编 程 学 习 笔 记

数据中心交换机

Data Center Switch

作 者 姓 名： 颜佳

目 录

[1 命令行格式约定 1](#_Toc80554933)

[2 配置设备的管理地址和路由 1](#_Toc80554934)

[3 路由协议视图 2](#_Toc80554935)

[4 配置生效模式 2](#_Toc80554936)

[4.1 立即生效模式 2](#_Toc80554937)

[4.2 两阶段生效模式 2](#_Toc80554938)

[5 命令行智能回退功能 2](#_Toc80554939)

[6 用户界面 3](#_Toc80554940)

[6.1 Console用户界面 3](#_Toc80554941)

[6.2 VTY用户界面 3](#_Toc80554942)

[7 用户界面的用户验证 3](#_Toc80554943)

[8 接口分类 4](#_Toc80554944)

[8.1 物理接口 4](#_Toc80554945)

[8.1.1 管理接口 4](#_Toc80554946)

[8.1.2 业务接口 4](#_Toc80554947)

[8.2 逻辑接口 4](#_Toc80554948)

[9 接口配置 5](#_Toc80554949)

[9.1 基础配置 5](#_Toc80554950)

[9.1.1 配置接口的描述信息 5](#_Toc80554951)

[9.1.2 配置接口的MTU值 5](#_Toc80554952)

[9.1.3 配置流量统计时间间隔 6](#_Toc80554953)

[9.1.4 配置开启或关闭接口 6](#_Toc80554954)

[9.2 以太网接口配置 6](#_Toc80554955)

[9.2.1 配置以太网接口的自协商功能 6](#_Toc80554956)

[9.2.2 配置以太网接口的二三层模式切换 6](#_Toc80554957)

[9.2.3 配置以太网光/电接口的接口速率 7](#_Toc80554958)

[9.2.4 配置以太网接口拆分 8](#_Toc80554959)

[9.3 逻辑接口配置 9](#_Toc80554960)

[9.4 端口隔离配置 9](#_Toc80554961)

[10 MAC配置 9](#_Toc80554962)

[10.1 MAC简介 9](#_Toc80554963)

[10.2 MAC地址表 10](#_Toc80554964)

[10.3 配置静态MAC表项 10](#_Toc80554965)

[10.4 配置黑洞MAC表项 10](#_Toc80554966)

[10.5 配置动态MAC表项的老化时间 10](#_Toc80554967)

[10.6 配置关闭MAC地址学习功能 11](#_Toc80554968)

[10.7 配置MAC地址限制 11](#_Toc80554969)

[10.8 配置MAC地址防漂移和漂移检测 11](#_Toc80554970)

[10.9 配置三层逻辑接口的MAC地址 11](#_Toc80554971)

[10.10 配置MAC刷新ARP功能 11](#_Toc80554972)

[10.11 配置端口桥功能 11](#_Toc80554973)

[10.12 配置丢弃全零MAC地址报文 11](#_Toc80554974)

[10.13 配置丢弃匹配不到MAC地址的报文 11](#_Toc80554975)

[10.14 配置MAC地址变化及时上报告警功能 11](#_Toc80554976)

[11 VLAN配置 12](#_Toc80554977)

[11.1 链接类型 12](#_Toc80554978)

[11.1.1 接入链路（Access Link） 12](#_Toc80554979)

[11.1.2 干道链路（Trunk Link） 12](#_Toc80554980)

[11.2 接口类型 12](#_Toc80554981)

[11.2.1 Access接口 12](#_Toc80554982)

[11.2.2 Trunk接口 13](#_Toc80554983)

[11.2.3 Hybrid接口 13](#_Toc80554984)

[11.2.4 QinQ接口 13](#_Toc80554985)

[12 VXLAN配置 13](#_Toc80554986)

[13 以太网技术 14](#_Toc80554987)

[13.1 物理层 14](#_Toc80554988)

[13.1.1 CSMA/CD（Carrier Sense Multiple Access/Collision Detection，载波侦听多路访问及冲突检测技术）。 14](#_Toc80554989)

[13.1.2 双工模式 14](#_Toc80554990)

[13.2 数据链路层 14](#_Toc80554991)

[13.2.1 媒体接入控制子层（MAC） 14](#_Toc80554992)

[13.2.2 逻辑链路控制子层（LLC）。 14](#_Toc80554993)

[13.3 以太网帧结构 15](#_Toc80554994)

[13.3.1 Ethernet II帧 15](#_Toc80554995)

[13.3.2 IEEE802.3的帧结构 15](#_Toc80554996)

[13.3.3 基于802.1Q的VLAN帧格式 15](#_Toc80554997)

[14 路由技术 16](#_Toc80554998)

[14.1 直连路由：通过链路层协议发现的路由。 16](#_Toc80554999)

[14.2 静态路由：通过网络管理员手动配置的路由。 16](#_Toc80555000)

[14.3 动态路由：通过动态路由协议发现的路由。 16](#_Toc80555001)

[14.3.1 内部网关协议（Interior Gateway Protocol，IGP） 17](#_Toc80555002)

[14.3.2 外部网关协议（Exterior Gateway Protocol，EGP） 17](#_Toc80555003)

[14.3.3 距离矢量协议（Distance-Vector Protocol）： 18](#_Toc80555004)

[14.3.4 链路状态协议（Link-State Protocol）： 18](#_Toc80555005)

[15 NETCONF 18](#_Toc80555006)

[15.1 Sync能力 19](#_Toc80555007)

[15.1.1 全量同步 19](#_Toc80555008)

[15.1.2 增量同步 20](#_Toc80555009)

[16 YANG模型 20](#_Toc80555010)

[16.1 YANG模型文件组成 20](#_Toc80555011)

[16.1.1 模块定义 20](#_Toc80555012)

[16.1.2 模块的名字空间 20](#_Toc80555013)

[16.1.3 模块的版本信息 21](#_Toc80555014)

[16.1.4 模型的描述和介绍信息 21](#_Toc80555015)

[16.2 操作定义 21](#_Toc80555016)

[16.2.1 •使用RPC定义操作 21](#_Toc80555017)

[16.2.2 使用Action定义操作 21](#_Toc80555018)

[16.2.3 通知机制（Notification） 22](#_Toc80555019)

[16.3 YANG模型节点类型 22](#_Toc80555020)

[16.3.1 Leaf（叶）节点 22](#_Toc80555021)

[16.3.2 Leaf-list（叶列表）节点 22](#_Toc80555022)

[16.3.3 Container（容器）节点 23](#_Toc80555023)

[16.3.4 List（列表）节点 23](#_Toc80555024)

[16.3.5 Grouping可重用节点组 23](#_Toc80555025)

[16.3.6 Choice（选择）节点 24](#_Toc80555026)

[16.4 YANG模型数据类型 24](#_Toc80555027)

[16.4.1 内置类型 24](#_Toc80555028)

[16.4.2 派生数据类型 24](#_Toc80555029)

[16.4.3 扩展数据类型 24](#_Toc80555030)

[16.5 NETCONF基本操作（YANG） 25](#_Toc80555031)

[16.5.1 <get>查询数据 25](#_Toc80555032)

[16.5.2 <get-config>获取配置数据 25](#_Toc80555033)

[16.5.3 <edit-config>编辑配置数据 25](#_Toc80555034)

