

# 第3篇 配置局域网交换

---

第11章 以太网交换机工作原理

第12章 配置VLAN

第13章 生成树协议

第14章 交换机端口安全技术

第15章 配置链路聚合

# 第11章 以太网交换机工作原理

在局域网中，交换机是非常重要的网络设备，负责在主机之间快速转发数据帧。交换机与集线器的不同之处在于，交换机工作在数据链路层，能够根据数据帧中的 MAC 地址进行转发。本章介绍了共享式以太网和交换式以太网的区别，最后重点讲述了交换机进行 MAC 地址学习以构建 MAC 地址表的过程，对数据帧的转发原理。

## 11.1 本章目标

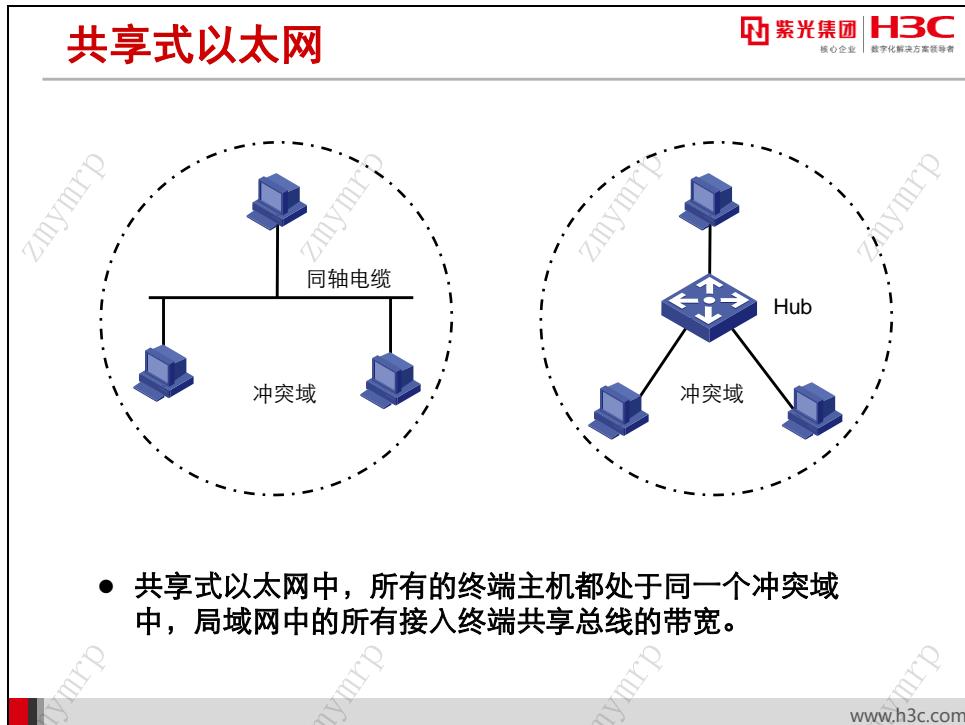
### 课程目标

● 学习完本课程，您应该能够：

- 了解共享式以太网和交换式以太网的区别
- 掌握交换机中MAC地址表的学习过程
- 掌握交换机的过滤、转发原理
- 掌握广播域的概念



## 11.2 共享式与交换式以太网

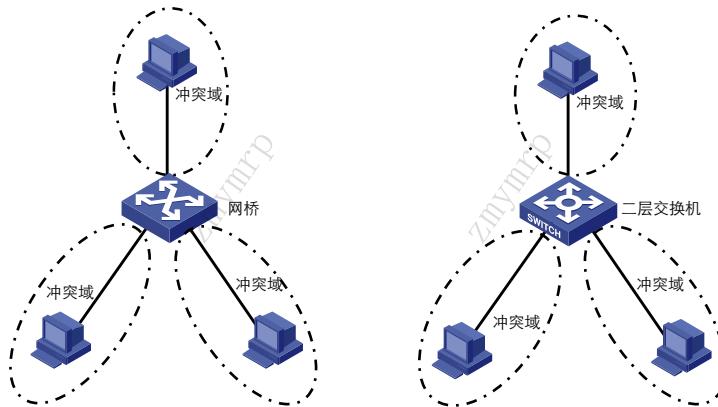


Hub 与同轴电缆都是典型的共享式以太网所使用的设备，工作在 OSI 模型的物理层。Hub 和同轴电缆所连接的设备位于一个冲突域中，域中的设备共享带宽，设备间利用 CSMA/CD 机制来检测及避免冲突。当网络中设备数量较少时，冲突较少发生，通信质量可以得到较好地保证；但是当设备数量增加到一定程度时，将导致冲突不断，网络的吞吐量受到严重影响，数据可能频繁地由于冲突而被拒绝发送。

