these, another VLAN is designated as the OpenFlow controller VLAN through which the switch communicates via the OpenFlow protocol to the remote controller.

**NOTE:** The OpenFlow controller VLAN also could be the default VLAN. The traffic that belongs to one OpenFlow instance must be contained within that instance and must not propagate to other OpenFlow instances or to production VLANs.

Specifying a valid VLAN not yet configured as a member VLAN of an OpenFlow instance is not an error, but when that VLAN is configured OpenFlow will start working on that OpenFlow instance.

### Port configuration

Switch ports are assigned to VLANs (see "Virtualization mode" (page 35)), but ports cannot be assigned directly to an OpenFlow instance. A port can be part of multiple VLANs when using tagged mode, so a port can be part of multiple OpenFlow instances and non-OpenFlow VLANs.

For more information about configuring interfaces (ports), see the "Port Status and Configuration" chapter in the *Management and Configuration Guide* for your HP switch.

### Example 2 Displaying the status of all ports

```
HP Switch(config)# show interfaces

Status and Counters - Port Status

                  | Intrusion                          MDI  Flow Bcast
Port Type         | Alert Enabled Status Mode          Mode Ctrl Limit
---- ---------- + ----- ------- ------ ------------ ---- ---- -----
1    100/1000T  | No    Yes     Down   Auto-10-100  Auto off  0
2    100/1000T  | No    Yes     Down   1000FDx      Auto off  0
3    100/1000T  | No    Yes     Down   1000FDx      Auto off  0
4    100/1000T  | No    Yes     Down   1000FDx      Auto off  0
5    100/1000T  | No    Yes     Down   1000FDx      Auto off  0
6    100/1000T  | No    Yes     Down   1000FDx      Auto off  0
```

### Example 3 Displaying configuration of all ports

```
HP Switch(config)# show interfaces config

 Port Settings

  Port  Type       | Enabled Mode          Flow Ctrl MDI
  ----- ---------- + ------- ------------- --------- ----
  1     100/1000T  | Yes     Auto-10-100   Disable   Auto
  2     100/1000T  | Yes     Auto          Disable   Auto
  3     100/1000T  | Yes     Auto          Disable   Auto
  4     100/1000T  | Yes     Auto          Disable   Auto
  5     100/1000T  | Yes     Auto          Disable   Auto
  6     100/1000T  | Yes     Auto          Disable   Auto
```

### Example 4 Displaying port statistics counters

```
HP Switch(config)# show interfaces brief

Status and Counters - Port Status

                  | Intrusion                          MDI  Flow Bcast
 Port Type        | Alert     Enabled Status Mode      Mode Ctrl Limit
 ---- ------- + --------- ------- ------ ------- ---- ---- -----
 1    10GbE-T  | No        Yes     Up     1000FDx MDIX off  0
 2    10GbE-T  | No        Yes     Down   10GigFD MDI  off  0
 3    10GbE-T  | No        Yes     Down   10GigFD MDIX off  0
 4    10GbE-T  | No        Yes     Down   10GigFD Auto off  0
 5    10GbE-T  | No        Yes     Down   10GigFD Auto off  0
 6    10GbE-T  | No        Yes     Down   10GigFD Auto off  0
 7    10GbE-T  | No        Yes     Down   10GigFD Auto off  0
 8    10GbE-T  | No        Yes     Down   10GigFD Auto off  0
```

### Example 5 Displaying VLANs associated with a port

```
HP Switch(config)# show vlans ports 1-3

 Status and Counters - VLAN Information - for ports 1-3

  VLAN ID Name                          Status       Voice Jumbo
  ------- ----------------------------- ------------ ----- -----
  1       DEFAULT_VLAN                  Port-based   No    Yes
  10      VLAN10                        Port-based   No    No
  15      VLAN15                        Port-based   No    No
```

## OpenFlow configuration

Configure OpenFlow with the switch CLI. To change a configuration, you must be in the configuration

context using the `config` command. Some commands can be executed in OpenFlow or in the instance context, see "Entering OpenFlow instance context" (page 16).

There can be multiple OpenFlow instances, and each instance is bound to a specific VLAN. The configuration of OpenFlow is done on a per-instance basis.

**NOTE:** Ensure that each VLAN used in OpenFlow commands below is already defined. See "Virtualization mode" (page 35) for information on how to create a VLAN or check currently defined VLANs.

## Aggregation mode

Aggregation mode enables the switch to have only a single aggregated OpenFlow instance managing all VLANs within the switch. It impacts how OpenFlow is managed inside the switch.

In aggregation mode, all VLANs on the switch are managed by a single OpenFlow instance. Because OpenFlow control traffic cannot be in-band, specific OpenFlow controller VLANs such as VLANs over which the controllers may be reached are excluded from aggregation. The management VLAN, if defined on the switch, is also excluded from aggregation. In aggregation mode, the OpenFlow instance manages all VLANs except the OpenFlow controller VLANs and management VLAN.

# 3 Group table

Groups represent sets of actions for flooding as well as more complex forwarding semantics (e.g. multipath, fast reroute, and link aggregation). As a general layer of indirection, groups also enable multiple flow entries to forward to a single identifier (e.g. IP forwarding to a common next hop). This abstraction allows common output actions across flow entries to be changed efficiently.

The group table contains group entries; each group entry contains a list of action buckets with specific semantics dependent on group type. The actions in one or more action buckets are applied to packets sent to the group. There are 4 types of groups:

1. All

   All the action buckets in the group should be executed when a packet hits the group table.

2. Select

   Execute any one action bucket in the group. The switch implementation uses round-robin to select the action bucket to be executed. Openflow specification defines a weight mechanism to do load sharing. However, this is not supported in the switch implementation. The weight MUST be given as 1. For all the other groups, weight MUST be specified as 0.

3. Indirect

   Execute the one defined bucket in this group. This group supports only a single bucket.

4. Fast failover

   Execute the first live bucket. The buckets are evaluated for liveness in the order defined by the group.

For the implementation of groups, the following is important to note:

- Group table is supported only in software. Hence, group cannot be referenced directly from a hardware flow entry.
- The number of groups per instance is capped to 32.
- The total number of groups in the switch is capped to 1024.
- Chaining of groups not supported. Because of this, watch_group is also not supported while doing group additions. Watch_group MUST always be set to OFPG_ANY for all the group types.

# 4 OpenFlow per-flow rate limiting

OpenFlow supports per-flow rate-limiters as HP vendor extensions.

A rate-limiter controls the rate of packets passing through a switch. Per-flow rate-limiters associate an arbitrary number of flows with a rate-limiter. Using OpenFlow with per flow rate-limiters, any number of flows can be flexibly mapped to a rate-limiter, regardless of their source and destination ports. The use of rate-limiters requires a version of ovs-ofctl which includes HP QoS extension. Rate-limiters are addressed by a limiter_id, an arbitrary 32 bit number. Configuration of rate-limiters is done through a simple message from the OpenFlow controller which can add, modify or remove a rate-limiter. Flows are directed to rate-limiters through an action. Multiple flows can be associated with the same rate-limiter. Statistics can be read from the OpenFlow controller for each rate-limiter.

**NOTE:** Per-flow rate-limiters are used only if the hardware rate-limiter for the instance is disabled.

## QoS extensions

HP QoS extension to the OpenFlow protocol provides support for rate-limiters. A rate-limiter controls the rate of packets passed through it. Per-flow rate-limiters associate an arbitrary number of flows with a rate-limiter. The HP QoS vendor extensions support per-flow rate-limiters with only drop rate flag and not mark rate or other flags.

## Maintain limiter in rule

The rule structure maintains the limiter identification and a flag to indicate if the rule has a limiter ID associated with it.

## Create a limiter

A per-flow rate-limiter is added/created from the OpenFlow controller using the add-limiter command. Note that this requires the controller to have the HP QoS extensions.

On receiving a vendor request from the OpenFlow controller, the vendor ID is checked for HPX_VENDOR_ID and then passed on to “ofputil_decode_hpx” to be decoded. On receiving a message of type OFPUTIL_HPX_ADD_LIMITER, a new meter is created with the parameters received in the message. The meters are created, updated and deleted by calls to platform independent functions, which in turn call the platform dependent functions.

## Get limiter details

The details on the limiters configured can be retrieved by issuing a `dump-limiters` command from the OpenFlow controller. These details can also be checked on the switch using `show openflow <inst_name> limiters`.

## Support flow with a limiter

A flow can be associated with a per-flow rate-limiter by giving the limiter ID in actions. For example, assume that a per flow rate-limiter with ID 100 is created with an `add-limiter` command from the OpenFlow controller. From an ovs controller with HP QoS extensions, a flow can be associated with this rate-limiter using the **rate_limit** key word in the actions as indicated below.

```
ovs-ofctl add-flow tcp:192.168.1.2:6633 idle_timeout=0,ip,nw_src=20.20.20.41,action=output:57,rate_limit:100
```

## Measures to keep OpenFlow in check

Rate limiting traffic may be required on OpenFlow enabled VLANs to control OpenFlow experiments and to prevent them from consuming significant switch resources. OpenFlow implementation has two rate-limiters for each instance, which have different purposes. Each instance has completely independent rate-limiters that can be set independently. OpenFlow supports these rate-limiters:

Software rate-limiter

Limits the number of packets that each line card can send to the software path on the CPU (i.e. packets which match with the default OpenFlow TCAM rule and are punted to master CPU.) Thus it indirectly controls the number of packets sent by the switch to the controller.This rate-limiter is enabled by default (default value =100.)

Hardware rate-limiter

Controls the bandwidth for an OpenFlow instance on module;

Disabled by default.

### Syntax

```
instance-name# limit [tab]
```

Limits the bandwidth per module in kilobits per second that an OpenFlow instance can utilize.

### Syntax

```
instance-name# limit
```

Limits the number of packets per second for each module that an OpenFlow instance can send to the software path on the CPU.

### Syntax

```
instance-name# limit hardware-rate
```

Limits the bandwidth per module in kilobits per second that this OpenFlow instance can utilize. Minimum rate is 0 kbps. Maximum rate is 10000000 kbps.

### Syntax

```
instance-name# limit software-rate
```

Limits the number of packets per second for each module, that this OpenFlow instance can send to the software path on the CPU.Minimum rate is 1 pps. Maximum rate is 10000 pps. Default rate is 100 pps.

# 5 Administering OpenFlow

Additional fields and filters added in OpenFlow version 1.3 increases the available show commands.

## Monitoring OpenFlow

OpenFlow can be monitored at several levels and the rate at which the information from the hardware is refreshed can be configured.

## Displaying OpenFlow information

Displays the versions of OpenFlow instance with status and flow data.

### Syntax

```
show openflow
```

**Example 6 Show OpenFlow**

```
HP-Switch# show openflow

OpenFlow               : Disabled
IP Control Table Mode  : Enabled

Instance Information
                                    No. of     No. of     OpenFlow
  Instance Name       Status        H/W Flows  S/W Flows  Version
  ------------------- ------------- ---------- ---------  ---------
  titan               Down             0          0      1.0
  marez               Up               0          0      1.3 only
```

## Setting OpenFlow statistics refresh rate

Choose the maximum time before hardware statistics are refreshed.

### Syntax

```
openflow hardware-statistics refresh-rate secs
```

*secs*

The hardware statistics refresh-rate for OpenFlow.

Default: 20 seconds
Range: 0— 3600 seconds

**NOTE:** With value of 0, the hardware is no longer polled to update the statistics.

## Viewing OpenFlow information

You can display OpenFlow information for all instances, ports, and flows. The returned information includes the OpenFlow version supported.

### Syntax

```
show openflow [ resources | controllers | instance instance-name [[
port-statistics ] | flows [ flow-filters auxiliary-connections] ] ]
```

Show OpenFlow information.

`resources`

Shows OpenFlow resource utilization. See "Viewing OpenFlow resources" (page 46).

`controllers`

Shows controllers configured for OpenFlow. See "Viewing OpenFlow controllers" (page 47)

`instance-name`

Instance information can be obtained for ports or flows. See "Viewing OpenFlow instance attributes" (page 47) for more information.

`port-statistics`

Shows port statistics.

`flows`

*`flow-type`*

Shows the flow table entries for a particular OpenFlow instance. The various flows displayed using *`flow-type`* are shown in the next two examples (???:

## Example of flow version 1.0

```
(<openflow>)# show openflow instance titan flows
Flow 1 Match  Incoming Port  :F24       Ethernet Type      : IP
       Source MAC    : 000000-000000  Destination MAC   :
000000-000000
       VLAN ID    :0            VLAN Priority     : 0
       Source Protocol Address: 255.255.255.255/32
          Target Protocol Address: 128.128.128.128/32
       IP Protocol   : 0x00       IP ToS Bits      : 0
       Source Port    : 0        Destination Port : 0
   Attributes
       Priority     : 32768      Duration      : 10  secs
        Hard Timeout   : 0 secs      Idle Timeout : 60 secs
        Byte Count    : 0        Packet Count : 0
        Controller ID  : 1         Cookie : 0x0
       Flow Location   : Software    Hardware Index : 1
       Reason Code    : 100
        Reason Description: Rule is in hardware
    Actions
        Modify Destination IP   : 183.23.45.64
       Modify Source IP      : 200.123.23.54
       Output        : A21
```

## Example of flow version 1.3

```
(<openflow>)# show openflow instance titan flows
Flow 1
   Match
       Incoming Port   : 1/17       Ethernet Type       : IP
       Source MAC   : 000000-000000    Destination MAC    : 000000-000000
       VLAN ID     : 0         VLAN Priority     : 0
       Source Protocol Address: 255.255.255.255/32
          Target Protocol Address: 128.128.128.128/32
       IP Protocol   : TCP          IP ToS Bits     : 0
       IP ECN     : 0           IP DSCP       : 18
       Source Port   : 0            Destination Port    : 0
   Attributes
       Priority      : 32768     Duration      : 10  secs
        Hard Timeout    : 0 secs       Idle Timeout     : 0 secs
        Byte Count     : 5040        Packet Count     : 28
       Flow Table ID   : 3        Controller ID    : 1
       Activity Count   : 0xffffffff     Cookie      : 0x0
       Hardware Index   : 1
   Instructions
    Clear Actions
```

```
    Write Actions   Pop VLAN
           Push VLAN
           Decrement TTL
           Output: : 3/24, 4/5, 1/18
           Goto Table ID: 2
Flow 2
   Match
        Incoming Port   : Trk1       Ethernet Type      : IPv6
        Source MAC    : 000000-000000   Destination MAC    : 000000-000000
        VLAN ID      : 0        VLAN Priority    : 0
        Source Protocol Address: 255.255.255.255/32
           Target Protocol Address: 128.128.128.128/32
        IPv6 Flow Label   : 0
        IPv6 Ext. Header  : Fragment
        ND Source MAC    : 000000-000000     ND Destination MAC    : 000000-000000

        ND Target IP   : 0:0:0:0:0:0:0:0
        IP Protocol    : 0x2C
        IP ECN      : 0           IP DSCP       : 20
        Source Port    : 0         Destination Port   : 0
  Attributes
        Priority     : 12345        Duration       : 10 secs
        Hard Timeout   : 300 secs        Idle Timeout    : 160 secs
        Byte Count     : 0          Packet Count     : 0
        Flow Table ID    : 6         Controller ID    : 1
        Activity Count   : 0xffffffff      Cookie        : 0x0
        Hardware Index : 1
  Instructions
   Apply Actions
        Modify Destination IP     : 2000::5
        Modify Source IP      : 2000::6
        Modify Source MAC      : 121212-121212
        Modify Destination MAC     :131313-131313
        Modify VLAN ID       : 123
        Modify IP DSCP       : 18
        Modify IP ECN        : 1
        Decrement TTL
        Meter ID          : 112
        Group ID         : 2
   Write Actions
        Decrement TTL
        Goto Table ID        : 4
Flow 3
  Match
        Incoming Port    : 0       Ethernet Type   : ARP
        Source MAC     : 000000-000000          Destination MAC   : 000000-000000
        VLAN ID     : 0        VLAN Priority   : 0
        ARP Opcode    : 1
        ARP Source MAC   : 00A0C9-22B210  ARP Target MAC   : 000000-000000
        Source Protocol Address: 255.255.255.255/32
           Target Protocol Address: 128.128.128.128/32
  Attributes
        Priority      : 32768       Duration      : 10 secs
        Hard Timeout    : 0 secs       Idle Timeout    : 0 secs
        Byte Count     : 12450       Packet Count   : 2323
        Flow Table ID    : 3        Controller ID   : 3
        Activity Count   : 0xffffffff      Cookie       : 0x0
        Hardware Index    : 1
Flow 4
  Match
        Source MAC     : 000000-000000    Ethernet Type   : 0x8035
        VLAN ID     : 0        Destination MAC  : 000000-000000
        ARP Opcode: 0           VLAN Priority    : 0
        ARP Source MAC    : 000000-000000
        ARP Target MAC   : 000000-000000
        Source Protocol Address: 0.0.0.0
        Target Protocol Address: 0.0.0.0


         Source IP     : 0.0.0.0
         Destination IP    : 0.0.0.0     ARP Target IP   : 0.0.0.0
         IPv6 Flow Label   : 0
         IPv6 Ext. Header  : None
         ND Source MAC   : 000000-000000
         ND Target IP    : 0:0:0:0:0:0:0:0   ND Destination MAC  :
000000-000000
         IP Protocol     : 0x00
         IP ECN      : 0          IP DSCP : 0
         Source Port     : 0        Destination Port : 0
```

```
Attributes
     Priority      : 32768       Duration:15 secs
     Hard Timeout    : 0 secs      Idle Timeout : 50 secs
     Byte Count     : 0           Packet Count: 0
     Flow Table ID    : 5       Controller ID: 5
     Activity Count     : 0xffffffff     Cookie : 0x0
     Hardware Index     : 1
Instructions
 Write Actions
     Output        : 2/1
     Output      : Controller
```

**Where** the various flows that can be shown using the *flow-type* are:

`destination-ip`
Show flows matching the destination IP address.

`destination-mac`
Show flows matching the destination MAC address.

`destination-port`
Show flows matching the destination port.

`ethernet-type`
Show flows matching the EtherType.

`ip-protocol`
Show flows matching the IP protocol.

`ip-tos-bits`
Show flows matching the IP ToS bits.

`source-ip`
Show flows matching the source IP address.

`source-mac`
Show flows matching the source MAC address.

`source-port`
Show flows matching the source port.

`vlan-id`
Show flows matching the VLAN ID.

`vlan-priority`
Show flows matching the VLAN priority.

destination-ipv6
Show flows matching the destination IPv6 address.

flow-table
Show flows that are hit most corresponding to the flow table number.

ingress-port
Show flows matching the ingress port.

source-ipv6
Show flows matching the source IPv6 address.

## Viewing OpenFlow instances

You can display OpenFlow information for a specific instance. This includes the memberships of OpenFlow instance, the controllers and listen-port for that instance and other relevant information.

### Syntax

```
show openflow instance { instance-name | aggregate }
```

*instance-name*

Displays the OpenFlow configuration for a specific instance.