[16.5.4 <copy-config>拷贝操作 26](#_Toc80555035)

[16.5.5 <delete-config>删除配置数据 26](#_Toc80555036)

[16.5.6 <lock>锁定配置数据库 27](#_Toc80555037)

[16.5.7 <unlock>取消锁定配置数据库 27](#_Toc80555038)

[16.5.8 <close-session>关闭NETCONF会话 27](#_Toc80555039)

[16.5.9 <kill-session>强制关闭NETCONF会话 27](#_Toc80555040)

[17 跨设备链路聚合组（Multichassis Link Aggregation Group，M-LAG） 27](#_Toc80555041)

[18 数据中心桥（Data Center Bridging，DCB） 28](#_Toc80555042)

[18.1 PFC（Priority-based Flow Control，基于优先级流量控制） 28](#_Toc80555043)

[18.2 ETS（Enhanced Transmission Selection，增强传输选择） 28](#_Toc80555044)

[18.2.1 基于优先级组的流量控制 29](#_Toc80555045)

[18.2.2 基于优先级队列的流量控制 29](#_Toc80555046)

[18.3 DCBX（Data Center Bridging Exchange Protocol，数据中心桥接交换协议） 29](#_Toc80555047)

[18.4 DCB查询命令 30](#_Toc80555048)

[19 智能无损网络（Artificial Intelligence Lossless Fabric，AI-Fabric） 30](#_Toc80555049)

[19.1 流量控制技术 30](#_Toc80555050)

[19.1.1 以太Pause机制 31](#_Toc80555051)

[19.1.2 PFC（Priority-based Flow Control，基于优先级的流量控制） 31](#_Toc80555052)

[19.2 拥塞控制技术 31](#_Toc80555053)

[19.2.1 DCQCN（Data Center Quantized Congestion Notification） 31](#_Toc80555054)

[19.2.2 AI ECN（Artificial Intelligence Explicit Congestion Notification） 31](#_Toc80555055)

[19.2.3 iQCN（intelligent Quantized Congestion Notification） 32](#_Toc80555056)

[19.2.4 ECN Overlay 33](#_Toc80555057)

[19.3 智能无损存储网络技术 33](#_Toc80555058)

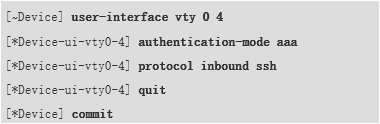
[19.3.1 iNOF（Intelligent Lossless NVMe Over Fabric，智能无损存储网络） 33](#_Toc80555059)

# 命令行格式约定

| **格式** | **意义** |
| --- | --- |
| **粗体** | 命令行关键字（命令中保持不变、必须照输的部分）采用**加粗**字体表示。 |
| *斜体* | 命令行参数（命令中必须由实际值进行替代的部分）采用*斜体*表示。 |
| [ ] | 表示用“[ ]”括起来的部分在命令配置时是可选的。 |
| { x | y | ... } | 表示从两个或多个选项中选取一个。 |
| [ x | y | ... ] | 表示从两个或多个选项中选取一个或者不选。 |
| { x | y | ... } \* | 表示从两个或多个选项中选取多个，最少选取一个，最多选取所有选项。 |
| [ x | y | ... ] \* | 表示从两个或多个选项中选取多个或者不选。 |
| &<1-n> | 表示符号&的参数可以重复1～n次。 |
| # | 由“#”开始的行表示为注释行。 |

# 配置设备的管理地址和路由

配置通过STelnet远程登录的0～4号用户的级别为3级，认证方式为AAA认证



配置接口参数的视图称为接口视图。在该视图下可以配置接口相关的物理属性、链路层特性及IP地址等重要参数。

# 路由协议视图

路由协议的大部分参数是在相应的路由协议视图下进行配置的。例如IS-IS协议视图、OSPF协议视图、RIP协议视图。

进入视图：在系统视图下，使用路由协议进程运行命令可以进入到相应的路由协议视图。

# 配置生效模式

系统支持两种配置生效模式，即立即生效模式和两阶段生效模式

## 立即生效模式

<HUAWEI> **system-view immediately**

## 两阶段生效模式

查看用户未提交的配置。display configuration candidate [ merge ]

指定merge参数，可以查看用户未提交的配置和系统已有的配置。不指定merge参数仅查看当前用户未提交的配置。

清除未提交的所有配置。clear configuration candidate

在提交配置之前，用户可以继续对未提交的配置进行编辑。

用户锁定配置的命令是**configuration exclusive**，如果配置被其他用户锁定，则首先需要联系该用户解除配置锁定。

# 命令行智能回退功能

如果命令行在当前视图下无法匹配成功，**系统会自动回溯到系统视图下**，如果可以匹配该命令行，则可以直接下发配置，从而减少了命令行操作步骤。

如果用户想关闭命令行智能回退功能，可以在用户视图下执行命令undo terminal command forward matched upper-view，关闭命令行智能回退功能。

智能回退需要输入完整命令行，不支持联想。

# 用户界面

系统支持的用户界面（User-interface）有**Console用户界面**和**VTY用户界面**。

每个用户界面有对应的**用户界面视图**。用户界面视图是系统提供的一种命令行视图，用来配置和管理所有工作在**异步交互方式下**的物理接口和逻辑接口，从而达到统一管理各种用户界面的目的。

## Console用户界面

控制口（Console Port）是一种通信串行口，由设备的主控板提供。

一块主控板提供一个Console口，接口类型为EIA/TIA-232 DCE。用户终端的串行口可以与设备Console口直接连接，实现对设备的本地访问。

## VTY用户界面

虚拟类型终端（Virtual Type Terminal）是一种虚拟线路端口。

用户通过终端与设备建立Telnet或安全外壳SSH（Secure Shell）连接后，即建立了一条VTY，即用户可以通过VTY方式登录设备。**最多支持21个用户同时通过VTY方式访问设备。**

用户登录时，系统会根据用户的登录方式，自动给用户分配一个当前空闲的、编号最小的某类型的用户界面，整个登录过程将受该用户界面视图下配置的约束。使用**display user-interface（不带参数）可查看到设备当前支持的用户界面以及它们的绝对编号。**每个主控板上Console口只有一个，但VTY类型的用户界面有21个，可以在系统视图下使用**user-interface maximum-vty命令设置最大用户界面个数，其缺省值为5**。VTY 16～VTY 20一直存在于系统中，不受user-interface maximum-vty命令的控制。

# 用户界面的用户验证

对用户的验证有如下几种方式：

•Password验证：只需要口令，不需要用户名。

•AAA验证：需要用户提供用户名和口令，对Telnet用户一般采用AAA验证。

•None验证：用户无需通过验证即可通过用户界面登录设备。

# 接口分类

接口是设备与网络中的其它设备交换数据并相互作用的部件，分为物理接口和逻辑接口两类。

## 物理接口

### 管理接口

| **接口名称** | **接口描述** | **接口用途** |
| --- | --- | --- |
| Console接口 | 遵循EIA/TIA-232标准，接口类型是DCE。 | 该接口和配置终端的COM串口连接，用于搭建现场配置环境。 |
| MEth接口 | 遵循10/100/1000BASE-TX标准。 | 该接口和配置终端或网管站的网口连接，用于搭建现场或远程配置环境。 |