通过 Hub 或同轴电缆接入的终端会共享总线的带宽，接入的终端数量越多，每个终端获得的网络带宽越少；并且一个终端发出的报文（无论是单播、组播、广播），其余终端都可以收到。

交换式以太网的出现有效地解决了这个问题，它大大减小了冲突域的范围。

## 交换式以太网



- 在交换式以太网中，交换机的每个端口处于独立的冲突域中，终端主机独占端口的带宽。

[www.h3c.com](http://www.h3c.com)

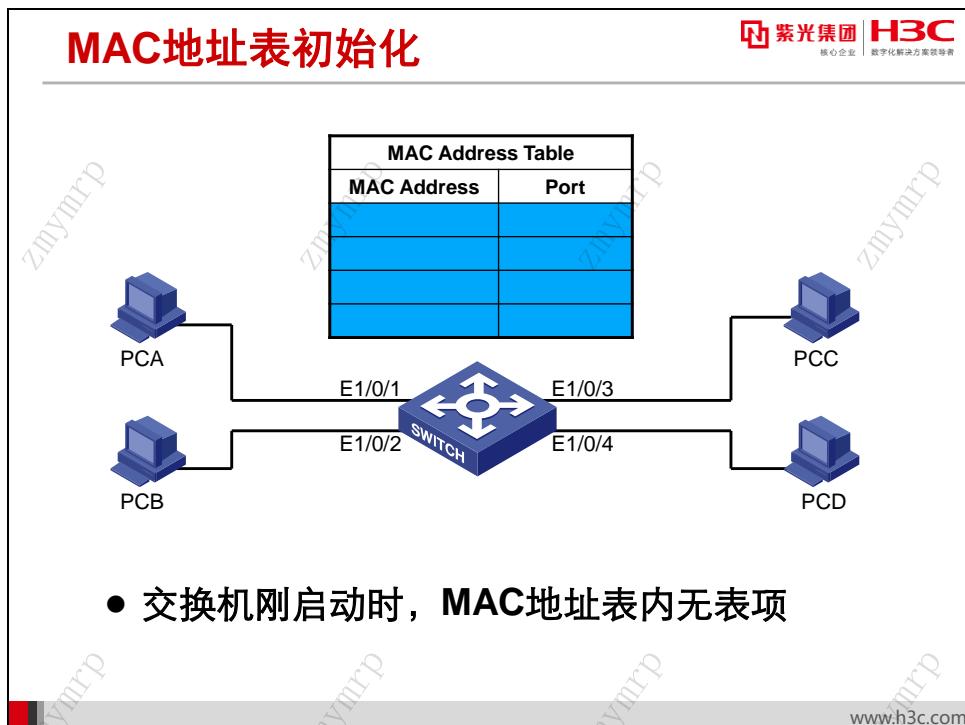
网桥（Bridge）是一种工作在数据链路层的设备，早期被用在网络中连接各个终端主机。对于终端主机来说，网桥好像是透明的，不需要由于网桥的存在而增加或改变配置，所以又称为透明网桥。网桥遵循的协议是 IEEE 802.1D，又称为透明桥接协议。

目前在交换式以太网中经常使用的网络设备是二层交换机。二层交换机和网桥的工作原理相同，都是按照 IEEE 802.1D 标准设计的局域网连接设备。他们的区别在于交换机比网桥的端口更多、转发能力更强、特性更加丰富。

二层交换机的端口在检测到网络中的比特流后，它会首先把比特流还原成数据链路层的数据帧，再对数据帧进行相应的操作。同样，二层交换机端口在发送数据时，会把数据帧转成比特流，再从端口发送出去。二层交换机也采用 CSMA/CD 机制来检测及避免冲突，但与 Hub 所不同的是，二层交换机各个端口会独立地进行冲突检测，发送和接受数据，互不干扰。所以，二层交换机中各个端口属于不同的冲突域，端口之间不会有竞争带宽的冲突发生。