### Example

```
HP-5406zl(of-inst-test)# show openflow instance test
Configured OF Version    : 1.0
Negotiated OF Version    : 1.0
Instance Name            : test
Admin. Status            : Enabled
Member List              : VLAN 2
Listen Port              : 6633
Oper. Status             : Up
Oper. Status Reason      : NA
Datapath ID              : 00000023209d1bf1
Mode                     : Active
Flow Location            : Hardware and Software
No. of Hw Flows          : 0
No. of Sw Flows          : 0
Hw. Rate Limit           : 0 kbps
Sw. Rate Limit           : 100 pps
Conn. Interrupt Mode     : Fail-Secure
Maximum Backoff Interval : 60 seconds
Probe Interval           : 10 seconds
Hw. Table Miss Count     : 0
No. of Sw Flow Tables    : NA
Egress Only Ports        : None

Controller Id Connection Status Connection State Secure Role
------------- ----------------- ---------------- ------ ------
1             Disconnected      Void             No     Equal
```

The `operational status` can be `down` if:

- The member VLAN of the OpenFlow instance does not exist on the switch
- The controller VLAN of the OpenFlow instance does not exist on the switch
- When multiple controllers connect over multiple controller VLANs, the operational status is`Down` when none of the controller VLANs exist on the switch
- The member VLAN is down, for example when all ports on the member VLAN are down

  If controllers are associated with the instance, then the following table appears:

  ```
  Controller-ID  Connection Status  Connection State
  -------------  -----------------  ----------------
  10             Connected          Active
  11             Disconnected       Void
  13             Connected          Idle
  ```

  Possible connection states are *Active*, *Idle*, *Backoff*, *Connecting*, or *Void*.

  Possible connection status values are *Connected* or *Disconnected*.

## Viewing instance aggregate

Display information of an OpenFlow aggregate instance.

**Example 7** Show OpenFlow instance aggregate

```
show openflow instance titan
Configured OF Version    : 1.3
Negotiated OF Version    : NA
OpenFlow Version         : 1.3
Instance Name            : titan
Admin. Status            : Disabled
Member List              : VLAN 1
Listen Port              : 6633
Oper. Status             : Down
Oper. Status Reason      : No port in member VLAN
[1]Datapath ID            : 0000002320e52b88
Mode                     : NA
Flow Location            : NA
No. of Hw Flows          : 0
No. of Sw Flows          : 0
Hw. Rate Limit           : 0 kbps
Sw. Rate Limit           : 100 pps
Conn. Interrupt Mode     : Fail-Secure
Maximum Backoff Interval : 60 seconds
Probe Interval           : 45 seconds
Hw. Table  Miss Count     : 100
[2]No. of Sw Flow Tables     : 4
[3]Egress Only Ports        : 1/11, 1/23, 1/24
[4]Table Model              : Single Table  [5]


Controller ID Connection Status Connection State Secure Role[6]
------------- ----------------- ---------------- ------ ------
1             Connected         Active           Yes    Master
2             Disconnected      Void             No     Slave

              Auxiliary               Auxiliary     Auxiliary
Controller ID Conn. index Auxiliary ID Conn. Status  Conn. State    Type [7]
 ------------- ----------- ------------ ------------ -------------- ----
1             1               111      Connected     Active         TCP
2             2               121      NA            Send Hello     UDP
```

# Viewing OpenFlow resources

## Syntax

```
show openflow resources
```

1. Reason will be N/A when operational status is up.
2. Only for 1.0 instance. NA for 1.3 instance.
3. Only for 1.3 instance. NA for 1.3 instance.
4. “None” if no ports available.
5. “Single Table” for 1.0 instance. For 1.3 instance the values could be “Policy Engine and Software or IP Control With Policy Engine and Software”
6. Will be equal for 1.0 instance.
7. This table appears only for 1.3 instance.

**Example 8** Show OpenFlow resources

```
HP Switch(config)# show openflow resources

Resource usage in Policy Enforcement Engine

               |   Rules    |  Rules Used
        Slots  | Available  | ACL | QoS | IDM |  VT | Mirr | PBR |  OF  | Other |
--------------+------------+-----+-----+-----+-----+------+-----+------+-------|
  A            |       3055|    0|    0|    0|    0|     0|    0|     0|      0|
  F            |       3055|    0|    0|    0|    0|     0|    0|     0|      0|

               |   Meters   |  Meters Used
        Slots  | Available  | ACL | QoS | IDM |  VT | Mirr | PBR |  OF  | Other |
--------------+------------+-----+-----+-----+-----+------+-----+------+-------|
  A            |        255|     |    0|    0|     |      |     |     0|      0|
  F            |        255|     |    0|    0|     |      |     |     0|      0|

               |Application|
               |Port Ranges|  Application Port Ranges Used
        Slots  | Available  | ACL | QoS | IDM |  VT | Mirr | PBR |  OF  | Other |
--------------+------------+-----+-----+-----+-----+------+-----+------+-------|
  A            |         14|    0|    0|    0|     |     0|    0|      |      0|
  F            |         14|    0|    0|    0|     |     0|    0|      |      0|

 0 of 8 Policy Engine management resources used.

 Key:
 ACL = Access Control Lists
 QoS = Device & Application Port Priority, QoS Policies, ICMP rate limits
 IDM = Identity Driven Management
 VT  = Virus Throttling blocks
 Mirr = Mirror Policies, Remote Intelligent Mirror endpoints
 PBR = Policy Based Routing Policies
 OF = OpenFlow
 Other = management VLAN, DHCP Snooping, ARP Protection, Jumbo IP-MTU,
         Transparent Mode, RA Guard.

 Resource usage includes resources actually in use, or reserved for future
 use by the listed feature. Internal dedicated-purpose resources, such as
 port bandwidth limits or VLAN QoS priority, are not included.
```

## Viewing OpenFlow controllers

Displays OpenFlow controllers configured for use by OpenFlow.

### Syntax

```
show openflow controllers
```

**Example 9 show OpenFlow controllers**

```
HP Switch(config)# show openflow controllers

Controller Information

Controller Id IP Address         Port    Interface
------------------------------- ------ ---------
1              20.0.0.2           6633    VLAN 6
```

## Viewing OpenFlow instance attributes

You can view information on a specific OpenFlow instance.

### Syntax

```
show openflow instance instance-name [ port-statistics | flows [
flow-type ] ]
```

**Where**

`port-statistics`

Shows port statistics.

`flows`

*`flow-type`*

### Example of flow

```
Flow
  Incoming Port :            Ethernet Type    :
  Source MAC    :                          Destination MAC  :
  VLAN ID       :                          VLAN priority    :
  Source IP     :                          Destination IP   :
  IP Protocol   :                          IP ToS Bits      :
  Source Port   :                          Destination Port :
  Priority      :
  Duration      :    seconds
  Idle Timeout  :    seconds               Hard Timeout     :
  Packet Count  :                          Byte Count       :
  Flow Location :
  Actions
    Controller Port
```

**Where** the various flows that can be shown using the *`flow-type`* are:

`destination-ip`

Show flows matching the destination IP address.

`destination-mac`

Show flows matching the destination MAC address.

`destination-port`

Show flows matching the destination port.

`ethernet-type`

Show flows matching the EtherType.

`ip-protocol`

Show flows matching the IP protocol.

`ip-tos-bits`

Show flows matching the IP ToS bits.

`source-ip`

Show flows matching the source IP address.

`source-mac`

Show flows matching the source MAC address.

`source-port`

Show flows matching the source port.

`vlan-id`

Show flows matching the VLAN ID.

`vlan-priority`

Show flows matching the VLAN priority.

`Source IP`
: Shows subnet mask for IP address of the source.

`Destination IP`
: Shows subnet mask for IP address of the destination.

destination-ipv6
: Show flows matching the destination IPv6 address.

flow-table
: Show flows that are hit most corresponding to the flow table number.

ingress-port
: Show flows matching the ingress port.

source-ipv6
: Show flows matching the source IPv6 address.

### Example

```
HP-3800-24G-PoEP-2SFPP# show openflow instance t2 flows\
OpenFlow Flow Table
Flow 1
Match
 Incoming Port : Any          Ethernet Type: IP
 Source MAC: Any           Destination MAC: Any
 VLAN ID       : Any           VLAN priority: Any
  Source IP: 10.0.0.1/32
  Destination IP: Any
  IP Protocol: Any
  IP ECN: AnyIP             DSCP: Any
  Source Port: Any           Destination Port : Any

Attributes
 Priority: 32768           Duration: 41 seconds
  Hard Timeout: 0 seconds        Idle Timeout: 0 seconds
 Byte Count: 0            Packet Count: 0
 Flow Table ID : 100          Controller ID: listen-port
 Activity Count: NA          Cookie: 0x0
 Hardware Index: 1

Instructions
  Apply Actions
 output :24
```

## Viewing flow information

These commands display per instance flow table statistics. This includes both the hardware and software flow tables.

**NOTE:** This option is available only for instances running OpenFlow version 1.0.

### Viewing additional flow information

#### Syntax

`show openflow instance` *`instance-name`* `flows` *`flow-table-id`*

Displays additional information in flows.

## Viewing global flow table information

### Syntax

`show openflow *flow-table*`

Displays global flow table information.

### Example

```
HP-5406zl(of-inst-t1)# show openflow flow-table

Flow Table Information

                                                Max.   Refresh     Flow
Table Name              Usage    Rate (seconds)     Count
----------------------- --------- ---------------  ------
IP Control Table    50%   12         0
Policy Engine Table   50%   20         0

            IP Control Table    Policy Engine Table
Slot ID   Current Usage (%)   Current Usage (%)
-------- -----------------    -------------------
1           0.000000               0.07
6           0.000000               0.07
```

Note: Current usage is percentage of OpenFlow maximum usage

## Viewing specific flow table information

### Syntax

`show openflow instance *instance-name* flow table *flow-table-id* table-capability`

Displays instance specific flow-table information.

## Viewing flow entries

### Syntax

`show openflow instance instance-name flows`

Shows flow entries

## Viewing flow matching

### Syntax

`show openflow instance instance-name flows destination-ip | destination-mac | destination-port | ether-type | ip-protocol | ip-tos-bits | source-ip | source-mac | source-port | vlan-priority | flow-table | ingress-port | destination-ipv6 | source-ipv6`

Shows flow matching the destination for:

IP
MAC
port
Ethernet
IP Protocol
IP ToS bits
IP address

MAC address
source port
source-MAC
VLAN priority
flow table ID[8]
Ingress port
Destination-IPv6
Source IPv6
destination-ipv6
flow-table
ingress-port
source-ipv6

**Example 10 Show flows matching the flow-table-ID**

Syntax

```
show openflow instance t1 flows flow-table
```

## Viewing OpenFlow information for a specific flow table

### Syntax

```
show openflow flow-table
```

Shows OpenFlow display information for a flow table. Option for a specific flow table ID.

**Example 11 Show openflow flow table**

```
show openflow flow-table
Flow Table Information
                                        Refresh
                                 Max.   Rate       Flow
Table Name                       Usage  (seconds)  Count
-------------------------------- ------ ---------- -----
IP Control Table                 50%    20         160
Policy Engine Table              50%    20         200

IP Control Table   Policy   Engine  Table
Slot ID    Current Usage    Current Usage
-------- -------  ------- --------  --------
1          50%               50%
2          50%               50%

Note: Current usage is percentage of OpenFlow maximum usage
```

## Viewing table capability

### Syntax

```
show openflow instance [instance-name]flow-table table-id
table-capability
```

Shows OpenFlow table capability information for a specific flow table ID.

8. This command only works for OpenFlow v1.3

### Example 12 Show table-capability

```
show openflow instance test flow-table 50 table-capability

OpenFlow IP Control Table

Table Match Capabilites:
   VLAN ID
   Source IPv4, IPv6
   Destination IPv4, IPv6
Table Instructions:
   Goto Table 101
Table-Miss Instructions:
   *Goto Table 102

*Currently configured action for table-miss flow.

show openflow instance test flow-table 101 table-capability

OpenFlow Accelerated Table

Table Match Capabilities:
   Incoming Port                  Ethernet Type: IPv4, IPv6
   VLAN ID                        VLAN Priority
   Source IPv4, IPv6              Destination IPv4, IPv6
   IPv6 Flow Label
   IP Protocol                    IP DSCP
   Source Port                    Destination Port
   ICMPv4 Type                    ICMPv4 Code
Table Instructions:
   Metering
      Band Type
         Drop
         Remark DSCP
   Apply-Actions
      Set-Field
         VLAN ID
         VLAN Priority
         Strip VLAN ID
         Source MAC
         Destination MAC
         IP DSCP
      Output
        Switch Port, Drop, Normal
Table-Miss Instructions:
   Apply-Actions
      Output
        Drop, Normal
   *Goto Table 200

*Currently configured action for table-miss flow.


show openflow instance test flow-table 200 table-capability

OpenFlow Software Table 1

Table Match Capabilites:
   Incoming Logical Port              Incoming Physical Port
   Metadata                           Destination MAC
   Ethernet Type                      Source MAC
   VLAN ID                            VLAN Priority
   IP DSCP                            IP ECN
```

```
   IP Proto
   Source IPv4, IPv6                    Destination IPv4, IPv6

   Source Port                          Destination Port
   IPV6 Flow Label
   Source SCTP Port                     Destination SCTP Port
   ICMPv4 Type                          ICMPv4 Code
   ARP Opcode
   ARP Source IPv4                      ARP Destination IPv4
   ARP Source MAC                       ARP Destination MAC
   ICMPv6 Type                          ICMPv6 Code
   IPv6 ND SLL                          IPv6 ND TLL
   ND IPv6 Target
Table Instructions:
   Apply-Actions
      Set-Field
         VLAN ID                        VLAN Priority
         Strip VLAN
         Source MAC                     Destination MAC
         Set TTL                        Decrement TTL
         IP ECN                         IP DSCP
      Output
         Drop, Normal
   Clear-Actions
   Write-Actions
   Write-Metadata
Table-Miss Instructions:
   Apply-Actions
      Output
         Drop, Normal
   Goto Table-201, *202, 203, 204

*Currently configured action for table-miss flow.
```

## Viewing group table information

### Syntax

`show openflow instance` *`instance-name-str`* `| aggregate groups` *`group-id`*

Shows OpenFlow group table information. Groups are supported in software tables. Up to 4 types of groups are supported with 32 groups supported per instance and 1024 groups across all instances. A select group uses the round-robin method for every packet and the number of action buckets capped to 8 per group.

### Example

```
OpenFlow Instance Groups
Group ID          : 32
Group Type        : All
Reference Count   : 1
Packet Count      : 0
Byte Count        : 0
Duration          : 6624
Action Buckets    : 1, 2
 Bucket 1
  Packet Count      : 0
  Byte Count        : 0
  Watch port        : Any
  Weight            : 0
  Action            : Output F24
 Bucket 2
```

```
Packet Count      : 0
Byte Count        : 0
Watch port        : Any
Weight            : 0
Action            : Output F23
```

## Viewing auxiliary information

Only one auxiliary connection is supported per main controller connection.

### Syntax

`show openflow auxiliary-connections`

Displays auxiliary connection information.

**Example 13 Show OpenFlow auxiliary connections**

```
show openflow auxiliary-connections

Auxiliary
Conn. Index Type Port
----------- ---- ----
1           TCP  7777
2           UDP  8888
```

## Viewing per flow rate information

### Syntax

`show openflow instance instance-name limiters`

Displays per-flow rate limiters information.

## Viewing group table information

### Syntax

`show openflow instance instance-name groups`

Displays group table information.

### Example 14 Group table information

```
show openflow instance [instance-name] groups

  Group ID         : 1
  Group Type       : ALL
  Reference Count  : 32767
  Packet Count     : 0
  Byte Count       : 0
  Duration         : 10 seconds
  Action Buckets   : 1, 2
     Bucket 1
        Packet Count     : 0
        Byte Count       : 0
        Watch Port       : Any
        Weight           : o
        Actions          : output A1
     Bucket 2
        Packet Count     : 0
        Byte Count       : 0
        Watch Port       : Any
        Weight           : 0
        Actions          :output F2

  Group ID         : 4
  Group Type       : SELECT
  Reference Count  : 0
  Packet Count     : 0
  Byte Count       : 0
  Duration         : 10
  Action Buckets   : 1
     Bucket 1
        Packet Count     : 0
        Byte Count       : 0
        Watch Port       : Any
        Weight           : 1
        Actions          : output A

  Group ID         : 7
  Group Type       : INDIRECT
  Reference Count  : 0
  Packet Count     : 0
  Byte Count       : 0
  Duration         : 10
  Action Buckets   : 1
     Bucket 1
        Packet Count     : 0
        Byte Count       : 0
        Watch Port       : Any
        Weight           : 0
        Actions          : output A1
  Group ID         : 32
  Group Type       : FAST FAIL OVER
  Reference Count  : 0
  Packet Count     : 0
  Byte Count       : 0
  Duration         : 10
  Action Buckets   : 1
     Bucket 1
        Packet Count     : 0
        Byte Count       : 0
        Watch Port       : A1
        Weight           : 0
        Actions          : output A1
```

## Viewing group information for a specific instance

### Syntax

```
show openflow instance instance-name
groups group-ID
```

Displays group information for a specific instance.

## Viewing meter information for a specific instance

### Syntax

```
show openflow instance instance-name meters
```

Displays meter information. Meters are instances specific. Meters are only supported in hardware tables and the maximum number of meters differ between platforms. DSCP remark type band supported only in extended match mode however DSCP remark type band meter cannot be attached to flows with a non-IP match.

### Example

```
HP-3800-24SFP-2SFPP# show open inst t3 meters

OpenFlow Instance Meters
Meter ID             : 1
Flow Count           : 1
Input Packet Count   : 0
Duration             : 0

          Packet
Band Type  Rate           Count
--------- ------------ --------
Drop        150 kbps        0
```

## Viewing multiport-filter-limit

### Syntax

```
show openflow multiport-filter-limit
```

Displays multiport filter information. (Only in OpenFlow version 1.3.)

**Example 15 Viewing multiport filter information**

```
HP-Switch# show openflow multiport-filter-limit

Total Multiport Filters: 4096

             Filters     Filters     Filters
Features     Allocated   Used        Free
------------ ----------- ----------- -----------
OpenFlow     2048        1500        500
```

# Viewing statistics

## Viewing port statistics per instance

### Syntax

```
show openflow instance instance-name port statistics
```

Displays port statistics information per instance.

### Example 16 Display port statistics for version 1.3

```
HP-Switch# show openflow instance test port-statistics
Number of Ports: 2
Port 47: Up
 Status
  Admin. Status    : Enabled          Flood       : Enabled
  Receive          : Enabled          Forward     : Enabled
  Packet_in        : Enabled
 Statistics
  Collisions       : 0
   Rx Packets       : 0                    Tx Packets   : 68
  Rx Bytes         : 0                    Tx Bytes     : 8066
  Rx Dropped       : 0                    Tx Dropped   : 0
  Rx Errors        : 0                    Tx Errors    : 0
  Frame Errors     : 0
  CRC Errors       : 0
  Overrun Errors   : 0
  Port 48: Down
 Status
   Admin. Status    :                  Flood       :
  Receive          :                  Forward     :
  Packet_in        :
 Statistics
  Collisions       :
   Rx Packets       : 0                    Tx Packets   : 0
  Rx Bytes         : 0                    Tx Bytes     : 0
   Rx Dropped       : 0                    Tx Dropped   : 0
   Rx Errors        : 0                    Tx Errors    : 0
   Frame Errors     : 0
  CRC Errors       : 0
   Overrun Errors   : 0
```

## Viewing message statistics for an instance

This command displays statistics for flow, port, group and meter modification message from the controller, the number of modification messages received from the controller and the number of messages rejected.

### Syntax

`show openflow instance` *`instance-name`* `|` `aggregate message-statistics`

Show message statistics information for an instance. This displays the number of OpenFlow modification messages received from the controller and the number of messages rejected by the switch.

**Example 17 Show OpenFlow instance message-statistics**

```
OpenFlow #: show openflow instance instance-name message statistics
OpenFlow
Message Type     Received Rejected
---------------- -------- --------
OFPT_FLOW_MOD    100      12
OFPT_PORT_MOD    120      22
OFPT_GROUP_MOD   22       2
OFPT_METER_MOD   12       0
```

## Viewing OpenFlow instance information

### Syntax

`show openflow instance`*`instance-name`* `capabilities`

Displays OpenFlow instance capabilities.

# 6 Troubleshooting OpenFlow

## Diagnostic Tools Overview and Usage

### Debug OpenFlow

You can display OpenFlow protocol packets or event description.