### 业务接口

|  |  |
| --- | --- |
| 三层以太网接口 | 接口工作在网络层，可以配置IPv4/IPv6地址，处理三层协议，提供路由功能。 |
| 二层以太网接口 | 接口工作在数据链路层，处理二层协议，实现二层快速转发。 |

## 逻辑接口

逻辑接口是指能够实现数据交换功能但物理上不存在、需要通过配置建立的接口。逻辑接口需要承担业务传输。

| **接口类型** | **接口描述** |
| --- | --- |
| 以太网二层子接口 | 以太网二层子接口是在物理接口上配置出来的具有二层特性的逻辑接口，可以在**一个物理接口上配置多个子接口**。 |
| 以太网三层子接口 | 以太网三层子接口是在物理接口上配置出来的具有三层特性的逻辑接口，可以在**一个物理接口上配置多个子接口**。 |
| VLANIF接口 | VLANIF接口是基于VLAN的具有三层特性的逻辑接口，**每个VLAN对应一个VLANIF接口。** |
| Eth-Trunk接口 | Eth-Trunk接口是将多个以太网接口捆绑成的一个具有二层特性或三层特性的逻辑接口。捆绑在一起的每个以太网接口称为成员接口。 |
| Eth-Trunk二层子接口 | Eth-Trunk二层子接口是在Eth-Trunk接口上配置出来的具有二层特性的逻辑接口，**可以在一个Eth-Trunk接口上配置多个Eth-Trunk子接口**。 |
| Eth-Trunk三层子接口 | Eth-Trunk三层子接口是在Eth-Trunk接口上配置出来的具有三层特性的逻辑接口，**可以在一个Eth-Trunk接口上配置多个Eth-Trunk子接口**。 |
| NVE接口 | NVE接口是实现网络虚拟化功能的逻辑接口。 |
| VBDIF接口 | VBDIF接口是基于BD创建的三层逻辑接口。 |
| Loopback接口 | Loopback接口是一种逻辑接口，任何送到该接口的数据报文都会被认为是送往设备自身的。 |
| NULL0接口 | NULL0接口是系统自动创建的一个接口。NULL0接口一直处于UP状态，但是不能转发数据报文，任何发送到该接口的网络数据报文都会被丢弃。不能在NULL0接口上配置IP地址，也不能在NULL0接口上封装任何链路层协议。 |
| Tunnel接口 | Tunnel接口是具有三层特性的逻辑接口，隧道两端的设备利用Tunnel接口发送报文、识别并处理来自隧道的报文。 |

# 接口配置

## 基础配置

### 配置接口的描述信息

**description** *description*

### 配置接口的MTU值

**mtu** *mtu*

重启接口使配置生效。**restart**

执行命令**display interface** { *interface-name* | *interface-type* *interface-number* }，查看接口当前运行状态信息。

### 配置流量统计时间间隔

通过配置接口的流量统计时间间隔功能，用户可以对感兴趣的报文进行统计与分析。同时，通过预先查看接口的流量统计，及时采取流量控制的措施，可以避免网络拥塞和业务中断。

**set flow-stat interval** *interval*

### 配置开启或关闭接口

当修改了接口的工作参数配置，且新的配置未能立即生效时，可以依次执行**shutdown和undo shutdown命令或restart命令关闭和重启接口，使新的配置生效。**

当接口闲置（即没有连接电缆或光纤）时，请使用shutdown命令关闭该接口，以防止由于干扰导致接口异常。

依次执行shutdown和undo shutdown相当于执行restart命令，不会修改或删除接口的配置信息。

## 以太网接口配置

### 配置以太网接口的自协商功能

**undo negotiation disable**

### 配置以太网接口的二三层模式切换

配置接口切换到二层模式。

**portswitch**

配置接口切换到三层模式。

undo portswitch

配置接口批量切换到二层模式。

portswitch batch interface-type { interface-number1 [ to interface-number2 ] } &<1-10>

配置接口批量切换到三层模式。

undo portswitch batch interface-type { interface-number1 [ to interface-number2 ] } &<1-10>

任意视图下执行命令**display interface** [ *interface-type* [ *interface-number* ] ]，或接口视图下执行命令**display this interface**，查看接口当前运行状态信息。回显信息中出现**Switch Port**字段表示接口是二层接口；出现**Route Port**字段表示接口是三层接口。

### 配置以太网光/电接口的接口速率

1. 接口速率

系统视图下配置以太网接口的接口速率

| **操作** | **命令** | **说明** |
| --- | --- | --- |
| 配置接口速率：25GE接口工作在50Gbit/s速率模式。 | **port mode 50ge interface** { *interface-type1* *interface-number1* [ **to** *interface-type2* *interface-number2* ] } &<1-18> | 缺省情况下，25GE接口插入25GE光模块工作在25Gbit/s速率模式，插入50GE光模块接口不up。 |
| 配置接口速率：100GE接口工作在200Gbit/s速率模式。 | **port mode 200ge interface** { *interface-type1* *interface-number1* [ **to** *interface-type2* *interface-number2* ] } &<1-18> | 缺省情况下，100GE接口插入100GE光模块工作在100Gbit/s速率模式，插入200GE光模块接口不up。 |

接口视图下配置以太网接口的接口速率

| **操作** | **命令** | **说明** |
| --- | --- | --- |
| 配置接口速率：25GE接口工作在10Gbit/s速率模式。 | **port mode 10g** | 缺省情况下，25GE接口和100GE接口拆分出来的25GE接口，工作在25Gbit/s速率模式。 |
| 配置接口速率：50GE接口工作在25Gbit/s速率模式。 | - | 50GE接口插入25GE光模块或者高速线缆，接口速率自适应为25Gbit/s。 |
| 配置接口速率：100GE接口工作在40Gbit/s速率模式。 | **speed 40000** | 缺省情况下，接口的速率为接口支持的最大速率，即100GE接口工作在100Gbit/s速率模式。  此命令只在接口下插入100GE高速线缆时生效。 |
| 配置接口速率：200GE接口工作在40Gbit/s速率模式。 | **speed 40000** | 缺省情况下，接口的速率为接口支持的最大速率，即200GE接口工作在200Gbit/s速率模式。  此命令只在接口下插入100GE高速线缆时生效。 |
| 配置接口速率：400GE接口工作在40Gbit/s速率模式。 | **speed 40000** | 缺省情况下，接口的速率为接口支持的最大速率，即400GE接口工作在400Gbit/s速率模式。  此命令只在接口下插入100GE高速线缆时生效。 |
| 配置接口速率：400GE接口工作在200Gbit/s速率模式。 | - | 400GE接口插入200GE光模块，接口速率自适应为200Gbit/s。 |

### 配置以太网接口拆分

接口拆分是指将设备上一个高带宽的物理接口，拆分为多个相互独立的低带宽接口。用户可以根据对接设备提供的接口类型，自由选择使用原有的高带宽接口，还是将其拆分为多个低带宽的接口使用，接口拆分功能可以增加组网灵活性，减少用户购置成本。

设备上支持拆分的接口

#### 100GE接口支持拆分为4个25GE接口。

#### 100GE接口支持拆分为2个50GE接口。

#### 400GE接口支持拆分为4个100GE接口。

**port split** **dimension** **interface** { *interface-type* *interface-number1* [ **to** *interface-type* *interface-number2* ] } &<1-18> [ **split-type** *split-type* ]