由于二层交换机的端口处于不同的冲突域中，终端主机可以独占端口的带宽，所以交换式以太网的交换效率大大高于共享式以太网。

## 11.3 MAC地址学习



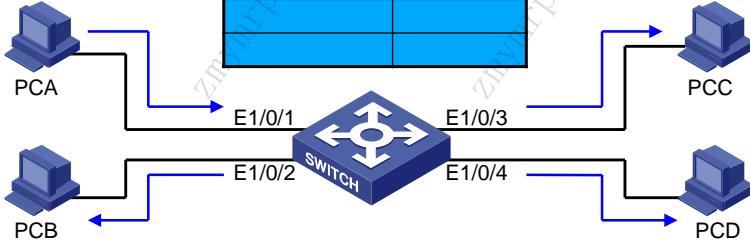
为了转发报文，以太网交换机需要维护 MAC 地址表。MAC 地址表的表项中包含了与本交换机相连的终端主机的 MAC 地址、本交换机连接主机的端口等信息。

在交换机刚启动时，它的 MAC 地址表中没有表项。此时如果交换机的某个端口收到数据帧，它会把数据帧从所有其它端口转发出去。这样，交换机就能确保网络中其它所有的终端主机都能收到此数据帧。但是，这种广播式转发的效率低下，占用了太多的网络带宽，并不是理想的转发模式。

为了能够仅转发目标主机所需要的数据，交换机就需要知道终端主机的位置，也就是主机连接在交换机的那个端口上。这就需要交换机进行 MAC 地址表的正确学习。

## MAC地址表学习过程（1）

MAC Address Table	
MAC Address	Port
MAC_A	E1/0/1



- PCA发出数据帧
- 交换机把PCA的帧中的源地址MAC\_A与接收到此帧的端口E1/0/1关联起来
- 交换机把PCA的帧从所有其他端口发送出去（除了接收到帧的端口E1/0/1）

www.h3c.com

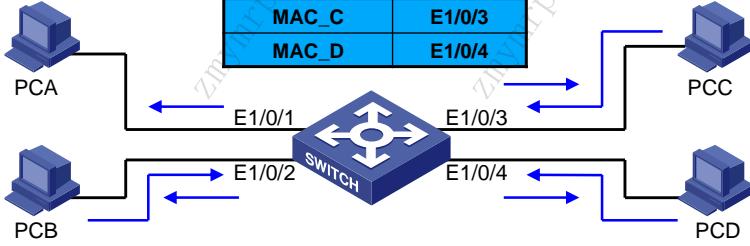
交换机通过记录端口接收数据帧中的源 MAC 地址和端口的对应关系来进行 MAC 地址表学习。

如上图，PCA 发出数据帧，其源地址是自己的地址 MAC\_A，目的地址是 PCD 的地址 MAC\_D。交换机在端口 E1/0/1 收到数据帧后，查看其中的源 MAC 地址，并添加到 MAC 地址表中，形成一条 MAC 地址表项。因为 MAC 地址表中没有 MAC\_D 的相关记录，所以交换机把此数据帧从所有其它端口都发送出去。

交换机在学习 MAC 地址时，同时给每条表项设定一个老化时间，如果在老化时间到期之前一直没有刷新，则表项会清空。交换机的 MAC 地址表空间是有限的，设定表项老化时间有助于回收长久不用的 MAC 表项空间。

## MAC地址表学习过程（2）

MAC Address Table	
MAC Address	Port
MAC_A	E1/0/1
MAC_B	E1/0/2
MAC_C	E1/0/3
MAC_D	E1/0/4