**NOTE:** The `debug openflow packets` option only displays OpenFlow protocol packets exchanged between the switch and the controller.

#### Syntax

```
HP Switch#[vlan-id]$ debug openflow
<errors|events|instance|packets>
```

errors
: Display OpenFlow error messages.

events
: Enable debug messages for all OpenFlow events like addition/deletion/modification, enable/disable etc.

instance
: Specify an OpenFlow instance for packet logging.

packets
: Enable debug messages for all OpenFlow packets.

**Example 18 Debug logs**

**Flow deletion**

```
mOFCtrlTask: 00020| DBG|Flow deletion:
idle_timeout=60,dl_type=0x0800,in_port=27,dl_vlan=65535,dl_vlan_pcp=0,
dl_src=00:50:56:9f:5f:0a,dl_dst=00:50:56:9f:19:92,nw_src=1.2.3.6,nw_dst=1.2.3.4,
icmp_type=0,icmp_code=0,actions=output:26
```

**Flow addition**

```
mOFCtrlTask: 00019| DBG|Flow addition:
idle_timeout=60,dl_type=0x0800,in_port=27,dl_vlan=65535,dl_vlan_pcp=0,
dl_src=00:50:56:9f:5f:0a,dl_dst=00:50:56:9f:19:92,nw_src=1.2.3.6,nw_dst=1.2.3.4,
icmp_type=0,icmp_code=0,actions=output:26
```

**Flow expiry**

```
mOFCtrlTask: 00018| DBG|Flow expiry:
idle_timeout=1200,dl_type=0x0800,nw_src=1.2.3.7,nw_dst=1.2.3.8,
actions=mod_nw_src:9.8.7.6
```

## Error messages

### Interoperability error messages

#### Enabling OpenFlow

Enabling OpenFlow when Meshing is enabled will result in an error message similar to the following.

```
OpenFlow cannot be enabled when meshing is configured.
```

#### Enabling meshing

Enabling meshing when OpenFlow is enabled will result in an error message similar to the following.

```
Meshing cannot be configured when OpenFlow is enabled.
```

### Enable OpenFlow with QinQ

Enabling OpenFlow when Q-in-Q is enabled will result in an error message similar to the following.

```
OpenFlow cannot be enabled when Q-in-Q is configured.
```

### Enabling QinQ with OpenFlow

Enabling Q-in-Q when OpenFlow is enabled will result in an error message similar to the following.

```
Q-in-Q cannot be configured when OpenFlow is enabled.
```

### Enabling transparent mode

Enabling Transparent Mode (TRmode) when OpenFlow is enabled will result in an error message similar to the following.

```
Transparent Mode cannot be enabled when OpenFlow is enabled.
```

### Enabling OpenFlow with transparent mode

Enabling OpenFlow when Transparent Mode is enabled will result in an error message similar to the following.

```
OpenFlow cannot be enabled when Transparent Mode is enabled.
```

### Enabling remote mirror endpoint

Enabling Remote Mirror Endpoint when OpenFlow is enabled will result in an error message similar to the following.

```
Remote Mirror Endpoint cannot be configured when OpenFlow
is enabled.
```

### Enabling OpenFlow with remote mirror endpoint

Enabling OpenFlow when Remote Mirror Endpoint is enabled will result in an error message similar to the following.

```
OpenFlow cannot be enabled when Remote Mirror Endpoint is
configured.
```

### Adding a port

When you are adding a port to a trunk which is part of an OpenFlow member VLAN will result in an error message similar to the following.

```
Trunk in use by an OpenFlow instance may not be modified.
```

### Deleting a port

When you are deleting a port from a trunk that is part of an OpenFlow member VLAN will result in an error message similar to the following.

```
Trunk in use by an OpenFlow instance may not be modified.
```

### Moving a trunk

When you are moving a trunk which is part of an OpenFlow member VLAN from one VLAN to another VLAN will result in an error message similar to the following.

```
Trunk in use by an OpenFlow instance may not be moved.
```

### Tagging/Untagging trunk

You are toggling the membership of the trunk from tagged to untagged and that trunk is part of an OpenFlow member VLAN will result in an error message similar to the following.

```
Trunk in use by an OpenFlow instance may not be modified.
```

### Enable LACP

You are trying to enable LACP while OpenFlow is enabled will result in an error message similar to the following.

```
LACP cannot be configured when OpenFlow is enabled.
```

### Enable OpenFlow with LACP

You are trying to enable OpenFlow when LACP is enabled will result in an error message similar to the following.

```
OpenFlow cannot be configured when LACP is enabled.
```

### Show limiters

You are trying to show limiters for an instance running OpenFlow version 1.3 will result in an error message similar to the following.

```
This command is supported only for an OpenFlow version 1.0
instance.
```

### no allow-v1–module

You are trying to run the command `[no] allow-v1-module` when OpenFlow is enabled will result in an error message similar to the following.

```
V1 modules cannot be disabled when OpenFlow is enabled.
```

### allow-v1–module

You are trying to run the command `allow-v1-module` when OpenFlow is enabled will result in an error message similar to the following.

```
V1 modules cannot be enabled when IP Control Table Mode is
enabled.
```

### Non-compatible mode

If you try to enable OpenFlow when a switch is in a non-compatible mode ( no allow-v1–module) will result in an error message similar to the following.

```
OpenFlow cannot be enabled when V1 modules are disabled.
```

### Enable virus throttling

If you are trying to enable virus throttling when OpenFlow is enabled will result in an error message similar to the following.

```
Virus throtting cannot be enabled when OpenFlow is enabled.
```

### Enable OpenFlow with virus throttling

If you are trying to enable OpenFlow when virus throttling is enabled will result in an error message similar to the following.

```
OpenFlow cannot be configured when virus throttling is
enabled.
```

## Controller error messages

### Deleting an unconfigured controller

Attempt to delete a controller that has not been configured will result in an error message similar to the following.

```
HP-8206zl(vlan-3)# no openflow controller-id 2
[controller-id] 2 not found.
```

### Configure or modifying an existing controller

Attempting to configure a controller that already exists or modifying the parameters of an existing controller will result in an error message similar to the following.

```
A controller is already configured with this ID.
```

### Associated controllers

Attempting to delete existing controllers previously associated with an OpenFlow instance will result in an error message similar to the following.

```
Controller cannot be removed when in use by an OpenFlow
instance.
```

### Setting IP Control Table mode

Attempting to set IP Control Table Mode when the switch is in compatible mode will result in an error message similar to the following.

```
IP Control Table Mode cannot be set when V1 module is
enabled.
```

### Specifying and invalid flow table

Attempting to specify an invalid flow table ID will result in an error message similar to the following.

```
Invalid flow table number
```

### Listen port or controller error

Commands issued from listen port or controller are not successful.

1. Enable `debug openflow` which will display the switch output helping you to identify whether the error is occuring at the switch or the controller.
2. Enable `debug openflow instance [instance-name]` to further identify the error.
3. Verify the packet capture for the request and reply to isolate whether the error is occuring at the switch or the controller.

   **NOTE:** This problem will appear in the event that some of the controllers are not fully developed and therefore able to handle replies from the switch. The replies in the packet capture will be visible from the switch but not from the controller.

4. Enable `debug destination session` to further identify the error.

### Specifying a port

If you are trying to specify an application port that is out of range will result in an error message similar to the following.

```
Invalid port. Valid range is 1024-65534.
```

## Port error messages

### Egress-only ports

If you are trying to enable or disable egress only ports when OpenFlow is disabled will result in an error message similar to the following.

```
Egress only ports can be set only when OpenFlow is disabled.
```

## Limiter error messages

### No limiters

If no limiters are configured for an instance will result in an error message similar to the following.

```
No limiters found for this OpenFlow instance.
```

# VLAN error messages

## Member to controller Vlan

Specifying a member VLAN as a controller VLAN will result in an error message similar to the following.

```
The specified VLAN is already member of OpenFlow instance
instance-name and hence cannot be added as controller
interface.
```

## Vlan in an OpenFlow instance

Specifying a VLAN that is already a part of a different OpenFlow instance will result in an error message similar to the following.

```
The VLAN specified is already a member of another OpenFlow
instance.
```

## More than one VLAN

Specifying more than one VLAN per instance will result in an error message similar to the following.

```
Only one VLAN can be configured as a member of an OpenFlow
instance.
```

## Vlan range

Specifying a VLAN that is outside the allowed VLAN range will result in an error message similar to the following.

```
Invalid Input : VLAN-ID
```

## Management VLAN

When the user tries to add the management VLAN to an OpenFlow instance will result in an error message similar to the following.

```
A management VLAN cannot be a member of an OpenFlow instance.
```

## Configure VLAN as management

When the user tries to configure an OpenFlow instance VLAN as management VLAN will result in an error message similar to the following.

```
Management VLAN cannot be configured. VLAN <n> is member of
an OpenFlow instance.
```

## Dynamic VLAN

When a dynamic VLAN is added as a member VLAN will result in an error message similar to the following.

```
Dynamic VLAN cannot be added as a member VLAN.
```

## Controller interface

Adding a controller interface as member VLAN will result in an error message similar to the following.

```
Controller interface cannot be added as member VLAN.
```

# Instance error messages

## Enable a named instance

Attempting to enable a named instance without a listen port or controller and a member VLAN will display an error message similar to the following.

```
A controller, and a member VLAN must be added to the named
instance before enabling it.
```

### Enable an aggregate instance

Attempt to enable an aggregate instance without a listen port or controller will display an error message similar to the following.

```
A listen-port or a controller must be added to the aggregate
instance before enabling it.
```

### Maximum number of instances

Configuring an instance when the maximum number of OpenFlow instances is already configured will display an error message similar to the following.

```
Maximum number of OpenFlow instances (128) already
configured.
```

### Instance name that exceeds length

Configuring an instance with a name that exceeds the maximum length requirement will display an error message similar to the following.

```
Maximum length of the instance-name is 32 characters.
```

### Create an aggregate instance

Trying to create an aggregate instance when a named instance already exists on the switch will display an error message similar to the following.

```
An aggregate instance cannot be created when named instances
exist.
```

### Create a named instance

Trying to create a named instance when an aggregate instance is already configured will display an error message similar to the following.

```
Named instances cannot be created when an aggregate instance
exists.
```

### Deleting an instance

Trying to delete a nonexistent instance will display an error message similar to the following.

```
Instance not found.
```

### Enabling an instance

Attempt to enable an OpenFlow instance without configuring a listen port or a controller will display an error message similar to the following.

```
A listen-port or a controller, and a member VLAN must be
added to the named instance before enabling it.
```

### Delete a member

Trying to delete a member which does not belong to the instance will display an error message similar to the following.

```
VLAN VLAN-ID is not a member of this instance.
```

### Modifying backoff interval

Trying to modify the backoff interval when the instance is enabled will display an error message similar to the following.

```
Instance configuration cannot be modified when the instance
is enabled.
```

### Instance name

When naming an instance, only alphanumeric characters, numerals and underscores are allowed in the instance name. Failure to following this rule will display an error message similar to the following.

```
Invalid name. Only alphanumeric characters and underscores
are allowed.
```

## Troubleshooting scenarios for instances

### Troubleshooting an instance

To troubleshoot an instance check the following.

- To connect a controller there must be ip-connectivity between controller and switch over the controller vlan.
- The controller must be capable of negotiating to a version equal to or less than the configured or supported version.
- The switch must be enabled for the commands `debug openflow` and `debug destination session`.
- If the controller connectivity is lost, the switch must be capable of an instance entered in fail-secure/fail-standalone mode.

### Following the suggested procedure:

- Use the `show openflow` command to check instances

### Example

```
HP-Stack-3800(config)# show openflow
OpenFlow                          : Enabled
IP Control Table Mode         : Disabled

Instance Information
                No. of        No. of       OpenFlow
Instance Name                      Oper.Status  H/W Flows    S/W Flows   Version
------------------------ ---------  ---------      ----------   ----------
test                                  Down          0              0             1.0
```

Or use the following command.

- Use the `show openflow instance [instance-name]` command

### Example

```
HP-Stack-3800(config)# show openflow instance test
Configured OF Version     : 1.0
Negotiated OF Version    : 1.0
Instance Name            : test
Admin. Status            : Enabled
Member List              : VLAN 3
Listen Port              : 6677
Oper. Status             : Down
Oper. Status Reason      : NA
Datapath ID              : 0003082e5f698e25
Mode                     : Active
Flow Location            : Hardware and Software
No. of Hw Flows          : 0
No. of Sw Flows          : 0
Hw. Rate Limit           : 0 kbps
Sw. Rate Limit           : 100 pps
Conn. Interrupt Mode     : Fail-Secure
Maximum Backoff Interval : 60 seconds
```

```
Probe Interval           : 10 seconds
Hw. Table Miss Count     : 0
No. of Sw Flow Tables    : NA
Egress Only Ports        : None
Table Model              : Single Table
Controller Id Connection Status Connection State Secure  Role
------------- ----------------- ---------------- ------  ------
1               Disconnected       Backoff            No      Equal
```

## Commands issued from listen port or controller are not successful

When commands issued from the listen port or the controller are not successful, the following commands can be used to isolate and troubleshoot the problem.

### Syntax

`show openflow instance [instance-name] flow-table [table#] table-capabilities`

To display the table-capabilities of the instance.

### Syntax

`show vlan [member-vlan-no`

To display the port-description.

### Syntax

`show openflow instance [instance-name]`

To see egress-only ports.

### Syntax

`show openflow instance [instance-name]port-statistics`

To display port statisics.

### Syntax

`show openflow instance [instance-name] meters [number]`

To display the meter statistics.

### Syntax

`show openflow instance [instance-name]meter-mod`

To display meter modification.

### Syntax

`show openflow instance [instance-name]group-mod`

To display group modification.

### Syntax

`show openflow instance [instance-name]groups [number]`

To display group statistics.

### Syntax

`show openflow instance [instance-name]port-statistics <egress-only>`

To display port configuration.

## Failover controller connection

For minimal impact to an underlying network when a switch loses connection to the controller, the recommended setting is `fail-standalone` mode.

**NOTE:** Failover flows are not present on 8200/3800 switch. Some controllers instruct switches to remove all flows on failover. Flows will not be present on the switch after failover.

# Flow errors

## Flow modification

### Add/Modify/Delete flow

When a request to add, modify or delete a flow mod is rejected by the switch, use the following command.

#### Syntax

```
show openflow instance [instance-name]message-statistics
```

#### Example

```
HP-Stack-3800(config-class)# show openflow instance test
message-statistics

OpenFlow
Message Type        Received          Rejected
 ----------------  ---------------  --------
OFPT_FLOW_MOD         0                  0
OFPT_PORT_MOD         0                  0
OFPT_GROUP_MOD        0                  0
OFPT_METER_MOD        0                  0
```

### Verifying flows

The flow can be verified at the switch by using the following command.

#### Syntax

```
show openflow instance [instance-name]flows
```

Enable `debug openflow` at the switch. Run the command and observe the debug output for more specific rejection reasons why the flow is rejected by the switch.

**NOTE:** Simular troublshooting techniques can be employed for port-modification, meter-modification and group-modification issues.

## Programming flow errors

When programming flows on a controller, an error can be resolved by using the following options.

- A error message displays while programming a flow in a policy table.
- Use the command `debug openflow` and `debug destination session` to locate a more descriptive error message in the debug log.

### Programming flow errors

While programming flows the controller displays an error message similar to the following.

- `Table 0 is read only, flow add/del/mod is not allowed in this table.`
- `Flows with priotiy "0" can not be programmed.`

- `srcipand dstIp are compulsory to be passed in the flow.`
- `srcIp/dstIp with subnet mask values are not allowed, exact match ip-address is only allowed.`
- `Only IPv4/IPv6 traffic is allowed to be matched.`

### Example

Error messages for Table 50 flow restrictions can be similar to the following.

- `Table 50 has a flow restriction - Default rule (flow with match on any, with priority 0) for table 50 is read only, add/del/mod is not allowed for this.`
- `Table 50 only unicastip-address are allowed to be programmed.`
- `Only "goto 101" instruction is allowed in this table.`
- `Flow with invalid member vlan (vlan which is not created and not a member vlan of an instance) is not allowed.`
- A rule programmed like `v1, srcIP1, dstIP1` will not be allowed if you try to program `*, srcIP1, dstIP1` in virtual mode.
- A rule programmed like `v1, srcIP1, dstIP1` will not be allowed if you try to program `v1, srcIP1, dstIP1, Priority2`

### Policy table restrictions

- For table 100, 101, 102 — An aggregate mode output port action is allowed only if the flow had matching criteria as member-vlan or modify-vlan also specified in the action.
- Flow with normal action in table 100, 101, and 102 — Modifications vlan-pcp and modify-tosmodify are allowed. All remaining modify actions are not allowed with normal action.

### Traffic flow is added but not seen

#### OpenFlow v1.0 instance

OpenFlow v 1.0 is a single table model. The table miss rule for 1.0 instance is `gotocontroller`.

#### OpenFlow v1.3 instance

OpenFlow v1.3 is a multi table model. For every table, the default table miss rule is to "drop" the packet. For a v1.3 instance, a flow is added in any table present, before or after that table. The table-miss rule needs to be modified so that the traffic hits the programmed flow.

#### Example

0, 100, 200-201-202-203 (in multi-table model without ip-control table mode)

- If a rule is programed into table 200 which permits traffic then the "table miss rule" for table 100 will not hit the rules in table 200.
- If a rule is programed into table 100 with "gototable 200", then traffic will not proceed until either a matching rule is programmed in table 200 or "table miss rule" is programmed for table 200.

**NOTE:** A table miss rule is a flow with the priority of 0 and all match fields wildcarded when the command `show openflow instance [instance-name] flows` is used.

## Missing flow after successful add flow-mod

After successfully adding a flow-mod the flow is not appearing on the switch, verify the idle-timeout/hard-timeout of the flow.

- Table 50 supports only 12 seconds as a hardware refresh rate. Flows programmed in this table should at least have idle-timeout of 24 seconds.
- Table 100/101/102 supports an idle-timeout within the 5 seconds to 5 minutes range. If the “policy-engine-table refresh-interval” is configured for 5 seconds, then minimum idle-timeout supported is for 10 seconds (double the time configured.)
- Increase the idle-timeout for the programmed flow. Depending on the activity, setting the idle-timeout to 0 will not expire the flow.

## Hardware accelerated flows

In learning switch application performance, flows with L2 and L3 information can be hardware accelerated on v2 blades without `ip-control-table` mode configured on the switch. By default hardware acceleration is disabled.

To enable hardware acceleration:

1. Run a `show module` command to verify v2 blades are present on the switch.
2. Configure `no allow-v1-module` command to run v2 capabilities on v2 blades at the switch.
3. If `ip-control-table mode` is enabled then run the command`no openflowip-control-table` to disable.

## Missing line rate performance after Flow-mod is successful.

If an instance is running OpenFlow v1.0 and the flow can not be accommodated in the hardware or a higher priority rule is present in the swor, we have reached the policy engine usage limit configured. When the policy engine usage limit is configured, the flow will be housed in the software table. This is verified by the `show openflow instance [instance-name] flows` command.

If the flow is programmed in swthen, the line rate performance will not be seen as packet forwarding.

Reasons for this missing line rate performance include:

- The rule has a match criterion for VLAN PCP.
- The rule has a match criterion for MAC address.
- The rule has a match criterion for non-IP Ether type.
- The rule has a match criterion for non-existing VLAN.
- The ICMP information is missing in the rule.
- The input port and output port is same for this rule.
- The action field provided in the rule is not supported.
- The VLAN information in the action field of the rule is invalid or missing.
- The output port in the action field of the rule is invalid.
- The action and output port in the rule is invalid or not supported.
- Hardware resources are not available.
- Rule cannot be accelerated in hardware.