执行命令**display port split** [ **slot** *slot-id* ]，查看接口拆分信息。

## 逻辑接口配置

## 端口隔离配置

端口隔离是将需要进行隔离的端口加入到一个**隔离组**中，实现隔离组中的端口之间**二层、三层数据的隔离**。配置隔离组后，**只有隔离组内各个端口之间**的报文不能互通，隔离组内端口与隔离组外端口的通信不会受到影响。

使能端口二层隔离功能。

port-isolate enable group group-id

使能端口三层隔离功能。

port-isolate l3 enable

# MAC配置

## MAC简介

**MAC（Media Access Control）**地址，也称为物理地址、硬件地址或链路地址，由网络设备制造商生产时写在网卡内部，用来定义网络设备的位置。MAC地址由**48比特长、12位的16进制数字组成**，其中从左到右开始，0到23bit是厂商向IETF等机构申请用来标识厂商的代码，24到47bit由厂商自行分派，是各个厂商制造的所有网卡的一个唯一编号。

MAC地址可以分为3种类型：

•**物理MAC地址**：这种类型的MAC地址唯一的标识了以太网上的一个终端，该地址为全球唯一的硬件地址。

•**广播MAC地址**：全1的MAC地址为广播地址（FF-FF-FF-FF-FF-FF），用来表示LAN上的所有终端设备。

•**组播MAC地址**：除广播地址外，第8bit为1的MAC地址为组播MAC地址（例如01-00-00-00-00-00），用来代表LAN上的一组终端。其中以01-80-c2开头的组播MAC地址叫BPDU MAC，一般作为协议报文的目的MAC地址标示某种协议报文。

## MAC地址表

MAC地址表记录了设备学习到的其他设备的**MAC地址与接口**的对应关系，以及接口**所属VLAN等信息**。MAC地址表是以**MAC地址和VLAN ID为索引**来唯一标识。当一台目的主机属于多个VLAN时，在MAC地址表中就会存在相同MAC地址拥有多个不同VLAN ID的情况。

MAC地址表用于指导报文进行单播转发。如果MAC地址表中包含与报文目的MAC地址对应的表项，则直接通过该表项中的出接口转发该报文；如果MAC地址表中没有包含报文目的MAC地址对应的表项时，设备将采取**广播方式在所属VLAN内除接收接口外的所有接口转发该报文。**

MAC地址表中的表项分为：

1. 动态MAC表项
2. 静态MAC表项
3. 黑洞MAC表项

## 配置静态MAC表项

**mac-address static** *mac-address* *interface-type interface-number* **vlan** *vlan-id*

## 配置黑洞MAC表项

mac-address blackhole mac-address vlan vlan-id

## 配置动态MAC表项的老化时间

**mac-address aging-time** *timeValue*

执行命令**display mac-address aging-time**，查看动态MAC表项的老化时间。

## 配置关闭MAC地址学习功能

## 配置MAC地址限制

## 配置MAC地址防漂移和漂移检测

## 配置三层逻辑接口的MAC地址

缺省情况下，三层接口的接口MAC是从系统MAC范围段中动态分配的。对于三层物理接口，接口MAC不可以重新指定；但对于**三层逻辑接口**，比如VLANIF接口、Eth-Trunk接口、VBDIF接口，支持重新配置接口MAC。

## 配置MAC刷新ARP功能

## 配置端口桥功能

## 配置丢弃全零MAC地址报文

## 配置丢弃匹配不到MAC地址的报文

## 配置MAC地址变化及时上报告警功能

# VLAN配置

**VLAN（Virtual Local Area Network）即虚拟局域网**，是将一个物理的LAN在逻辑上划分成多个广播域的通信技术。每个VLAN是一个广播域，使得VLAN内的主机可以互通，而VLAN间不能直接互通

## 链接类型

### 接入链路（Access Link）

连接不能或不需要识别VLAN Tag的终端设备。用户主机和设备之间的链路都是接入链路，接入链路上通过的帧为Untagged帧。

### 干道链路（Trunk Link）

连接设备和设备的链路。干道链路上通过的帧为Tagged帧，所有数据帧必须都打上VLAN Tag。

## 接口类型

在802.1Q协议中定义VLAN帧后，设备的有些接口可以识别VLAN帧，有些接口则不能识别VLAN帧。根据对VLAN帧的识别情况，将接口分以下几类：

### Access接口

是设备上用来连接终端设备的接口，这些设备不能或不需要识别VLAN Tag。该接口只能连接接入链路，有如下特点：

◾由于设备内部只处理Tagged帧，如果该接口收到Untagged帧，设备将强制加上该接口的PVID（缺省VLAN）。

◾如果该接口收到Tagged帧，只允许VLAN ID与接口的PVID（缺省VLAN）相同的VLAN通过该接口。

◾Access接口发送的数据帧永远是Untagged帧。

### Trunk接口

是设备上用来和其他设备连接的接口，这些设备能识别VLAN Tag。该接口只能连接干道链路，有如下特点：

◾允许多个VLAN的帧（Tagged帧）通过。

◾该接口发送的帧，接口缺省VLAN内的帧是Untagged帧（VLAN ID=PVID），其他VLAN内的帧都必须是Tagged帧。

### Hybrid接口

既可以连接Access接口能连接的设备，也可以连接Trunk接口能连接的设备。该接口既可以连接接入链路又可以连接干道链路，有如下特点：

◾不确定是否能识别VLAN Tag的情况下，建议使用该接口。

◾允许多个VLAN的帧（Tagged帧）通过，并且在出接口方向可根据需要设置某些VLAN内的帧带Tag，某些VLAN内的帧不带Tag。

### QinQ接口

是指使用QinQ协议的接口。QinQ接口可以给数据帧加上双重VLAN Tag，即在原来VLAN Tag的基础上，给数据帧加上一个新的VLAN Tag，从而满足网络对VLAN数量的需求。因此，也称为802.1Q-in-802.1Q接口。

# VXLAN配置

**VLAN扩展方案VXLAN（Virtual eXtensible Local Area Network）**。VXLAN采用MAC in UDP（User Datagram Protocol）封装方式，是NVO3（Network Virtualization over Layer 3）中的一种**网络虚拟化技术**。

# 以太网技术

## 物理层

### CSMA/CD（Carrier Sense Multiple Access/Collision Detection，载波侦听多路访问及冲突检测技术）。

•CS：载波侦听，在发送数据之前进行侦听，以确保线路空闲，减少冲突的机会。

•MA：多路访问，每个站点发送的数据，可以同时被多个站点接收。

•CD：冲突检测，由于两个站点同时发送信号，信号叠加后，会使线路上电压的摆动值超过正常值一倍。据此可判断冲突的产生。边发送边检测，发现冲突就停止发送，然后延迟一个随机时间之后继续发送。

### 双工模式

半双工模式

全双工模式

## 数据链路层

数据链路层分为**媒体接入控制子层（MAC）**和**逻辑链路控制子层（LLC**）。

### 媒体接入控制子层（MAC）

在以太网中，针对**物理层不同的双工模式**，提供不同的介质访问方法：

•在半双工模式下采用的是CSMA/CD的访问方式。

•在全双工模式下则可以直接进行收发，不用预先判断链路的忙闲状态。

不同的物理层有不同的MAC子层来进行访问。在以太网中，主要存在两种MAC子层：

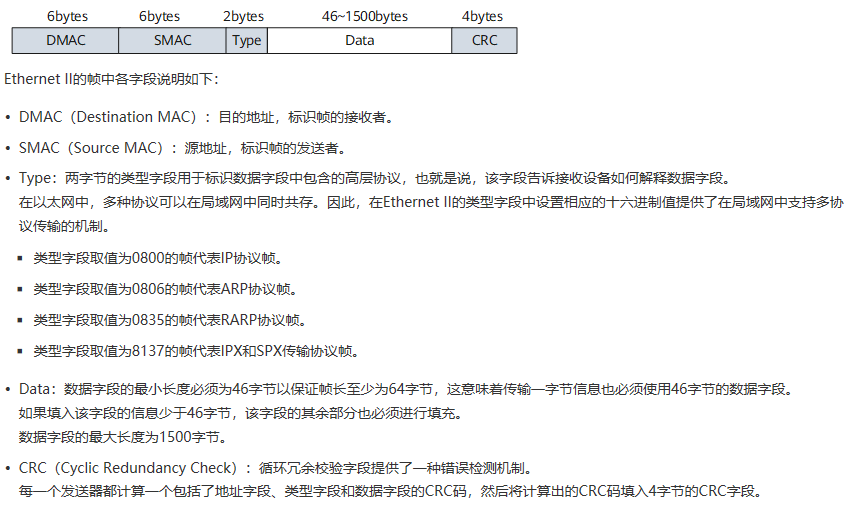
◾半双工MAC：物理层运行模式是半双工时提供访问。

◾全双工MAC：物理层运行模式是全双工时提供访问。

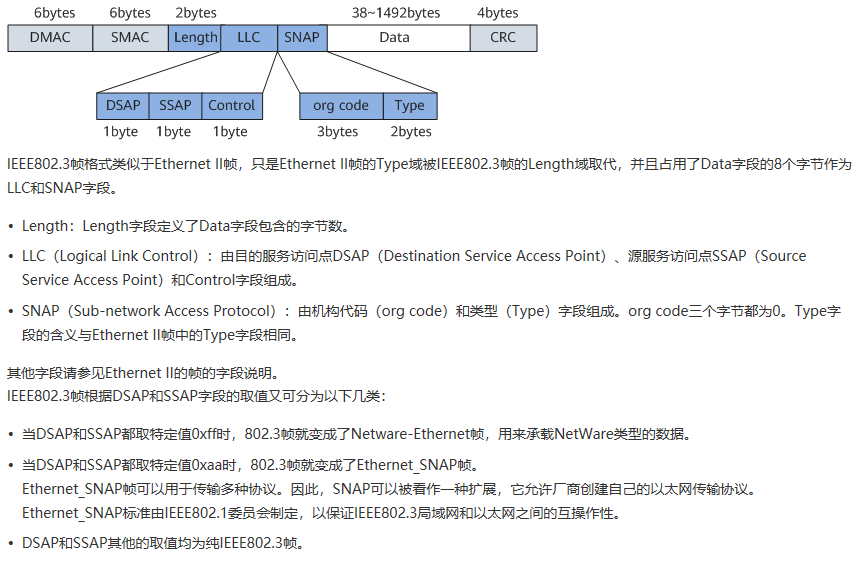
### 逻辑链路控制子层（LLC）。

## 以太网帧结构

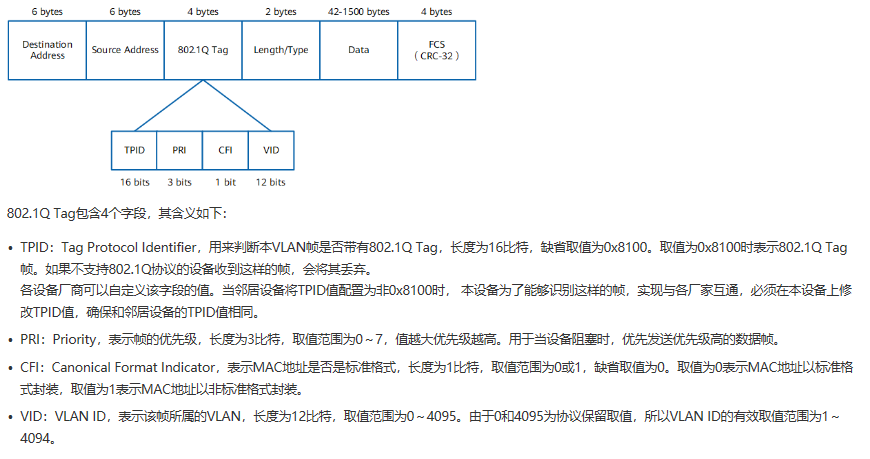
### Ethernet II帧



### IEEE802.3的帧结构



### 基于802.1Q的VLAN帧格式



报文的五元组信息指报文的源IP地址、目的IP地址、源端口号、目的端口号、协议类型，

# 路由技术

## 直连路由：通过链路层协议发现的路由。

## 静态路由：通过网络管理员手动配置的路由。

## 动态路由：通过动态路由协议发现的路由。

根据作用范围不同，路由协议可分为：

### 内部网关协议（Interior Gateway Protocol，IGP）

#### RIP (Routing Information Protocol，路由信息协议)

路由信息协议 ( Routing Information Protocol，RIP) 是一种较为简单的内部网关协议IGP（Interior Gateway Protocol）。

#### OSPF（Open Shortest Path First，开放式最短路径优先）

OSPF（Open Shortest Path First）是IETF组织开发的一个基于链路状态的内部网关协议IGP（Interior Gateway Protocol）。

#### IS-IS（Intermediate System to Intermediate System）

中间系统到中间系统（Intermediate System to Intermediate System，IS-IS）一个基于链路状态的内部网关协议IGP（Interior Gateway Protocol），使用最短路径优先SPF（Shortest Path First）算法进行路由计算。

区别

| **对比项** | **RIP** | **OSPF** | **IS-IS** |
| --- | --- | --- | --- |
| 协议类型 | IP层协议。 | IP层协议。 | 链路层协议。 |
| 适用范围 | 应用于规模较小的网络中。例如，校园网等结构简单的地区性网络。 | 应用于规模适中的网络，最多可支持几百台设备。例如中小型企业网络。 | 应用于规模较大的网络中。例如，大型ISP中 |
| 路由算法 | 采用距离矢量（Distance-Vector）算法计算路由。 | 采用最短路径SPF（Shortest Path First）算法。通过链路状态通告LSA（Link State Advertisement）描述网络拓扑，依据网络拓扑生成一棵最短路径树SPT（Shortest Path Tree），计算出到网络中所有目的地的最短路径。 | 采用最短路径SPF算法。依据网络拓扑生成一棵最短路径树SPT，计算出到网络中所有目的地的最短路径。  在IS-IS中，SPF算法分别独立的在Level-1和Level-2数据库中运行。 |
| 收敛速度 | 收敛速度慢。 | 收敛速度快，小于1s。 | 收敛速度快，小于1s。 |
| 扩展性 | 不能扩展。 | 通过划分区域扩展网路支撑能力。 | 通过Level路由器扩展网路支撑能力。 |