The `openflow instance [instance-name]` limit software-rate [`1-10000`] command can be used to modify the se rate limiting value, associate a separate meter with this flow or change the software rate limiter value.

The command `gotocontroller/goto 200`can be programmed in the policy table. This rule takes traffic to the software flow tables and either sends traffic to the controller or match the traffic in the swtable. This rule stays in hardware but it is limited by the shared global rate limiter.

For 1.3 instance we have separate swtables 200 to 203.

## Errors concerning auxiliary connections

### Removing auxiliary connection

Removing an auxiliary connection which is associated displays an error message similar to the following.

```
Auxiliary connection is in use by an OpenFlow instance and
cannot be removed.
```

### Deleting auxiliary connection

Deleting an auxiliary connection that has not been configured displays an error message similar to the following.

```
Auxiliary connection index not found.
```

### Associating multiple auxiliary indexes

Associating more than one auxiliary index displays an error message similar to the following.

```
Only one auxiliary connection can be configured per main
controller connection.
```

### Associating unconfigured auxiliary indexes

Associating an auxiliary index which is not configured displays an error message similar to the following.

```
No auxiliary connection is configured with this index.
```

### Checking for the static limit

Checking for the static limit and error out while configuring displays an error message similar to the following.

```
Maximum number of auxiliary connections configured.
```

### Associating auxiliary connection

Associating an auxiliary connection to an instance running version 1.0 displays an error message similar to the following.

```
Auxiliary connection can only be associated with instance
running version 1.3 and above.
```

### Associating multiple auxiliary connections

Associating more than one auxiliary connection to an instance controller connection displays an error message similar to the following.

```
Only one auxiliary connection can be configured per main
controller connection.
```

## Troubleshooting scenarios and error messages

### Setting policy engine resource usage when OpenFlow is enabled

When the policy engine resource usage is set while OpenFlow is enabled, will display an error message similar to the following.

```
Resource usage can be set only when OpenFlow is disabled.
```

### Securing a connection with no certificate configured

When securing a connection with no certificate configured for OpenFlow, will display an error message similar to the following.

```
Certificate for OpenFlow is not configured.
```

### Setting the protocol version with instances enabled:

Setting the protocol version when instances are enabled will display a message similar to the following.

```
Instance configuration cannot be modified when the instance
is enabled.
```

### Entering the wrong protocol version

Entering the wrong protocol version will display an error message similar to the following.

```
Invalid protocol version.
```

## How to troubleshoot if instance is not coming up

When an instance is not coming up, use the following commands to troubleshoot the instance status.

1. Run the command `HP-Stack-3800(config)# show openflow`

```
HP-Stack-3800(config)# show openflow
OpenFlow                    : Enabled
IP Control Table Mode       : Disabled
Instance Information
                    No. of       No. of       OpenFlow
Instance Name                      Oper. Status  H/W Flows   S/W Flows   Version
---------------------------------- ------------  ----------  ----------  --------
 test                                  Down     0             0           1.0
```

2. Run the command `HP-Stack-3800(config)# show openflowinstance test`

```
HP-Stack-3800(config)# show openflowinstance test
Configured OF Version     : 1.0
Negotiated OF Version    : 1.0
Instance Name            : test
Admin. Status            : Enabled
Member List              : VLAN 3
Listen Port              : 6677
Oper. Status             : Down
Oper. Status Reason      : NA
Datapath ID              : 0003082e5f698e25
Mode                     : Active
Flow Location            : Hardware and Software
No. of Hw Flows          : 0
No. of Sw Flows          : 0
Hw. Rate Limit           : 0 kbps
Sw. Rate Limit           : 100 pps
Conn. Interrupt Mode     : Fail-Secure
Maximum Backoff Interval : 60 seconds
Probe Interval           : 10 seconds
Hw. Table Miss Count     : 0
No. of Sw Flow Tables    : NA
Egress Only Ports        : None
Table Model              : Single Table

Controller Id Connection Status Connection State Secure Role
------------- ----------------- ---------------- ------ ------
1               Disconnected        Backoff           No     Equal
```

3. Make certain that the OperStatus is “Up” for the instance.
4. The member VLAN must be created on the switch and that the member VLAN must have at least one port associated with it.
5. The Vlan associated with the controller must be created on the switch.
6. The switch (hardware) should have the resource to program the default rule.
7. Run the command `show log – r` to see more reasons why the instance is not up.

To have Connection status “Connected” follow these guidelines:

- There should be ip-connectivity between controller and switch over the controller VLAN.
- The controller has to be capable of negotiating a version equal to or less than the configured or supported version.
- The commands `debug openflow` and `debug destination session` can be enabled on the switch to see the debug messages.
- On loss of controller connectivity, instance enters in fail-secure/fail-standalone mode.

## How to troubleshoot unsuccessful commands issued from listen port or controller

Attempt the following commands to determine access and programming on the switch from the controller.

### Syntax

```
show openflow instance [instance-name] flow-table
[table-numtable-capabilities]
```

Displays table capabilities.

### Syntax

```
show openflow instance [instance-name] capabilities
```

Displays table features.

### Syntax

```
show vlan [member-vlan-no]
```

and the command `show openflowinstance [instance-name]`

Displays a port description. The results of the command will display eggress-only ports.

### Syntax

```
show openflow instance [instance-name] port-statistics
```

Displays the statistics for the given port.

### Syntax

```
show openflow instance [instance-name] message-statistics
```

Displays the flow modification (add/delete/modifiy) for the given instance.

### Syntax

```
show openflow instance [instance-name]flows
```

Display the dump flows for the given instance.

### Syntax

```
show openflow instance [instance-name] message-statistics
```

Displays the meter modification (add/delete/modifiy) for the given instance.

Syntax

```
show openflow instance [instance-name] meters [number]
```

Syntax

```
show openflow instance [instance-name] message-statistics
```

Displays the group modification (add/delete/modifiy) for the given instance.

Syntax

```
show openflow instance [instance-name] groups [number]
```

Diplays the group statistics for a given instance.

Syntax

```
show openflow instance [instance-name] message-statistics
```

Displays the port modification (add/delete/modifiy) for the given instance.

Syntax

```
show openflow instance [instance-name] port-statistics
<egress-only>
```

Displays the port statistics (add/delete/modifiy)—egress-only—for the given instance.

After running the above listed commands to troubleshoot the issue, enable `debug openflow` on the switch. Run the next series of commands which will provide the output of the switch. This will help to identify if the problem is on the switch or the controller.

Syntax

```
debug openflow
```

Syntax

```
debug openflow instance [instance-name]
```

Syntax

```
debug destination session
```

Verify the Packet capture for request and reply to isolate if it is controller or switch problem.

**NOTE:** Some controllers are not fully developed to accept replies from the switch. When this problem occurs, the reply of the packet capture is seen from the switch but controller side of the packet capture will not displa.y

Example

In this example, the results of the command `dpctl tcp:<IP_addrs>:listen-port table-features` results in a **segmentation fault** on the controller. The next step is to verify the packet capture for the request and reply to isolate the issue to either the controller or switch.

```
HP-2920-48G(config)# sh openflow instance t1 flow-table 100 table-capability

 OpenFlow Flow Table Properties

 Table Match Capabilities:
   Incoming Port                     Ethernet Source
   Ethernet Type                     VLAN ID
   VLAN PCP                          IP DSCP
   IP Protocol                       IPv4 Source Address
   IPv4 Destination Address          TCP Source Port
   TCP Destination Port              UDP Source Port
   UDP Destination Port              ICMP Type
   ICMP Code                         IPv6 Source Address
   IPv6 Destination Address
 Table Instructions:
   Metering
   Apply Actions
      Set-Field
         Ethernet Destination         Ethernet Source
         VLAN ID                      VLAN PCP
         IP DSCP
      Output
   GoTo 200
 Table-Miss Instructions:
   Metering
   Apply Actions
      Output
   GoTo 200
```

```
openflow@openflow-ubuntu-16:~$ /home/openflow/dpctl tcp:10.20.30.40:6633 table-features

SENDING:
stat_req{type="table-features", flags="0x0"}

Oct 11 03:09:55|00001|ofl_str|WARN|Received property has invalid length (prop->length=4, data_len=112).
Oct 11 03:09:55|00002|ofl_str|WARN|Received property has invalid length (prop->length=5, data_len=256).
Oct 11 03:09:55|00003|ofl_str|WARN|Received property has invalid length (prop->length=7, data_len=880).
Oct 11 03:09:55|00004|ofl_str|WARN|Received property has invalid length (prop->length=6, data_len=880).
Oct 11 03:09:55|00005|ofl_str|WARN|Received property has invalid length (prop->length=5, data_len=880).
Oct 11 03:09:55|00006|ofl_str|WARN|Received property has invalid length (prop->length=4, data_len=880).
Oct 11 03:09:55|00007|ofl_msg_u|WARN|Received message seemed to be valid, but it contained unused data (4496).
Segmentation fault
```

```
HP-5406zl# sh openflow instance t1 capabilities

Policy Engine Match Capability :  Standard Match

 Switch Capabilities
 --------------------
 Flow Statistics
 Table Statistics
 Port Statistics
 Group Statistics
 Block Ports
```

## Example

The following are examples of commands showing switch information.

```
HP-2920-48G(config)# show vlan 3

 Status and Counters - VLAN Information - VLAN 3

  VLAN ID : 3
  Name : VLAN3
  Status : Port-based
  Voice : No
  Jumbo : No

  Port Information Mode     Unknown VLAN Status
  ---------------- -------- ------------ ----------
  2                Untagged Learn        Up
  47               Untagged Learn        Up
```

```
openflow@openflow-ubuntu-16:~$ ./dpctl tcp:10.20.30.40:6633 port-desc

SENDING:
stat_req{type="port-desc", flags="0x0"}


RECEIVED:
stat_repl{type="port-desc", flags="0x0"{no="47", hw_addr="d8:9d:67:8b:71:11", name="47", config="0x0", state="0x4", cu
rr="0x220", adv="0x0", supp="0x22f", peer="0x0", curr_spd="3567587328kbps", max_spd="3567587328kbps"}, {no="local", hw
_addr="d8:9d:67:8b:71:00", name="local", config="0x0", state="0x0", curr="0x0", adv="0x0", supp="0x0", peer="0x0", cur
r_spd="0kbps", max_spd="0kbps"}, {no="2", hw_addr="d8:9d:67:8b:71:3e", name="2", config="0x0", state="0x4", curr="0x22
0", adv="0x0", supp="0x22f", peer="0x0", curr_spd="3567587328kbps", max_spd="3567587328kbps"}}}
```

```
HP-Stack-2920# show openflow instance t1

 Configured OF Version    : 1.3
 Negotiated OF Version    : 1.3
 Instance Name            : t1
 Admin. Status            : Enabled
 Member List              : VLAN 3
 Listen Port              : 6633
 Oper. Status             : Up
 Oper. Status Reason      : NA
 Datapath ID              : 000338eaa72a490b
 Mode                     : Active
 Flow Location            : Hardware and Software
 No. of Hw Flows          : 3
 No. of Sw Flows          : 4
 Hw. Rate Limit           : 0 kbps
 Sw. Rate Limit           : 100 pps
 Conn. Interrupt Mode     : Fail-Secure
 Maximum Backoff Interval : 60 seconds
 Probe Interval           : 10 seconds
 Hw. Table Miss Count     : NA
 No. of Sw Flow Tables    : 4
 Egress Only Ports        : 1/10-1/48,2/1-2/24,3/10-3/24
 Table Model              : Policy Engine and Software

 Controller Id Connection Status Connection State Secure Role
 ------------- ----------------- ---------------- ------ ------
 1             Connected         Active           No     Equal
```

```
HP-2920-48G(config)# sh openflow instance t1

 Configured OF Version    : 1.3
 Negotiated OF Version    : 1.3
 Instance Name            : t1
 Admin. Status            : Enabled
 Member List              : VLAN 3
 Listen Port              : 6633
 Oper. Status             : Up
 Oper. Status Reason      : NA
 Datapath ID              : 0003d89d678b7100
 Mode                     : Active
 Flow Location            : Hardware and Software
 No. of Hw Flows          : 2
 No. of Sw Flows          : 4
 Hw. Rate Limit           : 0 kbps
 Sw. Rate Limit           : 100 pps
 Conn. Interrupt Mode     : Fail-Secure
 Maximum Backoff Interval : 60 seconds
 Probe Interval           : 10 seconds
 Hw. Table Miss Count     : NA
 No. of Sw Flow Tables    : 4
 Egress Only Ports        : None
 Table Model              : Policy Engine and Software

 Controller Id Connection Status Connection State Secure Role
 ------------- ----------------- ---------------- ------ ------
 1             Connected         Active           No     Equal
```

```
HP-2920-48G(config)# sh openflow instance t1 port-statistics

Number of Ports :2

Port 2: Up
Status
  Admin. Status   : Enabled          Flood       : NA
  Receive         : Enabled          Forward     : Enabled
  Packet_in       : Enabled
Statistics
  Collisions      : 0
  Rx Packets      : 0                    Tx Packets  : 317
  Rx Bytes        : 0                    Tx Bytes    : 63684
  Rx Dropped      : 0                    Tx Dropped  : 0
  Rx Errors       : 0                    Tx Errors   : 0
  Frame Errors    : 0
  CRC Errors      : 0
  Overrun Errors  : 0

Port 47: Up
Status
  Admin. Status   : Enabled          Flood       : NA
  Receive         : Enabled          Forward     : Enabled
  Packet_in       : Enabled
Statistics
  Collisions      : 0
  Rx Packets      : 0                    Tx Packets  : 318
  Rx Bytes        : 0                    Tx Bytes    : 64546
  Rx Dropped      : 0                    Tx Dropped  : 0
  Rx Errors       : 0                    Tx Errors   : 0
  Frame Errors    : 0
  CRC Errors      : 0
  Overrun Errors  : 0
```

```
openflow@openflow-ubuntu-16:~$ /home/openflow/dpctl tcp:10.20.30.40:6633 stats-port

SENDING:
stat_req{type="port", flags="0x0", port="any"}


RECEIVED:
stat_repl{type="port", flags="0x0", stats=[{port="47", rx_pkt="0", tx_pkt="474", rx_bytes="0", tx_bytes="100738", rx_d
rops="0", tx_drops="0", rx_errs="0", tx_errs="0", rx_frm="0", rx_over="0", rx_crc="0", coll="0"}, {port="local", rx_pk
t="0", tx_pkt="0", rx_bytes="0", tx_bytes="0", rx_drops="0", tx_drops="0", rx_errs="0", tx_errs="0", rx_frm="0", rx_ov
er="0", rx_crc="0", coll="0"}, {port="2", rx_pkt="0", tx_pkt="473", rx_bytes="0", tx_bytes="99564", rx_drops="0", tx_d
rops="0", rx_errs="0", tx_errs="0", rx_frm="0", rx_over="0", rx_crc="0", coll="0"}]}
```

## Example

Displays the flow modifications on the controller.

```
openflow@openflow-ubuntu-16:~$ /home/openflow/dpctl tcp:10.20.30.40:6633 flow-mod cmd=add,table=100, eth_type=0x800,ip
_dst=1.1.1.2,ip_src=1.1.1.1 apply:output=47

SENDING:
flow_mod{table="100", cmd="add", cookie="0x0", mask="0x0", idle="0", hard="0", prio="32768", buf="none", port="any", g
roup="any", flags="0x0", match=oxm{ipv4_dst="1.1.1.2", ipv4_src="1.1.1.1", eth_type="0x800"}, insts=[apply{acts=[out{p
ort="47"}]}]}


OK.
```

Displays the flow modifications on the switch.

```
HP-2920-48G(openflow)# show openflow instance t1 message-statistics

 OpenFlow
 Message Type     Received        Rejected
 ---------------- --------------- --------
 OFPT_FLOW_MOD    2               1
 OFPT_PORT_MOD    0               0
 OFPT_GROUP_MOD   0               0
 OFPT_METER_MOD   0               0
```

Dump flow switch command.

```
HP-2920-48G(config)# sh openflow instance t1 flows

 OpenFlow Flow Table

 Flow 1
 Match
  Incoming Port : Any                  Ethernet Type    : Any
  Source MAC    : Any                  Destination MAC  : Any
  VLAN ID       : Any                  VLAN priority    : Any
  Source Protocol Address : Any
  Target Protocol Address : Any
  IP Protocol   : Any
  IP ECN        : Any                  IP DSCP          : Any
  Source Port   : Any                  Destination Port : Any
 Attributes
  Priority      : 0                    Duration         : 3367 seconds
  Hard Timeout  : 0 seconds            Idle Timeout     : 0 seconds
  Byte Count    : 0                    Packet Count     : NA
  Flow Table ID : 0                    Controller ID    : NA
  Activity Count: NA                   Cookie           : 0x0
  Hardware Index     : NA
 Instructions
   Goto Table ID                : 100

 Flow 2
 Match
  Incoming Port : Any                  Ethernet Type    : Any
  Source MAC    : Any                  Destination MAC  : Any
  VLAN ID       : Any                  VLAN priority    : Any
  Source Protocol Address : Any
  Target Protocol Address : Any
  IP Protocol   : Any
  IP ECN        : Any                  IP DSCP          : Any
  Source Port   : Any                  Destination Port : Any
 Attributes
  Priority      : 0                    Duration         : 3367 seconds
  Hard Timeout  : 0 seconds            Idle Timeout     : 0 seconds
  Byte Count    : NA                   Packet Count     : 0
  Flow Table ID : 100                  Controller ID    : NA
  Activity Count: NA                   Cookie           : 0x0
  Hardware Index     : NA
 Instructions
    Drop
```

## Example

Meter modification controller command.

```
openflow@openflow-ubuntu-16:~$ ./dpctl tcp:10.20.30.40:6633 meter-mod cmd=add,flags=1,meter=125 dscp_remark:rate=512

SENDING:
meter_mod{cmd="add", flags="0x1"", meter_id="7d"", bands=[{type = dscp_remark, rate="512", burst_size="0", prec_level=
"0"}]}


OK.
```

Meter modification switch command.

```
HP-2920-48G(openflow)# show openflow instance t1 message-statistics

 OpenFlow
 Message Type      Received        Rejected
 ----------------- --------------- --------
 OFPT_FLOW_MOD     2               1
 OFPT_PORT_MOD     0               0
 OFPT_GROUP_MOD    0               0
 OFPT_METER_MOD    4               2
```

## Example

Meter statistics controller command.

```
openflow@openflow-ubuntu-16:~$ /home/openflow/dpctl tcp:10.20.30.40:6633 stats-meter

SENDING:
stat_req{type="mstats", flags="0x0"{meter_id= ffffffff"}


RECEIVED:
stat_repl{type="mstats", flags="0x0", stats=[{meter= 7c"", flow_cnt="0", pkt_in_cnt="0", byte_in_cnt="0", duration_sec
="3664437096", duration_nsec="3646967296", bands=[{pkt_band_cnt="0", byte_band_cnt="0"}]}, {meter= 7d"", flow_cnt="0",
 pkt_in_cnt="0", byte_in_cnt="0", duration_sec="3664436823", duration_nsec="3366967296", bands=[{pkt_band_cnt="0", byt
e_band_cnt="0"}]}]}
```

Meter statistics switch command.

```
HP-2920-48G(openflow)# show openflow instance t1 meters 125

 OpenFlow Instance Meters

 Meter ID             : 125
 Flow Count           : 0
 Input Packet Count   : 0
 Duration             : 0

                        Packet
 Band Type Rate         Count
 --------- ------------ --------
 Mark      512 kbps     0
```

## Example

Displays the group modifications for the controller.

```
openflow@openflow-ubuntu-16:~$ ./dpctl tcp:10.20.30.40:6633 group-mod cmd=add,type=all,group=1 weight=0,port=any,group
=any  output=1 weight=0,port=any,group=any output=2

SENDING:
grp_mod{group="1", cmd="add", type="all", buckets=[{w="0", wprt="any", wgrp="any", acts=[out{port="1"}]}, {w="0", wprt
="any", wgrp="any", acts=[out{port="2"}]}]}


RECEIVED:
error{type="BAD_ACTION", code="BAD_OUT_PORT", dlen="64"}

openflow@openflow-ubuntu-16:~$ ./dpctl tcp:10.20.30.40:6633 group-mod cmd=add,type=all,group=1 weight=0,port=any,group
=any  output=2 weight=0,port=any,group=any output=2

SENDING:
grp_mod{group="1", cmd="add", type="all", buckets=[{w="0", wprt="any", wgrp="any", acts=[out{port="2"}]}, {w="0", wprt
="any", wgrp="any", acts=[out{port="2"}]}]}


OK.
```

Displays the group modifications on the switch.

```
HP-2920-48G(config)# show openflow instance t1 message-statistics

 OpenFlow
 Message Type     Received        Rejected
 ---------------- --------------- --------
 OFPT_FLOW_MOD    2               1
 OFPT_PORT_MOD    0               0
 OFPT_GROUP_MOD   7               6
 OFPT_METER_MOD   4               2
```

## Example

Group statistics switch command.

```
HP-2920-48G(config)# show openflow instance t1 groups 1

 OpenFlow Instance Groups

   Group ID         : 1
   Group Type       : All
   Reference Count  : 0
   Packet Count     : 0
   Byte Count       : 0
   Duration         : 525
   Action Buckets   : 1, 2
      Bucket 1
         Packet Count     : 0
         Byte Count       : 0
         Watch port       : Any
         Weight           : 0
         Action           : Output 2
      Bucket 2
         Packet Count     : 0
         Byte Count       : 0
         Watch port       : Any
         Weight           : 0
         Action           : Output 2
```

Displays group statistics on the controller.

```
openflow@openflow-ubuntu-08:~$ ./dpctl tcp:10.20.30.40:6633 stats-group

SENDING:
stat_req{type="grp", flags="0x0", group="all"}


RECEIVED:
stat_repl{type="grp", flags="0x0", stats=[{group="1", ref_cnt="0", pkt_cnt="0", byte_cnt="0", cntrs=[{pkt_cnt="0", by
te_cnt="0"}]}, {group="2", ref_cnt="0", pkt_cnt="0", byte_cnt="0", cntrs=[{pkt_cnt="0", byte_cnt="0"}]}]}
```

```
HP-Stack-2920$ show openflow instance t1 port-statistics

Number of Ports :17

Port 1/2: Up
Status
  Admin. Status    : Enabled          Flood       : NA
  Receive          : Enabled          Forward     : Enabled
  Packet_in        : Disabled
Statistics
  Collisions       : 0
  Rx Packets       : 635                  Tx Packets  : 3154604
  Rx Bytes         : 47842                Tx Bytes    : 201973090
  Rx Dropped       : 0                    Tx Dropped  : 0
  Rx Errors        : 0                    Tx Errors   : 0
  Frame Errors     : 0
  CRC Errors       : 0
  Overrun Errors   : 0

Port 1/3: Down
Status
  Admin. Status    :                  Flood       :
  Receive          :                  Forward     :
  Packet_in        :
Statistics
  Collisions       :
  Rx Packets       : 0                    Tx Packets  : 0
  Rx Bytes         : 0                    Tx Bytes    : 0
  Rx Dropped       : 0                    Tx Dropped  : 0
  Rx Errors        : 0                    Tx Errors   : 0
  Frame Errors     : 0
  CRC Errors       : 0
  Overrun Errors   : 0
```

Port Modification shown on the controller.

```
openflow@openflow-ubuntu-08:~$ ./dpctl tcp:10.20.30.40:6633 port-mod port=2,addr=38:ea:a7:2a:49:3f,conf=0x40,mask=0x4
0

SENDING:
port_mod{port="2", hwaddr="38:ea:a7:2a:49:3f", config="0x00000040", mask="0x40", adv="0x0"}


RECEIVED:
error{type="PORT_MOD_FAILED", code="BAD_HW_ADDR", dlen="40"}

openflow@openflow-ubuntu-08:~$ ./dpctl tcp:10.20.30.40:6633 port-mod port=2,addr=38:ea:a7:2a:49:3e,conf=0x40,mask=0x4
0

SENDING:
port_mod{port="2", hwaddr="38:ea:a7:2a:49:3e", config="0x00000040", mask="0x40", adv="0x0"}


OK.
```

Port modification shown on the switch.

```
HP-Stack-2920(config)# sh openflow instance t1 message-statistics

 OpenFlow
 Message Type      Received        Rejected
 ----------------- --------------- --------
 OFPT_FLOW_MOD     0               0
 OFPT_PORT_MOD     2               1
 OFPT_GROUP_MOD    0               0
 OFPT_METER_MOD    0               0
```

## Example

Port configuration shown on the switch.

```
HP-2920-48G(config)# sh openflow instance t1 port-statistics

Number of Ports :2

Port 2: Up
Status
  Admin. Status   : Enabled          Flood       : NA
  Receive         : Enabled          Forward     : Enabled
  Packet_in       : Enabled
Statistics
  Collisions      : 0
  Rx Packets      : 0                    Tx Packets  : 317
  Rx Bytes        : 0                    Tx Bytes    : 63684
  Rx Dropped      : 0                    Tx Dropped  : 0
  Rx Errors       : 0                    Tx Errors   : 0
  Frame Errors    : 0
  CRC Errors      : 0
  Overrun Errors  : 0

Port 47: Up
Status
  Admin. Status   : Enabled          Flood       : NA
  Receive         : Enabled          Forward     : Enabled
  Packet_in       : Enabled
Statistics
  Collisions      : 0
  Rx Packets      : 0                    Tx Packets  : 318
  Rx Bytes        : 0                    Tx Bytes    : 64546
  Rx Dropped      : 0                    Tx Dropped  : 0
  Rx Errors       : 0                    Tx Errors   : 0
  Frame Errors    : 0
  CRC Errors      : 0
  Overrun Errors  : 0
```

Port configuration shown on the switch.

```
HP-Stack-2920# show openflow instance t1 port-statistics egress-only

Number of Ports :78

Port 1/10: Down
Status
  Admin. Status   :                  Flood       :
  Receive         :                  Forward     :
  Packet_in       :
Statistics
  Collisions      :
  Rx Packets      : 0                    Tx Packets  : 0
  Rx Bytes        : 0                    Tx Bytes    : 0
  Rx Dropped      : 0                    Tx Dropped  : 0
  Rx Errors       : 0                    Tx Errors   : 0
  Frame Errors    : 0
  CRC Errors      : 0
  Overrun Errors  : 0

Port 2/1: Down
Status
  Admin. Status   :                  Flood       :
  Receive         :                  Forward     :
  Packet_in       :
Statistics
  Collisions      :
  Rx Packets      : 23                   Tx Packets  : 227
  Rx Bytes        : 1500                 Tx Bytes    : 20513
  Rx Dropped      : 0                    Tx Dropped  : 0
  Rx Errors       : 0                    Tx Errors   : 0
  Frame Errors    : 0
  CRC Errors      : 0
  Overrun Errors  : 0
```

## Flow not added/deleted/modified

For flow modification troubleshooting, run the command `show openflow instance test message-statistics` which indicates if the flow-mod request was accepted or rejected by the switch.

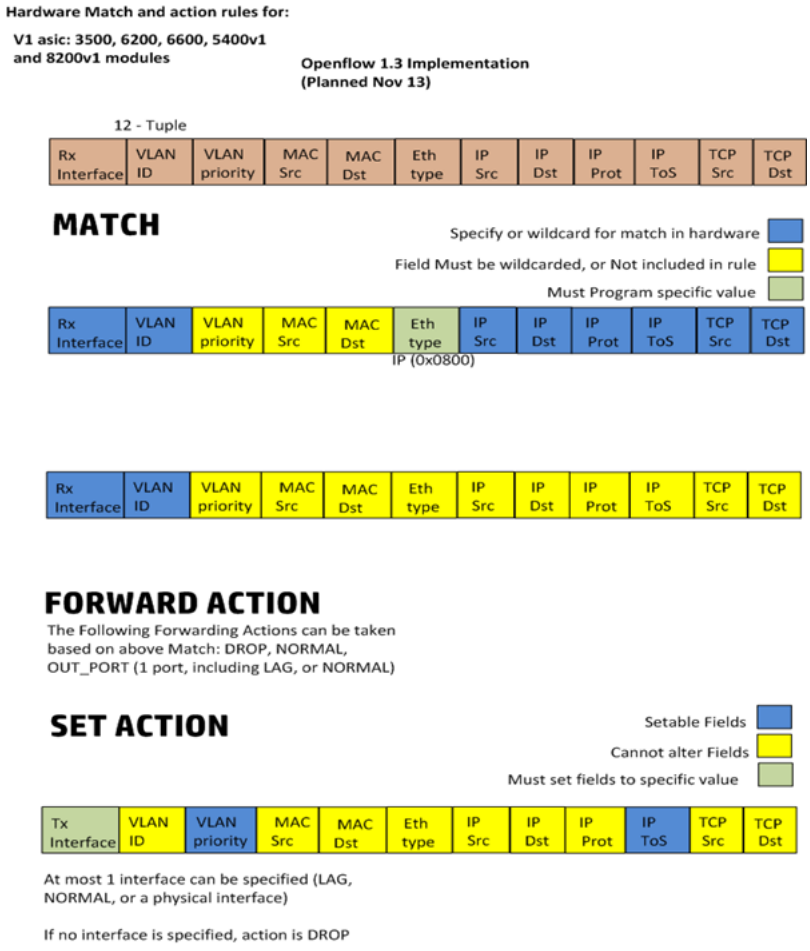
```
HP-Stack-3800(config-class)#
     OpenFlow
Message Type        Received          Rejected
----------------- ---------------   --------
OFPT_FLOW_MOD          0                  0
OFPT_PORT_MOD          0                  0
OFPT_GROUP_MOD         0                  0
OFPT_METER_MOD         0                  0
```

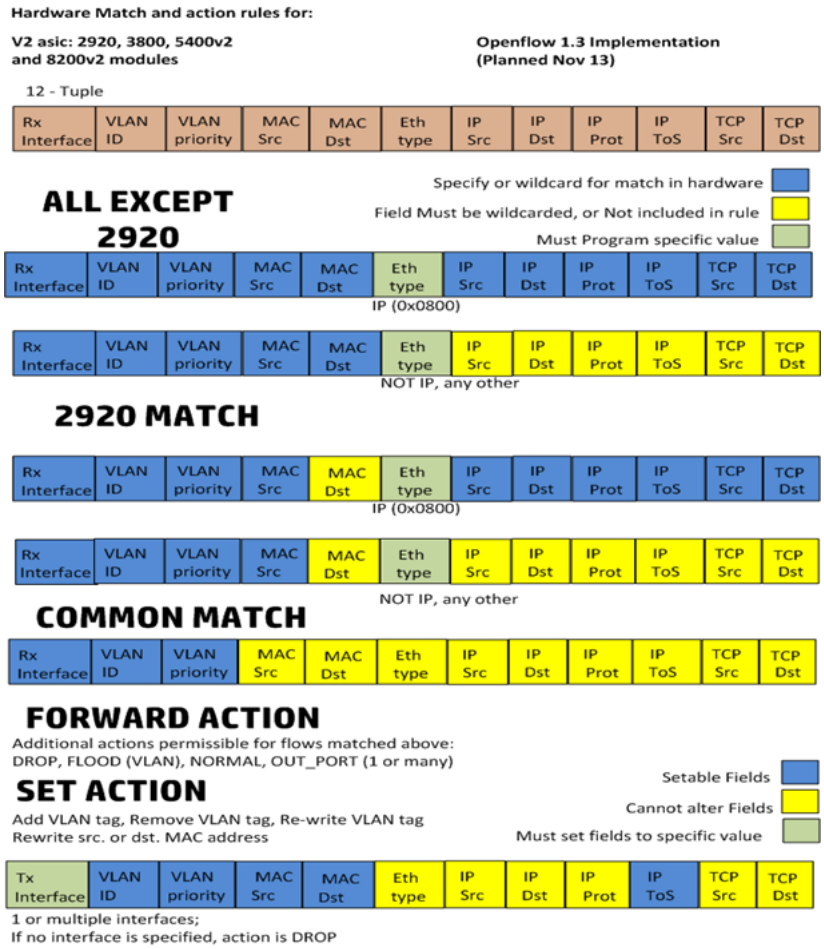
The flow in the switch can be verified by using the command `show openflowinstance [instance-name] flows`.

Rejection of the flow modification can be displayed by observing enable `debug openflow` results.

The same troubleshooting techniques can be used when issues arise with port modification, meter modification and group modification commands.

Review the Hardware Match tables in the figure below to review correct flows.






Following are the results of analyzing the Hardware Match tables with regard to the flow not being added/deleted/modified:

- Table 0 is read only: flow add/del/mod is not allowed in this table.
- Table 50 the flow add failed. The following errors are returned to the controller:

  OFPERR_OFPMFC_EPERM
  OFPERR_OFPFMFC_TABLE_FULL
  OFPERR_OFPBMC_BAD_WILDCARDS (Bad or unsupported match parameter in the flow)
  OFPERR_OFPBMC_BAD_VALUE (Bad value passed for match parameter in the flow)
  OFPERR_OFPFMFC_BAD_COMMAND
  OFPERR_OFPFMFC_UNKNOWN (Any internal system error)

  - Default rule (flow with match on any, with priority 0) for table 50 is read only, add/del/mod is not allowed.
  - Flows with priotiy 0 cannot be programmed.
  - srcip and dstIpare have to be passed in the flow.
  - srcIp/dstIpwith subnet mask values are not allowed with the exception of match ip-address.
  - Only IPv4/IPv6 traffic is allowed matched.
  - In table 50 a rule is not allowed with it is programmed with `v1, srcIP1, dstIP1` and the user tries to program `*, srcIP1, dstIP1` in virtual mode.
  - In table 50 a rule is not allowed with it is programmed with `v1, srcIP1, dstIP1, Priority1` and the user tries to program `v1, srcIP1, dstIP1, Priority2`.

- In Table 50 only unicast ip-address are allowed programming.
- Flow with invalid member VLAN is not allowed.
- Only `goto101` instruction is allowed in table 50.

- Table 100 the flow add failed. The following errors are returned to the controller:

  OFPERR_OFPMFC_EPERM
  OFPERR_OFPFMFC_TABLE_FULL
  OFPERR_OFPBMC_BAD_FIELD (Bad or unsupported match parameter in the flow)
  OFPERR_OFPBAC_BAD_TYPE (Bad or unsupported action in the flow)
  OFPERR_OFPBIC_BAD_TABLE_ID
  OFPERR_OFPFMFC_UNKNOWN (Any internal system error)

  - In table 100, 101, 102 aggregate mode output port action is allowed only if flow has matching criteria as a member-vlan or has modify-vlan specified in the action.
  - Match on VLAN-PCP is not allowed on v1 blades.
  - Match on non-ipflows is not allowed on v1 blades.
  - 12-tuple match for a flow is not allowed on v2 blades if switch is running in `ip-control-table` mode.
  - `Gototable 201/202` is not allowed.

# Reporting problems

If you are unable to solve a problem with OpenFlow, do the following:

1. Read the release notes for OpenFlow to see if the problem is known. If it is, follow the solution offered to solve the problem.
2. Determine whether the product is still under warranty or whether your company purchased support services for the product. Your operations manager can supply you with the necessary information.
3. Access HP Support Center and search the technical knowledge databases to determine if the problem you are experiencing has already been reported. The type of documentation and resources you have access to depend on your level of entitlement.

   **NOTE:** The HP Support Center at HP Support Center offers peer-to-peer support to solve problems and is free to users after registration.

   If this is a new problem or if you need additional help, log your problem with the HP Support Center, either on line through the support case manager at HP Support Center, or by calling HP Support. If your warranty has expired or if you do not have a valid support contract for your product, you can still obtain support services for a fee, based on the amount of time and material required to solve your problem.
4. If you are requested to supply any information pertaining to the problem, gather the necessary information and submit it. The following sections describe some of the information that you may be asked to submit.

# 7 Support and other resources

## Contacting HP

### Before you contact HP

Be sure to have the following information available before you call contact HP:

- Technical support registration number (if applicable)
- Product serial number
- Product model name and number
- Product identification number
- Applicable error message
- Add-on boards or hardware
- Third-party hardware or software
- Operating system type and revision level

### HP contact information

For the name of the nearest HP authorized reseller:

- See the Contact HP worldwide (in English) website at http://welcome.hp.com/country/us/en/wwcontact.html.

For HP technical support:

- In the United States, for contact options see the Contact HP United States website at http://www8.hp.com/us/en/home.html.

  To contact HP by phone:

  - Call 1-800-HP-INVENT (1-800-474-6836.) This service is available 24 hours a day, 7 days a week. For continuous quality improvement, calls may be recorded or monitored.
  - If you have purchased a Care Pack (service upgrade), call 1-800-633-3600. For more information about Care Packs, see the HP website at http://www.hp.com/hps.
  - In other locations, see the Contact HP worldwide (in English) website at http://www8.hp.com/us/en/contact-hp/contact.html.

### Subscription service

HP recommends that you register your product at the Subscriber's Choice for Business website at http://www.hp.com/country/us/en/contact_us.html. After registering, you will receive email notification of product enhancements, new driver versions, firmware updates, and other product resources.

## Documents

You can find additional switch documents by using the Manuals page of the HP Business Support Center website at http://pro-networking-h17007.external.hp.com/us/en/support/converter/index.aspx.

Additional documentation for your HP Switch may include:

- *Access Security Guide*
- *Advanced Traffic Management Guide*

- *Basic Operation Guide*
- *IPv6 Configuration Guide*
- *Management and Configuration Guide*
- *Multicast and Routing Guide*
- *Event Log Message Reference Guide*
- *Comware CLI Commands in ProVision Software*

## Websites

HP product websites are available for additional information.

- HP Switch Networking web site: http://www.hp.com/networking/support
- HP Technical Support website: http://www.hp.com/support

## Typographic conventions

This document uses the following typographical conventions:

`%`, `$`, or `#`
: A percent sign represents the C shell system prompt. A dollar sign represents the system prompt for the Bourne, Korn, and POSIX shells. A number sign represents the superuser prompt.

*audit*(5)
: A manpage. The manpage name is *audit*, and it is located in Section 5.

`Command`
: A command name or qualified command phrase.

`Computer output`
: Text displayed by the computer.

**Ctrl+x**
: A key sequence. A sequence such as **Ctrl+x** indicates that you must hold down the key labeled **Ctrl** while you press another key or mouse button.

`ENVIRONMENT VARIABLE`
: The name of an environment variable, for example, `PATH`.

ERROR NAME
: The name of an error, usually returned in the `errno` variable.

**Key**
: The name of a keyboard key. **Return** and **Enter** both refer to the same key.

Term
: The defined use of an important word or phrase.

**`User input`**
: Commands and other text that you type.

*`Variable`*
: The name of a placeholder in a command, function, or other syntax display that you replace with an actual value.

[ ]
: The contents are optional in syntax. If the contents are a list separated by `|`, you must choose one of the items.

{ }

The contents are required in syntax. If the contents are a list separated by |, you must choose one of the items.

...

The preceding element can be repeated an arbitrary number of times.

Indicates the continuation of a code example.

|

Separates items in a list of choices.

WARNING

A warning calls attention to important information that if not understood or followed will result in personal injury or nonrecoverable system problems.

CAUTION

A caution calls attention to important information that if not understood or followed will result in data loss, data corruption, or damage to hardware or software.