### 外部网关协议（Exterior Gateway Protocol，EGP）

#### BGP（Border Gateway Protocol）

BGP（Border Gateway Protocol）是一种用于自治系统AS（Autonomous System）之间的动态路由协议。

根据使用算法不同，路由协议可分为：

### 距离矢量协议（Distance-Vector Protocol）：

#### RIP

#### BGP

### 链路状态协议（Link-State Protocol）：

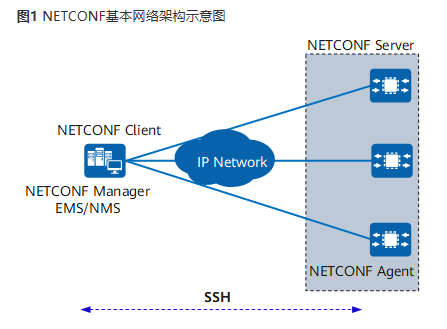
#### OSPF

#### IS-IS

# NETCONF

**NETCONF（Network Configuration Protocol）**是基于可扩展标记语言XML（Extensible Markup Language）的网络配置和管理协议，并使用简单的基于RPC（Remote Procedure Call）机制实现客户端和服务器之间通信。NETCONF提供了一种通过运行网络管理软件的中心计算机（即网络管理工作站，客户端）来远程管理和监控设备（服务器端）的方法。

简单网络管理协议SNMP（Simple Network Management Protocol）



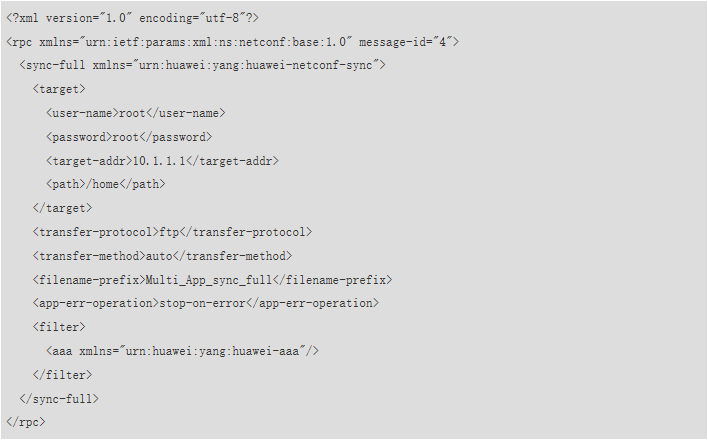
整套系统必须包含至少一个**NMS（Network Management System）**作为整个网络的网管中心，NMS运行在NMS服务器上，对设备进行管理。

## Sync能力

指明设备具备数据同步能力，即**全量同步和增量同步**。通过全量同步或增量同步，使得**管理网络设备的网管或控制器**中配置数据实时保持与网元配置数据一致。

### 全量同步

服务器端将获取到的全量同步数据中的AAA模块配置，通过FTP文件传输协议自动传输到用户名为root、密码为root、IP地址为10.1.1.1的服务器的home路径下，存储的文件名为Multi\_App\_sync\_full.zip。



### 增量同步

<sync-increment>用来请求设备执行增量配置数据同步。当配置发生变更时，客户端通过每次的配置变更标识符flow-id来感知变化，每变更一次，flow-id的值加1。当客户端只想获取两个配置变更点之间的配置时，就使用增量同步。

<sync-increment>操作成功，Server回复的<rpc-reply>元素中含有<data>元素，此<data>元素中包括两次提交之间产生的数据变化。否则在<rpc-reply>元素中含有<rpc-error>元素。

<sync-increment>通过difference属性标识配置数据实例的变更操作，YANG模型定义在huawei-netconf-metadata.yang文件中。

# YANG模型

## YANG模型文件组成

### 模块定义

### 模块的名字空间

### 模块的版本信息

### 模型的描述和介绍信息

## 操作定义

YANG支持定义操作，可以使用**RPC或者Action**对YANG模型中的操作进行定义，包含操作名称、输入参数和输出参数。

### •使用RPC定义操作

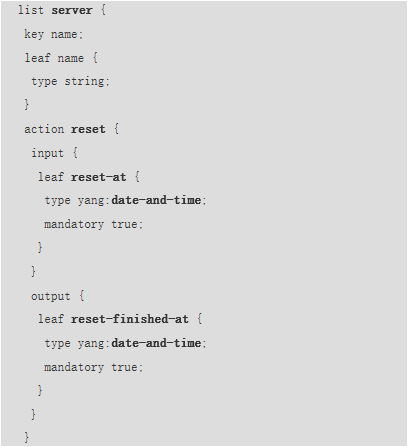
YANG提供RPC关键字，可以对**其模型顶层的操作**进行定义。

使用RPC关键字建模示例，其中，操作名称为activate-software-image；输入参数为image-name，并指定该参数的类型为字符串形式；输出参数为status，并指定该参数的类型为字符串形式。



### 使用Action定义操作

Action是YANG的一种语法，可以和container节点、list节点相关联，用来给其中的叶子节点定义操作，比如：reset、reboot，copy等。Action包含input和output两个声明。一个叶子节点可以定义多个action操作，但是每次只能下发一个，可以分多次下发。



### 通知机制（Notification）

NETCONF Notification是一种基于NETCONF协议的告警和事件的订阅和上报机制，提供了一种基于数据模型的告警和事件异步上报服务。YANG提供了**Notification关键字来对通知机制**进行建模。

不同于RPC报文的“一发一收”机制，依赖客户端来查询数据，NETCONF Notification是当告警或者事件发生时，由服务端主动通过Notification报文上报给客户端，主要用于那些需要实时感知设备情况的场景，例如告警和事件通过NETCONF协议上报网管。

## YANG模型节点类型

### Leaf（叶）节点

一个叶节点包含简单的数据，如整形数据或字符串。该节点对每个特定类型仅有一个确定值，并且没有子节点。

### Leaf-list（叶列表）节点

叶列表节点是一系列具备特定类型的**叶子节点的集合**，每个叶节点对特定数据类型有一个确定值。

### Container（容器）节点

容器节点用来描述若干相关节点的**集合**。Container节点只有子节点，而没有值。对于子节点数目和类型不作要求，可以是**叶节点、叶列表节点、容器节点或者列表节点**。

容器分为存在容器和非存在容器。

•**存在容器**：**容器自身的存在代表某种含义**，对于配置数据来说，这些容器节点既是配置按钮，又是一种组织相关配置的方法。

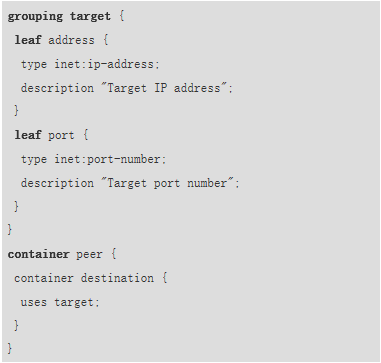
•**非存在容器**：容器自身没有任何意义，仅为了组织数据节点的层次结构，容纳子节点。是容器的默认形式。

### List（列表）节点

列表节点定义了列表条目序列，每个条目就像一个结构体或者**一个记录实例，由其关键叶节点的值（key值）唯一识别**。列表节点可定义多个关键叶节点，也可能是一系列包含任意数据类型（包括叶节点、列表节点、容器节点等）的子节点。

### Grouping可重用节点组

可以使用“grouping”声明将一组节点放到**可重用的集合**中。一个分组定义了一系列节点，这些节点可以使用**“uses”声明实例化**。



### Choice（选择）节点

YANG能够使用Choice和Case声明**不能同时出现的节点**。Choice声明包含一系列Case声明，这些Case定义了不能同时出现的模式节点集，每个Case都可能包含多个节点，但是每个节点只能在一个Case中出现。

当choice中的一个Case生效，其他Case将被隐式删除，设备会强制执行这个约束，防止出现非法的配置。

## YANG模型数据类型

### 内置类型

### 派生数据类型

YANG能够使用“typedef”声明根据基本类型定义派生类型，基本类型可以是内置类型或者派生类型。

### 扩展数据类型

YANG允许通过augment将外部的节点插入到数据模型中。这种方式比较有用，例如，供应商以可互操作的方式向标准数据模型插入供应商特定的参数。

## NETCONF基本操作（YANG）

### <get>查询数据

<get>查询的是设备**当前运行的状态数据**，即只能从<running/>配置数据库中获取数据。

<get>和<get-config>区别在于：

查询数据库的范围：

•<get>只能查询<running/>配置数据库的数据，<get-config>可以查询<running/>、<candidate/>和<startup/>中数据。

查询数据库的数据范围

•<get>可以查询配置数据和状态数据，<get-config>只能查询配置数据。

### <get-config>获取配置数据

<get-config>操作用来查询全部或部分指定**配置数据集**。

•source：正在查询的配置数据库名称，<running/>、<candidate/>或<startup/>。

•filter：此参数用来标识要查询配置数据库的范围。如果此参数不存在，则返回整个配置。

### <edit-config>编辑配置数据

#### <config>：一组由数据模型定义的层次化的配置数据。

◾merge：在数据库中修改存在或不存在的目标数据，如果目标数据不存在则创建，如果目标数据存在则修改。

◾create：当且仅当配置数据库中不存在待创建的配置数据时，才能成功添加到配置数据库。**如果配置数据存在，则会返回<rpc-error>，其中包含一个<error-tag>值**”**data-exists**”**。**

◾remove：删除配置数据库中指定的配置数据记录。如果数据存在，则删除该数据，如果数据不存在，则返回成功。

◾delete：删除配置数据库中指定的配置数据记录。如果数据存在，则删除该数据，**如果数据不存在，则返回<rpc-error>，其中包含一个<error-tag>值”data-missing”。**

◾replace：在数据库中替换存在或不存在的目标数据，如果目标数据不存在则创建，如果目标数据存在则替换。

#### <target>编辑的配置数据库。

◾ <running/>

◾ <candidate/>

#### <default-operation>：为<edit-config>操作设置默认操作。

◾merge： <config>参数中的配置数据与目标配置数据库中的**配置合并**。

◾replace：<config>参数中的配置数据，**完全替换**目标配置数据库中的配置。

◾none：除了执行operation属性指定的操作外，目标配置数据库中的配置数据不受<config>中其他配置数据影响。

#### <error-option>：用于设置一个实例配置出错后，后续实例配置的处理方式.

◾stop-on-error：出现错误后停止操作，并按照rollback-on-error进行后续处理。

◾continue-on-error：出现错误后记录错误信息并继续执行，只要发生错误，NETCONF服务器会给客户端返回一个操作失败的<rpc-reply>消息。

◾rollback-on-error：出现错误后停止操作，并将配置回退到执行<edit-config>操作之前的状态。只有设备支持rollback-on-error能力才支持此操作。

#### <test-option>

◾test-then-set

### <copy-config>拷贝操作

<copy-config>用**源配置数据库替换目标配置数据库**。如果目标配置数据库没有创建，则直接创建配置数据库，否则用源配置数据库直接覆盖目标配置数据库。

#### •支持将<candidate/>、<running/>数据库中的配置保存到url指定文件。

#### •支持将<running/>数据库中的配置拷贝至<candidate/>或<startup/>数据库。

#### •支持将url指定文件（远端文件、本地文件）中的配置数据替换<candidate/>或者<startup/>数据库中的配置。

### <delete-config>删除配置数据

用来删除一个配置数据库，但不能删除<running/>配置数据库。

<delete-config>操作成功，Server回复的<rpc-reply>元素中含有<ok>元素，否则在<rpc-reply>元素中含有<rpc-error>元素。

删除<startup/>数据库示例删除<startup/>数据库示例。

### <lock>锁定配置数据库

用来**锁定设备的配置数据库**，独占配置数据库的修改权。这种锁定允许用户在配置时，不会受到如NETCONF manager、SNMP或者CLI脚本的配置影响，防止产生冲突。

### <unlock>取消锁定配置数据库

用来取消用户自己之前执行的<lock>操作，但不能取消其他用户的<lock>操作。

### <close-session>关闭NETCONF会话

用来正常关闭当前NETCONF会话。

当NETCONF Server收到<close-session>请求时，它将关闭当前的NETCONF会话。服务器将释放与该会话相关的所有锁和资源。在收到<close-session>请求后，NETCONF Server将忽略这个会话的所有请求消息。

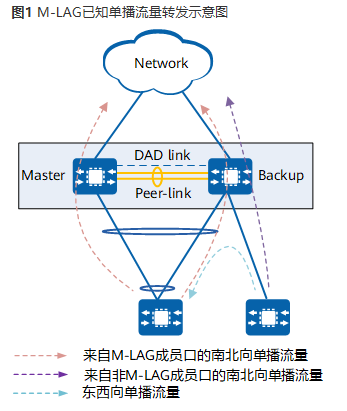
### <kill-session>强制关闭NETCONF会话

用来强制关闭另一个NETCONF会话，只有管理员用户才有权限执行<kill-session>操作。

当NETCONF Server收到<kill-session>请求时，它将中止该会话正在进行的所有操作，释放与该会话相关联的任何锁和资源，并关闭该会话。

如果NETCONF服务器在处理<commit>操作时收到<kill-session>请求，它必须将配置恢复到发出确认提交之前的状态。

# 跨设备链路聚合组（Multichassis Link Aggregation Group，M-LAG）



# 数据中心桥（Data Center Bridging，DCB）

数据中心桥接（Data Center Bridging，DCB）协议是一组由IEEE 802.1工作组定义的**以太网扩展协议**。DCB协议组主要用于构建无丢包以太网，以满足数据中心网络融合后的QoS需求。

DCB特性主要包括PFC、ETS、DCBX。

## PFC（Priority-based Flow Control，基于优先级流量控制）

## ETS（Enhanced Transmission Selection，增强传输选择）

数据中心网络融合后，LAN、SAN和IPC流量的QoS需求上存在较大差异：

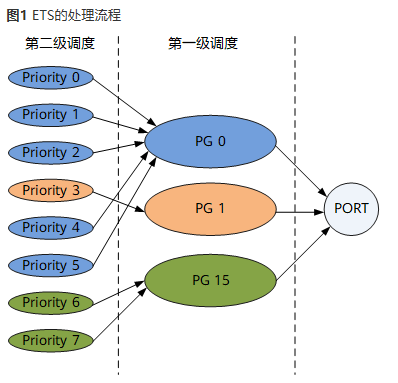
•LAN流量允许丢包，只需要设备提供尽力而为的服务BE(Best Effort)。