IMPORTANT

This alert provides essential information to explain a concept or to complete a task

NOTE

A note contains additional information to emphasize or supplement important points of the main text.

## Customer self repair

HP products are designed with many Customer Self Repair parts to minimize repair time and allow for greater flexibility in performing defective parts replacement. If during the diagnosis period HP (or HP service providers or service partners) identifies that the repair can be accomplished by the use of a Customer Self Repair part, HP will ship that part directly to you for replacement. There are two categories of Customer Self Repair parts:

- Mandatory—Parts for which Customer Self Repair is mandatory. If you request HP to replace these parts, you will be charged for the travel and labor costs of this service.
- Optional—Parts for which Customer Self Repair is optional. These parts are also designed for customer self repair. If, however, you require that HP replace them for you, there may or may not be additional charges, depending on the type of warranty service designated for your product.

**NOTE:** Some HP parts are not designed for Customer Self Repair. In order to satisfy the customer warranty, HP requires that an authorized service provider replace the part. These parts are identified as *No* in the Illustrated Parts Catalog.

Based on availability and where geography permits, Customer Self Repair parts will be shipped for next business day delivery. Same day or four-hour delivery may be offered at an additional charge where geography permits. If assistance is required, you can call the HP Technical Support Center and a technician will help you over the telephone. HP specifies in the materials shipped with a replacement Customer Self Repair part whether a defective part must be returned to HP. In cases where it is required to return the defective part to HP, you must ship the defective part back to HP within a defined period of time, normally five (5) business days. The defective part must be returned with the associated documentation in the provided shipping material. Failure to return the defective part may result in HP billing you for the replacement. With a Customer Self Repair, HP will pay all shipping and part return costs and determine the courier/carrier to be used.

For more information about the HP Customer Self Repair program, contact your local service provider. For the North American program, visit the HP website at http://www.hp.com/go/selfrepair.

# 8 Documentation feedback

HP is committed to providing documentation that meets your needs. To help us improve the documentation, send any errors, suggestions, or comments to Documentation Feedback (docsfeedback@hp.com.) Include the document title and part number, version number, or the URL when submitting your feedback.

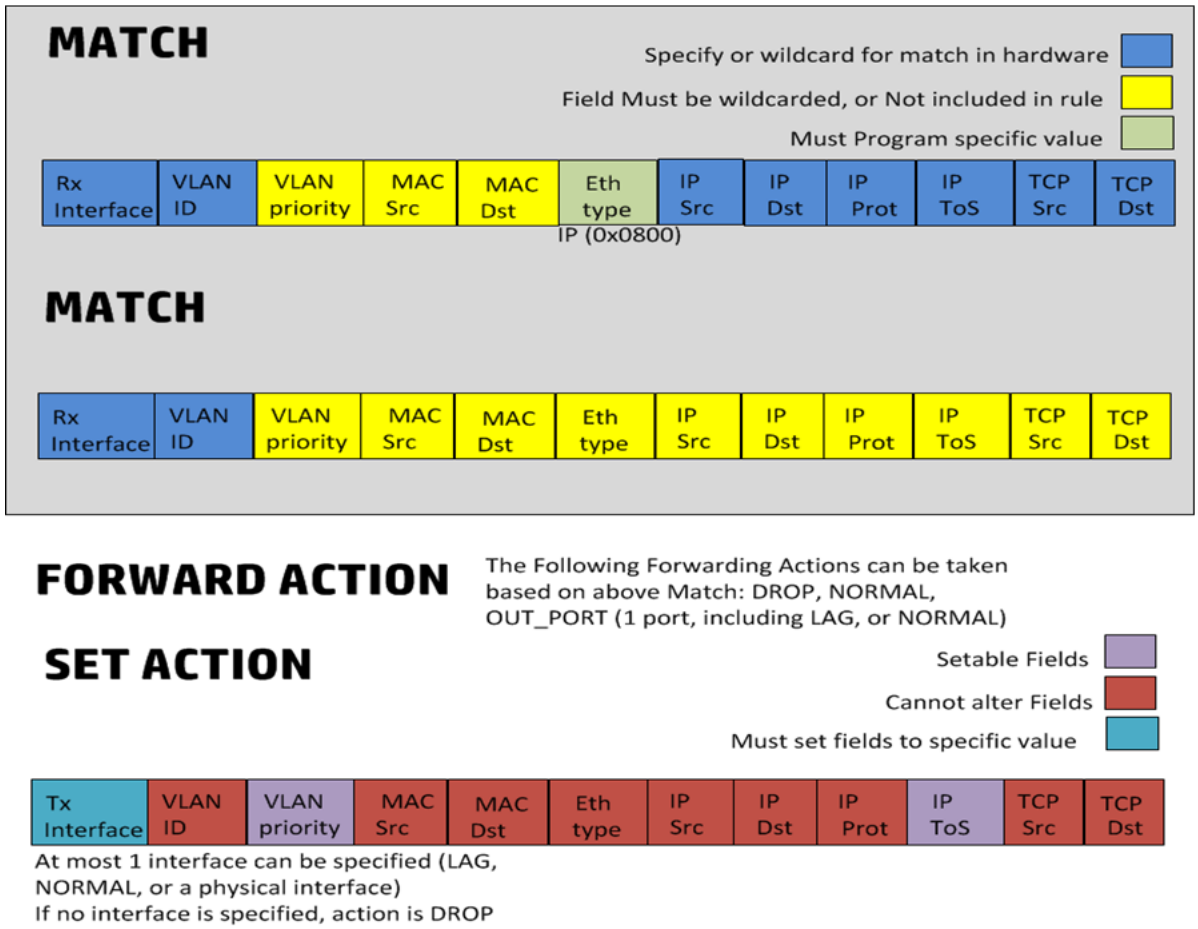
**NOTE:** There has been a change to the style of the documentation with the newest release. The "Command Table" commonly seen at the beginning of the chapters has been replaced with the ability to search for commands using the index. All commands are now listed in the index within the category "Command Syntax".

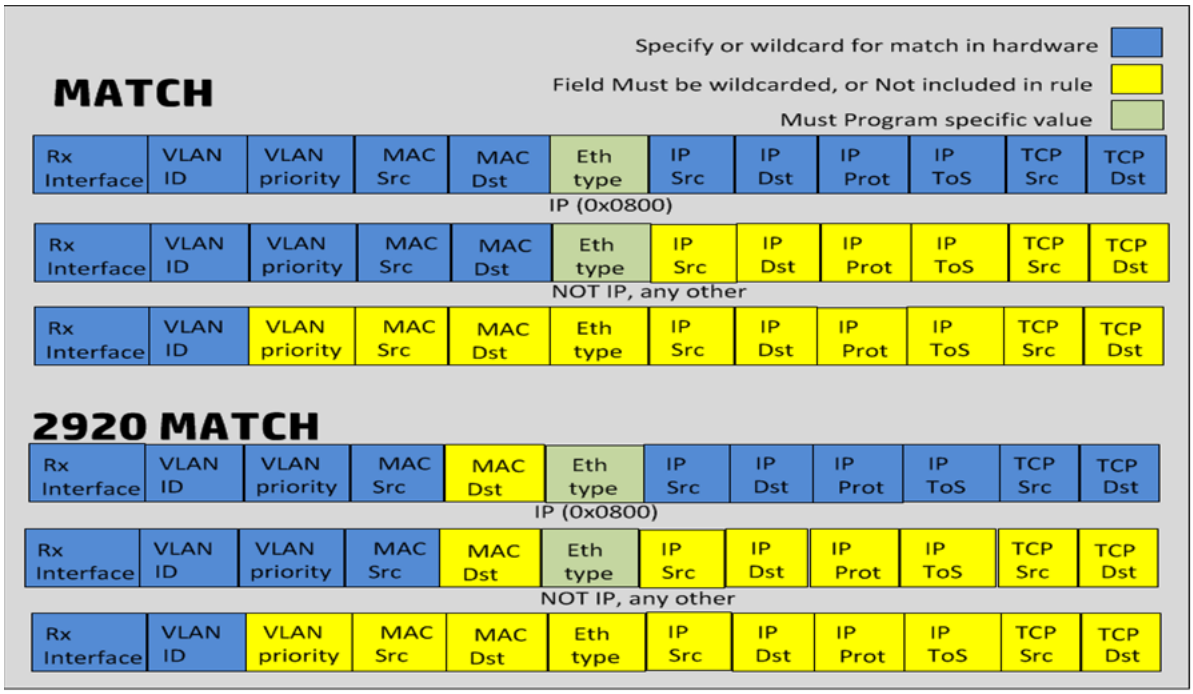
# A Flow classification on v1 and v2 modules

Hardware differences between v1 & v2 Modules affect flow match capabilities.

For additional information about v1 & v2 Modules, compatibility and inter-operation of v2 zl Modules with v1 zl Modules in a chassis switch, see the latest Release Notes for your switch in the Compatibility Mode section, and the *HP 8200 zl, 5400 zl, 3500, and 6200 yl Switch Series Technical Overview White Paper*, 4AA0-5388ENW.pdf available on the HP Switch Networking web site at http:\\www.hp.com/networking/support.

## Hardware match chart






**FORWARD ACTION** Additional actions permissible for flows matched above: DROP, FLOOD (VLAN), NORMAL, OUT_PORT (1 or many)

**SET ACTION**

Add VLAN tag, Remove VLAN tag, Re-write VLAN tag
Rewrite src. or dst. MAC address

Setable Fields
Cannot alter Fields
Must set fields to specific value

| Tx Interface | VLAN ID | VLAN priority | MAC Src | MAC Dst | Eth type | IP Src | IP Dst | IP Prot | IP ToS | TCP Src | TCP Dst |
|---|---|---|---|---|---|---|---|---|---|---|---|

1 or multiple interfaces;
If no interface is specified, action is DROP

# B Implementation Notes

This section documents some of the behaviors exhibited during the implementation of OpenFlow. These behaviors were exposed during testing and may include unit, conformance, integration, interoperability, stress and system testing.

## A hardware flow with an idle timeout of 10 seconds gets deleted even though packets match the flow within the idle timeout

### Problem statement

A hardware rule is programmed with idle timeout as 10 seconds and hard timeout as 0. Packets are pumped at 1000 pps to the switch matching the flow. However, after 10 seconds, the rule gets removed from the switch.

### Reason for this behavior

By default the hardware statistics refresh rate (set using `openflow hardware statistics refresh rate` and information available through `show openflow`) is 20 seconds. This means that the packet count statistics get updated only every 20 seconds. So, when the idle timeout is set to less than 20 seconds, when a check is done for flow statistics after 10 seconds, it would not be updated. Hence, the flow would get deleted.

### Customer Note

The user has the option of reducing or incriminating the refresh rate. However, the user needs to be aware of its implications. An increase in refresh rate would lead to deletion of flows which has an idle timeout less than the configured refresh rate. A decrease in refresh rate would lead to over-use of CPU (because of polling hardware statistics more frequently.)

## Controller flows — flow in hardware and processing software

Flows with an action to send matching traffic to controller are actually installed on hardware. But, the actual traffic forwarding takes place in software as we need to add the required OpenFlow specific headers. Due to this characteristic, the actual forwarding will not take place at the line rate. A sample controller flow would look like:

### Example

In this example, any packet that comes on port A1, will be forwarded to the controller after adding required OpenFlow packet headers (as the packet will be sent as a packet_in) to the controller. Since this processing is done on software, we will not be able to send the incoming traffic at line rate.

```
HP-8206zl(openflow)# show openflow instance test flows
Flow 1
     Incoming Port     : A1       Ethernet Type        : 0x0000
     Source MAC        : 000000-000000  Destination MAC       :
000000-000000
     VLAN ID           : 0        VLAN priority         :
     Source IP         : 0.0.0.0    Destination IP        : 0.0.0.0

     IP Protocol      : 0        IP ToS Bits           : 0
     Source Port       : 0       Destination Port      : 0
     Priority          : 2
     Duration          : 1 seconds
     Idle Timeout      : 60 seconds    Hard Timeout         : 0
seconds
     Packet Count      : 1       Byte Count            : 0
```

```
  Flow Location    : Hardware
Actions
  Controller Port
```

## DUT matches and processes incoming untagged packets for VLAN id

For certain flows with a match on the VLAN ID, even untagged packets are matched. This happens on untagged ports only. The existing behavior exists because L2 hardware adds the VLAN id and VLAN priority meta-information irrespective of whether the packet came in tagged or untagged. Flows which can be accelerated into hardware are put into hardware whereas flows which cannot be accelerated in hardware are put into software. The observed behavior is observed for hardware flows. For software flows, the match happens for packets which come with a VLAN tag only and with proper VLAN id.

## Events that change the Operational Status of the OpenFlow instance

The `Oper. Status` field indicates the operational status of the instance and can be either up or down. The operational status will be down when either the member VLAN of the OpenFlow instance does not exist on the switch or the controller VLAN of the OpenFlow instance does not exist on the switch. In the case when multiple controllers connect over multiple controller VLANs, the operational status will be down when none of the controller VLANs exist on the switch. When the member VLAN is down - all ports on the member VLAN are down.

For example, the `show openflow instance` displays all the OpenFlow instance related information as follows:

`show openflow instance <test>`

**NOTE:** Note that for purposes of this example the instance *<test>* has been created.

```
Instance Name             : Test
Admin. Status             : Enabled
Member List               : VLAN 3
Listen Port               : 6633
Oper. Status              : Down
Datapath ID               : 00030026f1212000
Mode                      : Active
Flow Location             : Hardware and Software
No. of Hw Flows           : 0
No. of Sw Flows           : 0
Hw. Rate Limit            : 0 kbps
Sw. Rate Limit            : 100 pps
Conn. Interrupt Mode      : Fail-Secure
Maximum Backoff Interval  : 60 seconds

Controller Id Connection Status Connection State
------------- ----------------- ----------------
2             Disconnected      Backoff
```

## OpenFlow's influence on CPU generated packets

In some cases, the CPU generated packets will be effected by the TCAM rules. OpenFlow Specification 1.0.0. does not clearly outline the behavior for CPU generated packets. One example of such a case is when a rule is in place with the `in_port`as a wild-card but has a SRC IP address that matches the IP address configured on the switch.

## OpenFlow supports IP address masking

OpenFlow supports IP subnet mask. Controllers can specify the subnet mask associated with an IP address and sent to the OpenFlow switch. The switch accepts the IP address with the subnet mask and associates any packets coming with the subnet mask with the rule.

For example the K.15.10. OpenFlow implementation supports the ability to match on IP address and subnet mask when the OpenFlow controller programs such flows. Consider this example where the ovs-ofctl utility is used to add a flow that matches on a network source address of 1.1.1.1 with a subnet mask of /24. Note that 10.10.10.1 here is the IP address of the switch which has an OpenFlow listen port open on port 6633.

```
openflow@openflow-ubuntu-08:~$ ovs-ofctl add-flow
tcp:10.10.0.1:6633 ip,nw_src=1.1.1.1/24,actions=output:1
```

To verify that this flow has been installed on the switch, we run the `ovs-ofctl` command and verify the output.

```
openflow@openflow-ubuntu-08:~$ ovs-ofctl dump-flows tcp:10.10.0.1:6633
NXST_FLOW reply (xid=0x4): cookie=0x0, duration=13.535s, table=0,
n_packets=0, n_bytes=0, ip,nw_src=1.1.0.0/24 actions=output:1
```

The `show openflow instance t1 flows` command when executed on the HP switch displays the following:

### Example

```
HP-3500yl-48G-PoEP(of-inst-t1)# show openflow instance t1
Configured OF Version    : 1.3
Negotiated OF Version    : 1.3
Instance Name            : t1
Admin. Status            : Enabled
Member List              : VLAN 3
Listen Port              : None
Oper. Status             : Up
Oper. Status Reason      : NA
Datapath ID              : 0003b499ba86bf80
Mode                     : Active
Flow Location            : Hardware and Software
No. of Hw Flows          : 0
No. of Sw Flows          : 0
Hw. Rate Limit           : 0 kbps
Sw. Rate Limit           : 100 pps
Conn. Interrupt Mode     : Fail-Secure
Maximum Backoff Interval : 60 seconds
Probe Interval           : 10 seconds
Hw. Table Miss Count     : NA
No. of Sw Flow Tables    : 1
Egress Only Ports        : None
Table Model              : Policy Engine and Software

Controller Id Connection Status Connection State Secure Role
------------- ----------------- ---------------- ------ ------
1               Connected           Active             Yes     Equal
```

## Virtualization mode verses Aggregation mode — VLAN tags in `packet_in` messages

There is a difference in the `packet_in` messages that are sent to the OpenFlow controller by the switch based on the mode that the OpenFlow instance is operating in. In Virtualization mode, no VLAN tags are sent in `packet_in` messages sent to the OpenFlow controller. Even if the packets that came in to the switch on the OpenFlow instance had VLAN tags, they will get removed by the switch in `packet_in` messages sent to the controller. Flows that match on VLAN PCP or modify VLAN PCP are not supported in Virtualization mode. Any tagged packets that are received in Virtualization mode may have their PCP modified to default. VLAN PCP isn't matched because tag is always stripped in Virtualization mode.

In Aggregate mode, VLAN tags are always sent by the switch in `packet_in` messages sent to the OpenFlow controller. Even if the packets that came in to the switch on the OpenFlow instance

did not have VLAN tags, they will be added by the switch in `packet_in` messages sent to the controller. The switch adds a VLAN tag either based on the tag that the packet already carried when it came in to the switch or based on the membership of the port that the packet came in to the switch.

## Precedence level in meters

As per the OpenFlow specification 1.3.1, the prec_level given in the ofp_meter_band_dscp_remark indicates by what amount the DSCP value in the packets should be incremented if the packets exceed the band. However, the switch implementation directly replaces the DSCP value in the IP packets with the prec_level when the band exceeds the meter defined by the controller

## Support for miss_len field in 'Switch Configuration' messages

The switch implementation does not honor the miss_len miss_send_len field specified in the packet-in Switch configuration messages. This is because, switch doesn't buffer packets. Due to this controller will see the entire packet copied in packet-in message with buffer_id set as OFP_NO_BUFFER.

# C Configuring OpenFlow switch to HP VAN SDN controller

HP Switches running OpenFlow can securely connect to HP VAN SDN controller.

Follow the procedures to accomplish the secure connection.