•SAN流量对丢包很敏感且要求报文在传输过程中是保序的。

•IPC用于服务器之间的通信，流量要求低时延。

传统的QoS已经无法满足融合网络的需求，而增强传输选择ETS（Enhanced Transmission Selection）通过**灵活的层次化的调度**实现网络融合后的QoS。

ETS提供两级调度，分别基于优先级组PG（Priority Group）和优先级队列PQ（Priority Queue）。接口首先对优先级组进行第一级调度，然后对优先级组的优先级队列进行第二级调度。相比普通QoS，ETS的优势在于提供了**基于优先级组的调度**，将同一类型的流量归入同一优先级组，使得同一类流量能够获得相同的服务等级。



### 基于优先级组的流量控制

优先级组即一组拥有**相同调度方式**的优先级队列，用户可通过设置将不同的优先级队列加入到优先级组中。基于优先级组的调度被称为第一级调度。

缺省情况下，在ETS中定义了3个优先级组**PG0、PG1和PG15**，分别代表是LAN流量、SAN流量和IPC流量。协议规定，PG0、PG1的调度方式是赤字轮循队列调度DRR（Deficit Round Robin）。由于PG15承载IPC流量，对延时要求很高，因此调度方式为是PQ（Priority Queue）。另外，用户也可根据实际情况对优先级组划分带宽。

### 基于优先级队列的流量控制

队列拥塞管理、队列整形、队列拥塞避免

## DCBX（Data Center Bridging Exchange Protocol，数据中心桥接交换协议）

在数据中心网络融合场景下，为实现无丢包以太网，**链路两端的PFC和ETS的参数配置需要保持一致**。如果依靠管理员手工配置，不仅工作量庞大而且容易出错。数据中心桥接交换协议DCBX（Data Center Bridging Exchange Protocol）作为一种链路发现协议，能够使链路两端的设备发现并交换DCB配置信息，大大减轻了管理员的工作量。

DCBX协议将需要交互的DCB配置信息封装入链路层发现协议LLDP（Link Layer Discovery Protocol）中的TLV中，由LLDP来进行链路两端设备的DCB配置交换。

## DCB查询命令

执行命令display dcb [ interface interface-type interface-number ]，查看DCB的配置信息和协商状态信息。

•执行命令display dcb ets-profile [ profile-name ]，查看ETS模板的配置信息。

•执行命令display dcb pfc-profile [ profile-name ]，查看PFC模板的配置信息。

•执行命令display dcb app-profile [ profile-name ]，查看APP模板的配置信息。

•执行命令display dcb pfc [ interface interface-type interface-number ]，查看PFC的反压帧计数。

•执行命令display dcb fail-record [ interface interface-type interface-number ] ，查看接口下DCB协商失败记录。

在用户视图下执行命令reset dcb pfc [ interface interface-type interface-number ]，清除接口下PFC的反压计数。

•在用户视图下执行命令reset dcb fail-record [ interface interface-type interface-number ]，清除接口下DCB协商失败记录。

# 智能无损网络（Artificial Intelligence Lossless Fabric，AI-Fabric）

## 流量控制技术

流量控制，也称为**链路级流控**，流量控制所要做的就是抑制流量**发送端**发送数据的速率，以便流量**接收端**来得及接收，防止设备端口在拥塞的情况下出现丢包。

### 以太Pause机制

### PFC（Priority-based Flow Control，基于优先级的流量控制）

## 拥塞控制技术

拥塞控制是一个全局性的过程，目的是让网络能承受现有的网络负荷，往往需要**转发设备、流量发送端、流量接收端**协同作用，并结合网络中的拥塞反馈机制来调节整网流量才能起到缓解拥塞、解除拥塞的效果。

### DCQCN（Data Center Quantized Congestion Notification）

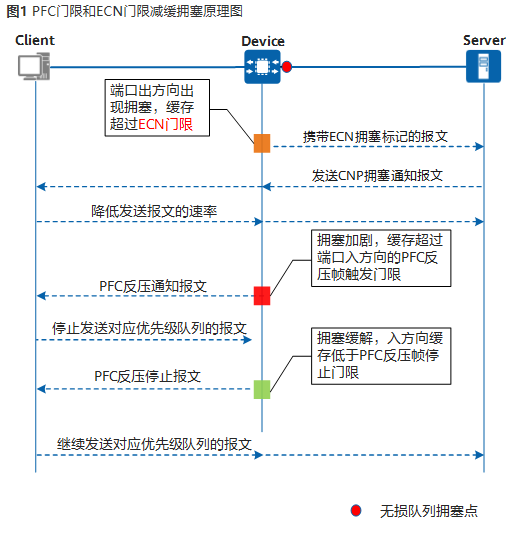
DCQCN（Data Center Quantized Congestion Notification）目前是RDMA网络应用最广泛的拥塞控制算法，DCQCN只需要可以支持ECN功能的网络设备，其他的协议功能在主机的网卡上实现。

DCQCN提供的拥塞控制机制，是在转发设备上发现队列拥塞后，由转发设备向接收端发送ECN拥塞标记报文。接收端收到ECN报文后，向发送端发送CNP拥塞通知报文，以通知发送端的网卡降低发包速率。

传统的静态ECN功能需要在转发设备上**手工配置ECN的高低门限、ECN的标记概率等参数**，对于需要无丢包传输的无损业务，无法使ECN门限适应队列中不断变化的缓存空间，在兼顾时延敏感小流和吞吐敏感大流的情况下尽量避免触发PFC流控。

### AI ECN（Artificial Intelligence Explicit Congestion Notification）

AI ECN（Artificial Intelligence Explicit Congestion Notification）可以根据**现网流量模型**进行AI训练，对网络流量变化进行预测，及时推理最优ECN门限，并且支持根据现网流量变化实时调整ECN门限，进行无损队列缓存的精确管控，保障整网的最优性能。同时，与队列调度技术配合使用时，无损队列的AI ECN功能可以实现网络中TCP流量与RoCEv2流量的混合调度，保障RoCEv2流量的无损传输的同时实现低时延和高吞吐。



1.当Device的无损队列出现拥塞，队列已使用的缓存超过**ECN门限**时，Device在转发报文中打上ECN拥塞标记（将ECN字段置为11）。

2.Server收到携带ECN拥塞标记的报文后，**向Client发送CNP拥塞通知报文**。Client收到CNP拥塞通知报文后，**降低发包速率**。

3.当Device的无损队列拥塞加剧，队列已使用的缓存超过PFC反压帧触发门限时，**Device向Client发送PFC反压通知报文**。Client收到PFC反压通知报文后，**停止发送**对应优先级队列的报文。

4.当Device的无损队列拥塞缓解，队列已使用的缓存低于PFC反压帧停止门限时，Device向Client发送PFC反压停止报文。Client收到PFC反压停止报文后，继续发送对应优先级队列的报文。

### iQCN（intelligent Quantized Congestion Notification）

### ECN Overlay

## 智能无损存储网络技术

### iNOF（Intelligent Lossless NVMe Over Fabric，智能无损存储网络）