1. On the HP Switch running OpenFlow, create a **crypto profile**.

   ### Syntax

   ```
   crypto pki ta-profile flareProfile
   ```

2. Copy root certificate to the HP switch using this command:

   ### Syntax

   ```
   copy tftp ta-certificate flareProfile [103.0.11.34]
   HpRoot.pem
   ```

3. Create an identity profile on the HP switch using this command:

   ### Syntax

   ```
   crypto pki identity-profile flareIdentity subject
   [common-name]
   ```

4. Make a certificate signing request.

   ### Syntax

   ```
   crypto pki create-csr certificate-name flarecert ta-profile
   flareProfile usage openflow
   ```

5. Copy the CSR request text in step 4 and paste to a file named “switch.csr”
6. Execute the command:

   ### Syntax

   ```
   ./signCSR.pl-in switch.csr-out switch-l 2
   ```

7. Execute the command:

   ### Syntax

   ```
   crypto pki install signed certificate
   ```

8. Copy and paste the contents of switch.pem into the HP switch console.
9. Configure OpenFlow for Flare.

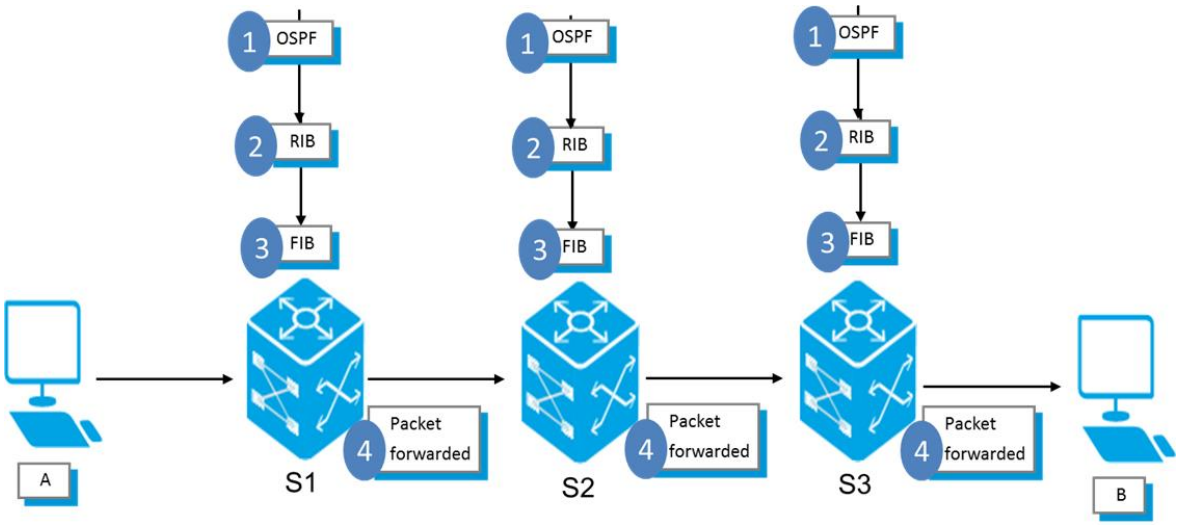
```
openflow
controller id 3 ip 103.0.11.31 port 6634 controller interface vlan 1 instance "flare"
member vlan 100
controller id 3 secure
version 1.3
limit hardware rate 10000000
limit software rate 10000
enable
exit
enable
```

# D Training Materials

## Traditional method: Control and data planes

In a traditional network environment, distributed routing protocols update local control planes on each device. Devices in the network are running a complicated distributed algorithm (much like the Dijkstra's SPF algorithm for OSPF) for a specific function. This cannot be easily changed. The algorithm would have to be redesigned if changes in function are required. Routing protocols (or static configuration) populate the RIB. The RIB in turn populates the FIB. Forwarding decisions are made based on entries in the FIB.

**Figure 7 Traditional control and data planes**

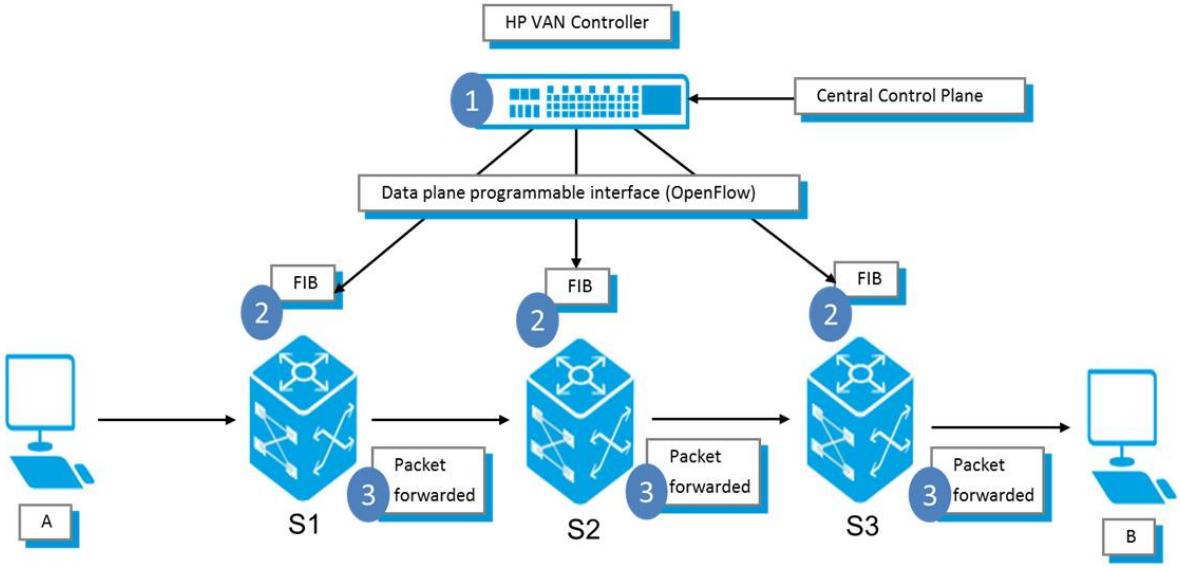



## New Paradigm: Control and data plane

In the new paradigm. an OpenFlow router or switch separates the control and data functions. The data path portion still resides on the switch, while high-level routing decisions are moved to a separate controller, in this case the HP SDN controller. Network devices thus perform packet and frame forwarding and maintenance functions such as checking if interfaces are up or down.

The OpenFlow router or switch and controller communicate via the OpenFlow protocol, which defines messages, such as packet-received, send-packet-out, modify-forwarding-table, and get-stats. The OpenFlow controller could be a separate appliance or virtual machine.

**Figure 8 OpenFlow control and data planes**



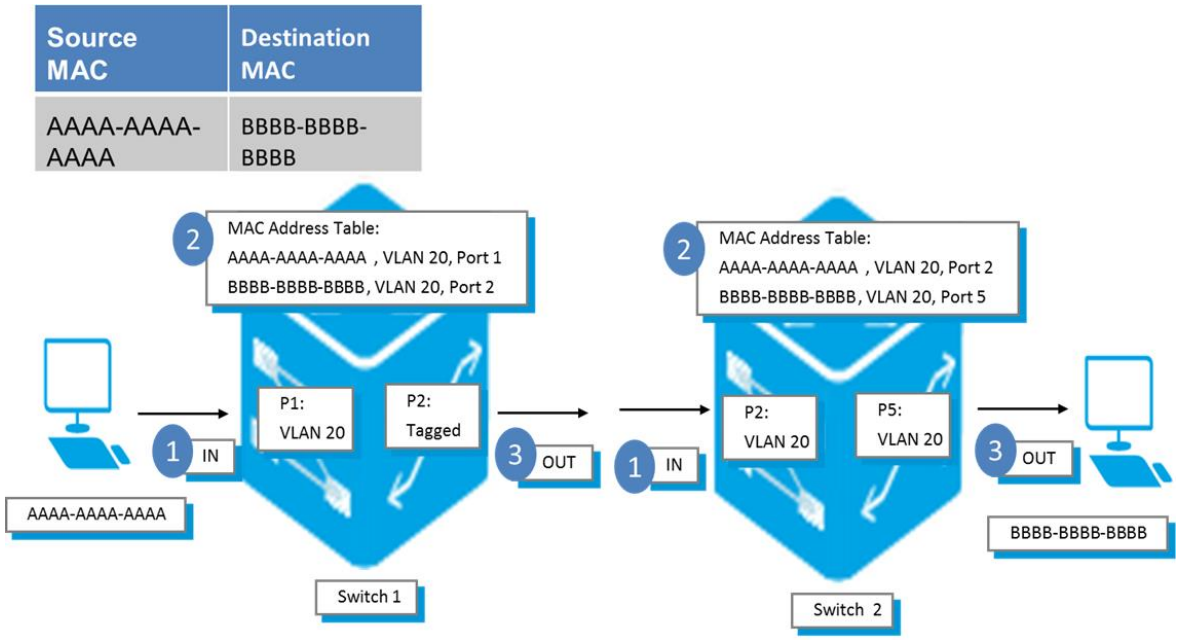

# Traditional switching

In a traditional Layer 2 switching environment, switching is performed based on the destination MAC address. Each switch has it's own MAC address table and each switch learns where the devices are. The process works as follows:

1. A frame arrives at switch 1 from PC A (MAC = AAAA.AAAA.AAAA) to PC B (MAC = BBBB.BBBB.BBBB)
2. The MAC address table is checked for the location of PC B
3. The entry is found in the forwarding table
4. The frame is transmitted out of port 2

This process is repeated for every switch in the network. A router would use a similar process based on destination IP address (unicast routing), a routing table (RIB), and forwarding information based (FIB).

**Figure 9 Traditional switching environment**

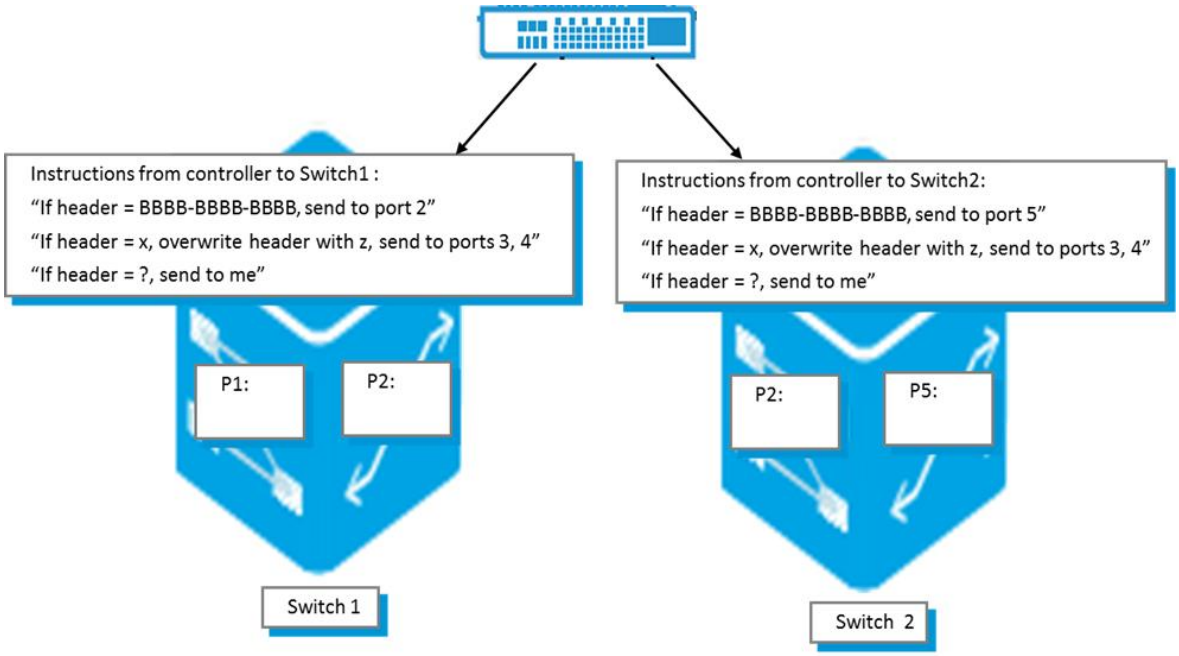


# Flow switching

In an OpenFlow environment, flow tables are used by devices rather than routing or MAC address tables. Each flow entry has an action associated with it. The three basic actions (that all dedicated OpenFlow switches must support) are:

- The first options is to forward this flow's packets to a given port (or set of ports). This allows packets to be switched through the network. In most switches this is expected to take place at line rate speeds.
- The second option is to encapsulate the packet and forward this flow's packets to the SDN controller. The packet is delivered via a secure channel using TLS. The controller makes a decision and forwards the packet back to the switch. Typically, this method is only used for the first packet in a new flow, so a controller can decide if the flow should be added to the Flow Table. Or in some cases, it could be used to forward all packets to a controller for processing.
- The third option is to drop this flow's packets. This can be used for security reasons to block unauthorized traffic, or to stop denial of service attacks, or to reduce spurious broadcast traffic from end-hosts. HP's Sentinal application can be used for this purpose.

**Figure 10 Flow switching environment**

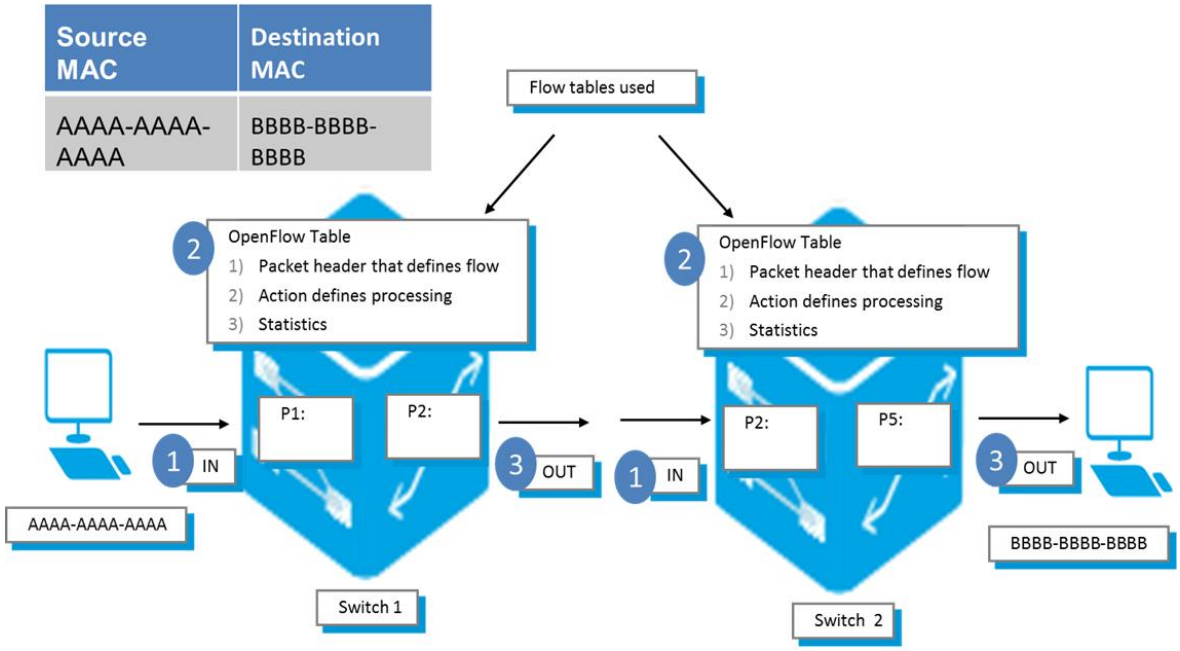



# Switching decisions made on flow table

An entry in the flow table has three fields:

- A packet header which defines the flow (for example, TCP port 80 traffic)
- The action which defines how the packets should be processed (forward out of port G1/0/1)
- Statistics which keep track of the number of packets and bytes for each flow (for example, 100 packets and 8000 bytes). The time since the last packet matched the flow is also recorded so as to remove inactive flows. This can be configured within the HP SDN Controller. The default is a flow that is active for 300 seconds.

**Figure 11 Switching decisions made on flow table**




## Solutions for control plane

The control plane computes the forwarding state while meeting these requirements:

- **Problem 1: Compatible with low-level hardware / software devices**
- Problem 2: Ability to make decisions based on the entire network topology
- Problem 3: Need to configure all switches and routers

The solution to these problems is abstraction. Abstraction of the forwarding model hides the complexity of the implementation. Rather than trying to change the operation system code used by network devices, you use an open interface to tell the network devices what to do. This means that there is no longer any concern with regards to specific vendor, or specific device, or specific ASICs used.

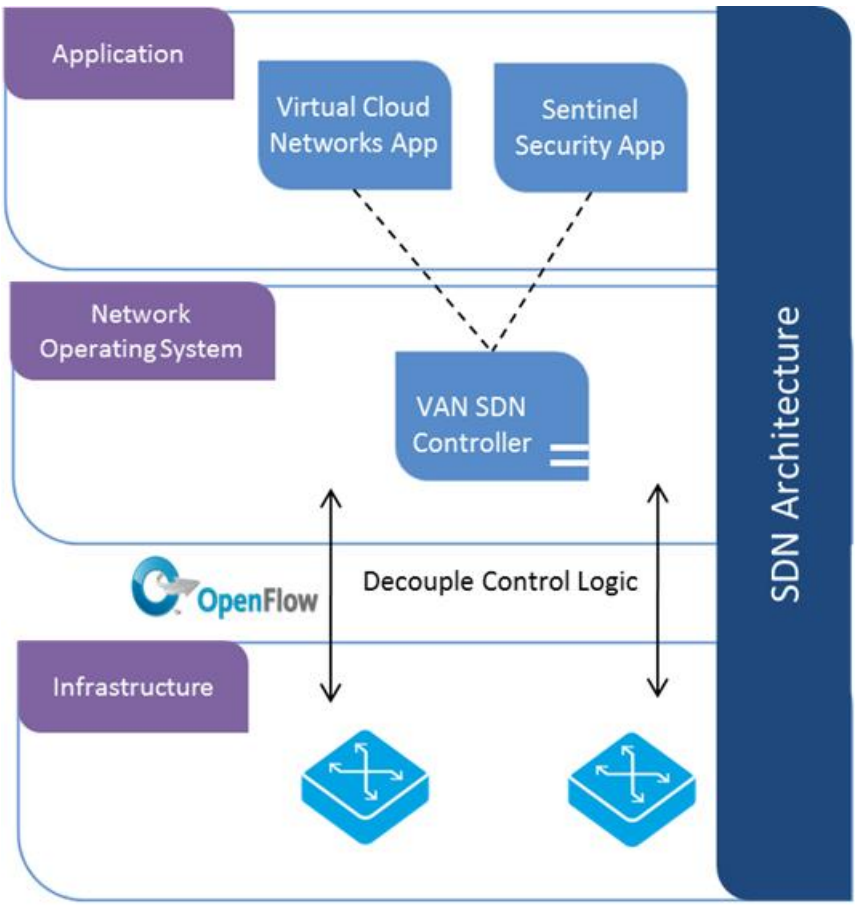
OpenFlow is the interface or protocol that instructs networking devices about what to do with traffic. So, for example, if the packet has a destination of 10.1.1.1, flows within the network device are configured to drop the packet. If a packet has a destination of 10.1.2.1, flows are created within the device to forward the packet out of port GE1/0/1. The configuration is performed based on flow entries <header, action>, rather than routing table entries. To clarify, OpenFlow isn't SDN, but is rather a protocol that allows abstraction of hardware. SDN can use the OpenFlow protocol, but doesn't have to. SDN can make use of other protocols instead of using OpenFlow. SDN is a generic term explaining the separation of the control plane and the data plane in networking.

Every network device has one thing in common whether that device is a switch, a router, a firewall, a load balancer, a WAN optimization device or an intrusion protection device. Devices receive a packet (packet in), they look at it, they inspect it using the header field or maybe some user data field, and they perform a function. The function can be forwarding in terms of switch or router based on Layer 2 and Layer 3 information, it can be load balancing, or dropping the packet in the case of a firewall. It could be a multicast feature where you move or forward the packet on multiple ports based on the variety of group settings.

All of these are functions that are performed by a traditional networking devices. What SDN does is to separate that function into a central control layer and a distributed forwarding layer. In the same way as wireless has moved to a central controller (or teamed controllers) and controlled (thin/lightweight) access points, SDN suggests using a central controller and controlled network devices. This allows for reduced complexity and configuration on the network devices and therefore management overhead because these functions are managed from a central location.

OpenFlow tries to solve this problem in an open standards way. Unfortunately in the wireless industry even though the ideal solution is a standardized communication protocol between controllers and APs, vendors have gone in different directions so you don't have that level of interoperability. We are at the same stage with SDN. The industry is trying to use a standards based model to link the controller through a protocol called OpenFlow which is defined by the ONF. But it remains to be seen if vendors come out with their own implementation of SDN. Will they stay with the standards based protocol OpenFlow or will they develop their own proprietary protocols? This is to be seen, but does pose a threat to SDN.

**Figure 12 Abstraction of the forwarding model used by devices**

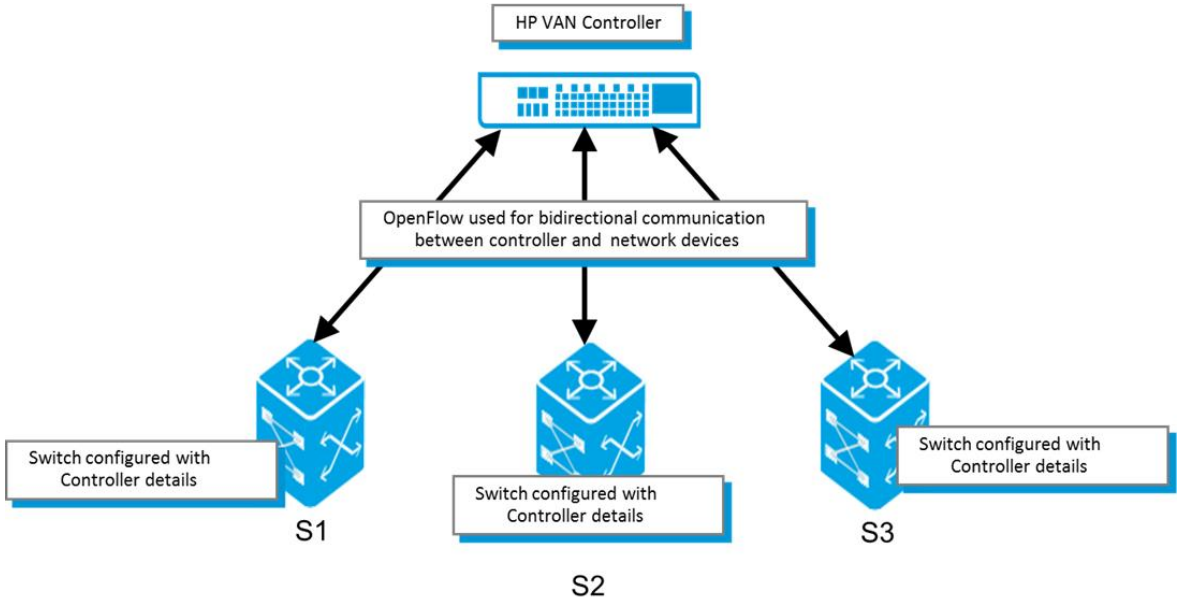



## OpenFlow communication

OpenFlow is used for communication between the network devices and the controller. This uses TCP port default port 6633. In OpenFlow 1.0, this communication was in clear text. In OpenFlow 1.3 this communication is encrypted using TLS. OpenFlow version 1.3 is available in the HP switch software release 15.14.

An out-of-band channel is recommended. On HP switches for example, a separate non-OpenFlow VLAN needs to be configured for communication between the switches and the SDN Controller.

Figure 13 OpenFlow communication

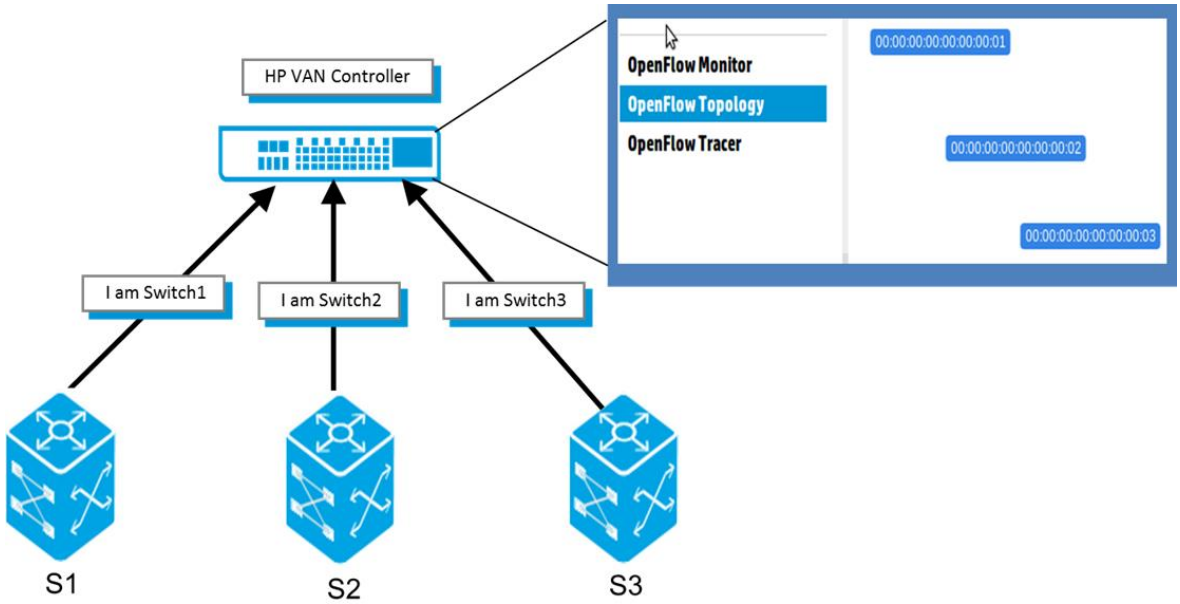



## OpenFlow switch discovery

When switches running OpenFlow start up, they contact the SDN controller using the defined IP address and default OpenFlow port of 6633. The controller can thus discover network devices in the topology and build a model of the network topology.

When the network devices establish a TCP session with the controller, the controller sends a feature request message to the device. The device returns information about features supported such as a list of ports, flow_stats, table_stats, port_stats, and so forth. The controller is thus able to discover the ports on individual switches. At this point, however, the controller doesn't know how devices are connected.

Figure 14 OpenFlow switch discovery




## OpenFlow link discovery

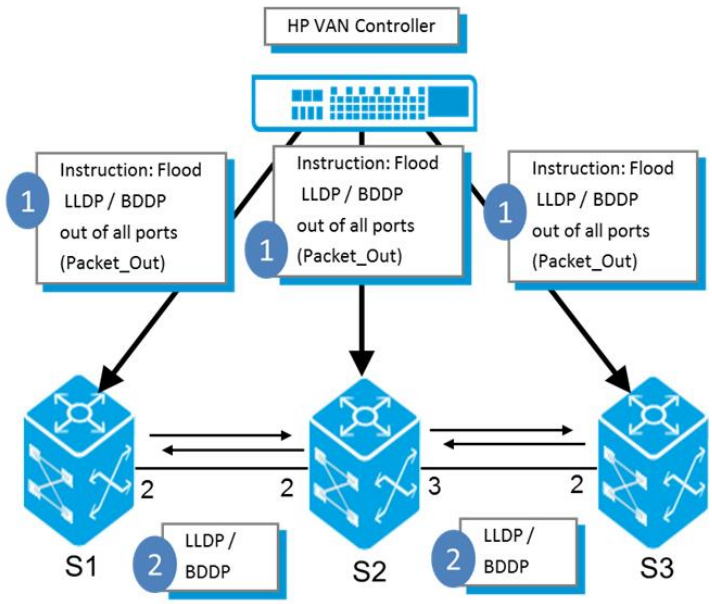
The controller instructs switches to send Link Layer Discovery Protocol (LLDP) and Broadcast Domain Discovery Protocol (BDDP) packets to discover neighboring devices. BDDP is essentially a broadcast version of LLDP allowing for discovering neighboring OpenFlow switches through a non-OpenFlow

switch. The LLDP and BDDP messages contain the Data Path Identifier (DPID) of each switch, the controller-ID, and the port number the message originated from. This allows the HP SDN controller to discovery how devices in the network are connected.

The controller instructs switches to send packets out using the Packet_Out message. Packet_Out allows the controller to specify the type of packet the switch should send and out of which ports.

For now there is no differentiation between LAG and non-LAG ports. This is because the OpenFlow specification does not provide for this.

**Figure 15 OpenFlow link discovery**

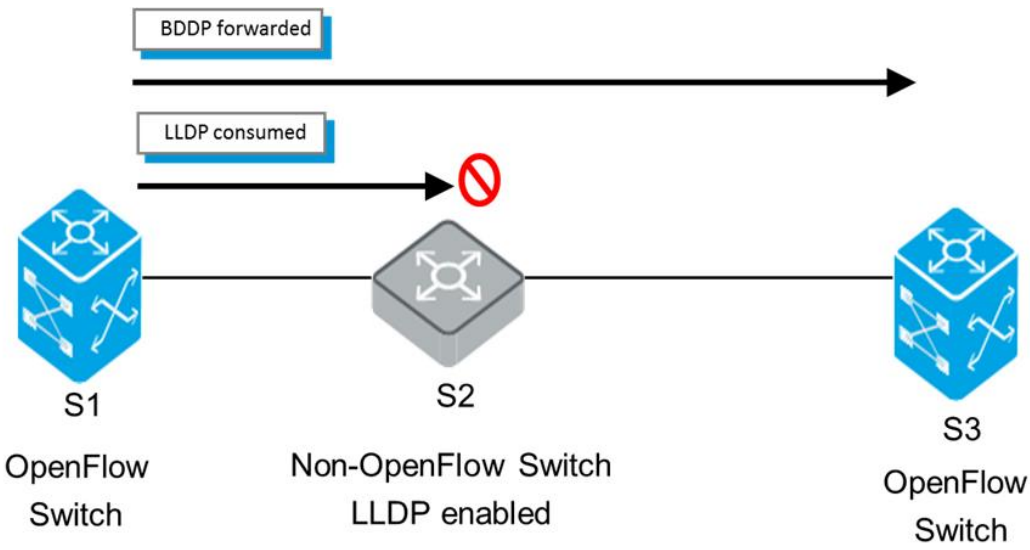



## OpenFlow BDDP

LLDP is used to discover other OpenFlow enabled switches that are directly connected. However, if two OpenFlow switches are separated by a non-OpenFlow switch, this process may fail. The intermediate switch may also be running LLDP and is thus subscribed to the LLDP multicast. In that case, the LLDP message from one OpenFlow switch is not seen by the other OpenFlow switch.

However, a broadcast frame (BDDP) will typically be forwarded by the intermediate switch within the same broadcast domain. BDDP has the same TLV information as LLDP.

**Figure 16 OpenFlow BDDP**

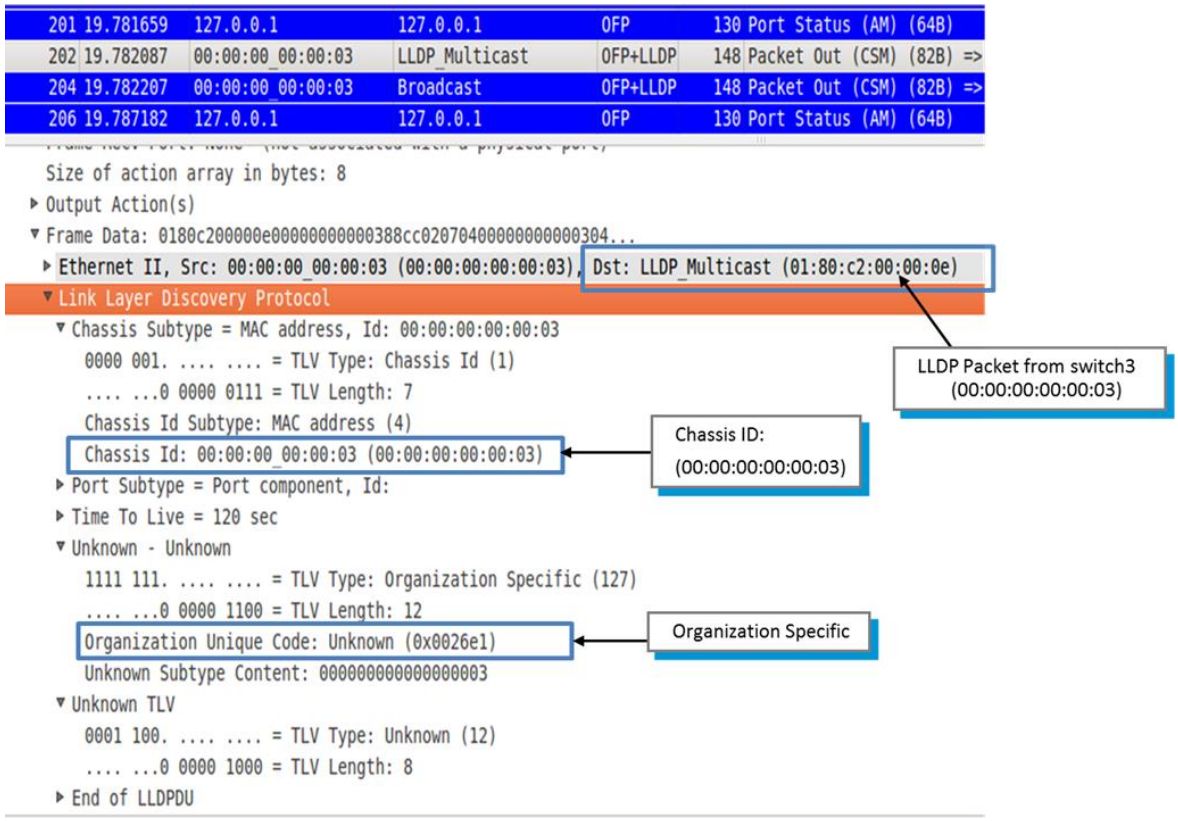


# LLDP message format

The figure below shows an LLDP message from switch 3. Additional TLV information in the LLDP packet is also shown:

- Switch Chassis_ID = DPID (Data Path ID)
- Sys_Descr=Controller ID (Organizational specific TLV). In a teaming environment, Sys_Decr is set to the team ID.

**Figure 17 LLDP message format**

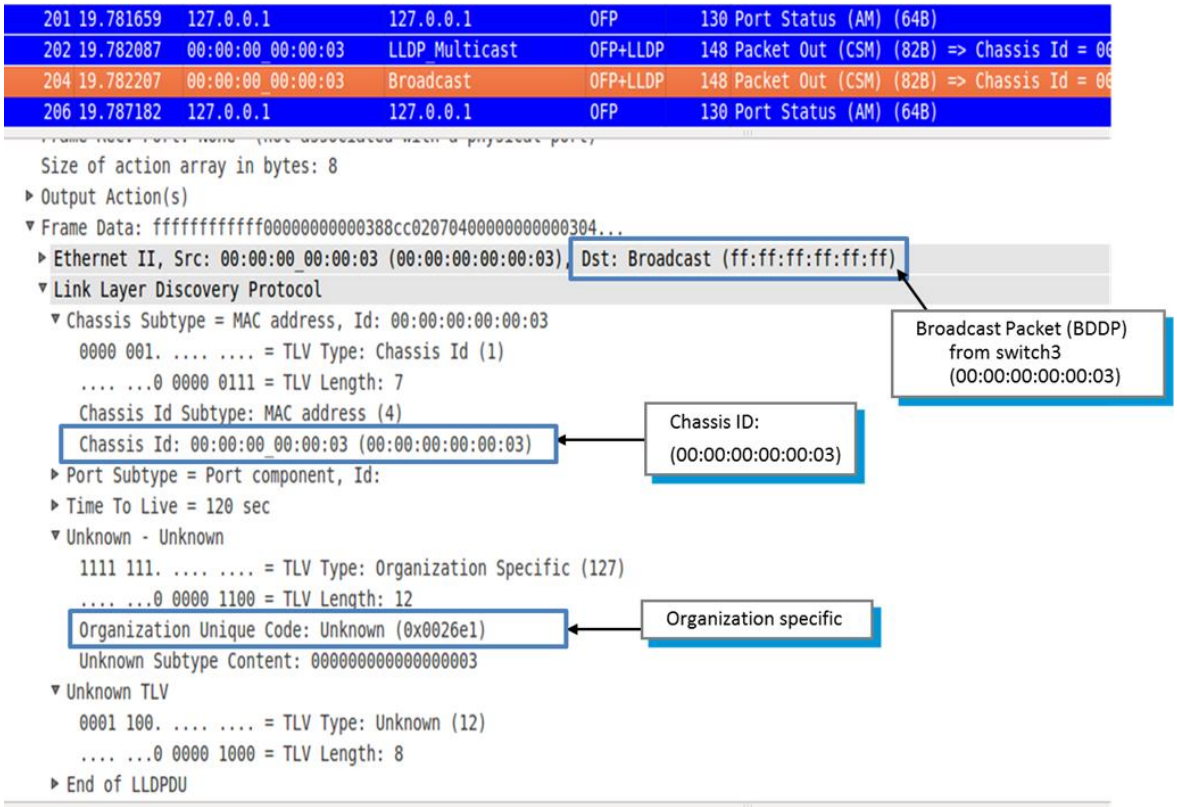



# BDDP message format

The figure below shows a BDDP message from switch 3. Note that the destination address is a broadcast rather than multicast address. Additional TLV information in the BDDP packet is also shown:

- Switch Chassis_ID = DPID (Data Path ID)
- Sys_Descr=Controller ID (Organizational specific TLV). In a teaming environment, Sys_Decr is set to the team ID.

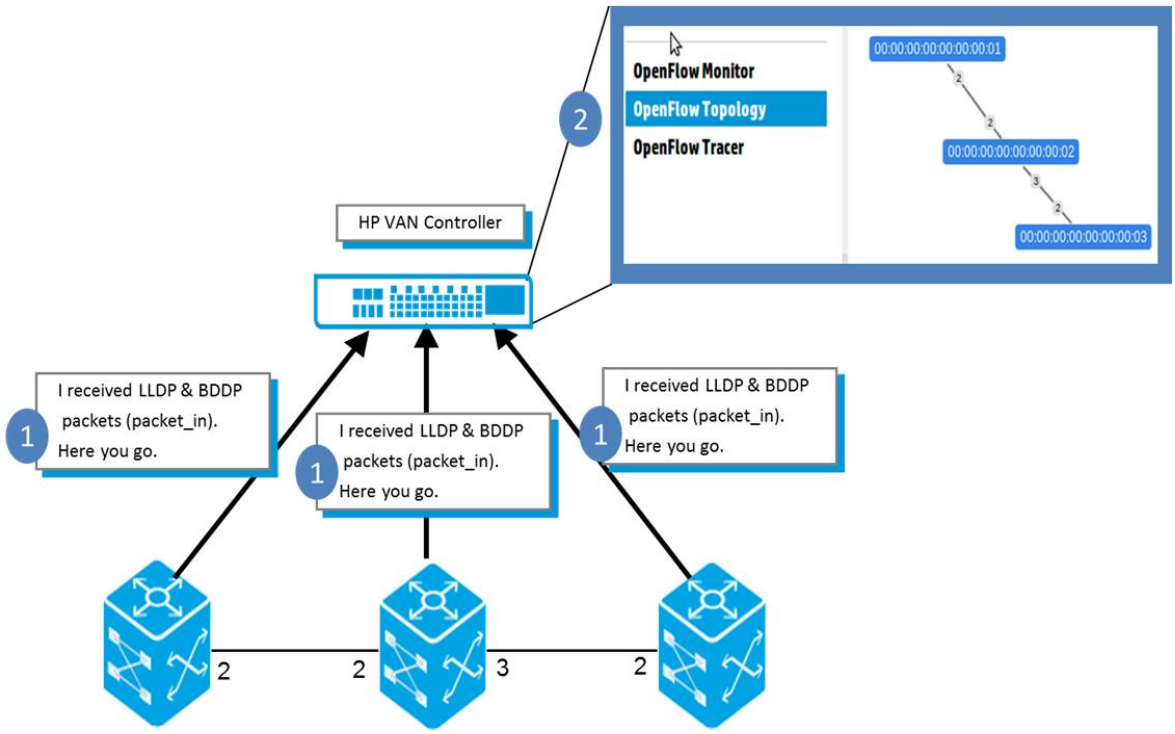
### Figure 18 BDDP message format




# OpenFlow link discovery

Network devices forward the received LLDP and BDDP messages to the controller since they have no local intelligence and don't know what to do with the packet. The controller works out how the devices are connected and then builds a topology diagram.

### Figure 19 OpenFlow link discovery



# Index

## W