- PCB、PCC、PCD发出数据帧
- 交换机把接收到的帧中的源地址与相应的端口关联起来

www.h3c.com

同样的，当网络中其它 PC 发出数据帧时，交换机记录其中的源 MAC 地址，与接收到数据帧端口相关联起来，形成 MAC 地址表项。

当网络中所有的主机的 MAC 地址在交换机中都有记录后，意味着 MAC 地址学习完成，也可以说交换机知道了所有主机的位置。

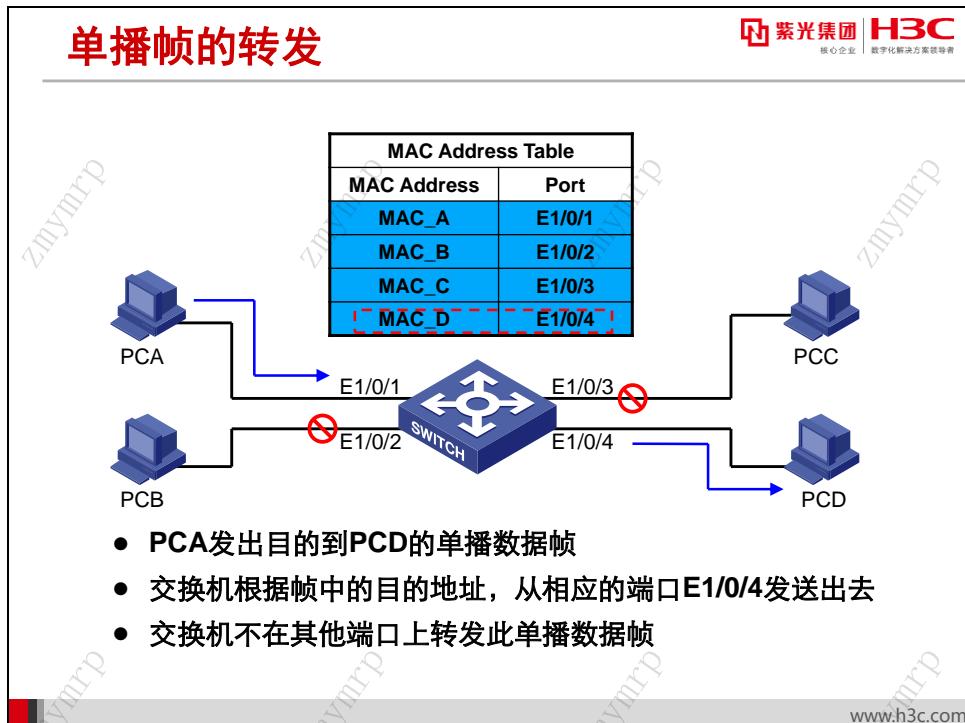
交换机在 MAC 地址学习时，需要遵循以下原则：

- 一个 MAC 地址只能被一个端口学习
- 一个端口可学习多个 MAC 地址

交换机进行 MAC 地址表学习的目的是知道主机所处的位置，所以只要有一个端口能到达主机就可以，多个端口到达主机反而造成带宽浪费，所以系统设定 MAC 地址只与一个端口关联。如果一个主机从一个端口转移到另一个端口，交换机在新的端口学习到了此主机 MAC 地址，则会删除原有表项。

一个端口上可关联多个 MAC 地址。比如端口连接到一个 Hub，Hub 连接多个主机，则此端口会关联多个 MAC 地址。

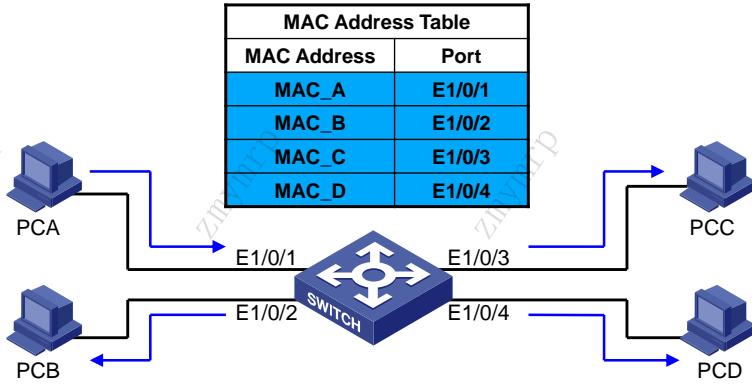
## 11.4 交换机对数据帧的转发和过滤



交换机根据 MAC 地址表项进行数据帧转发。

上图中，PCA 发出数据帧，其目的地址是 PCD 的地址 MAC\_D。交换机在端口 E1/0/1 收到数据帧后，检索 MAC 地址表项，发现目的 MAC 地址 MAC\_D 所对应的端口是 E1/0/4，就把此数据帧从 E1/0/4 转发，不在端口 E1/0/2 和 E1/0/3 转发，PCB 和 PCC 也不会收到目的到 PCD 的数据帧。

## 广播、组播和未知单播帧的转发



- 交换机会把广播、组播和未知单播帧从所有其他端口发送出去（除了接收到帧的端口）

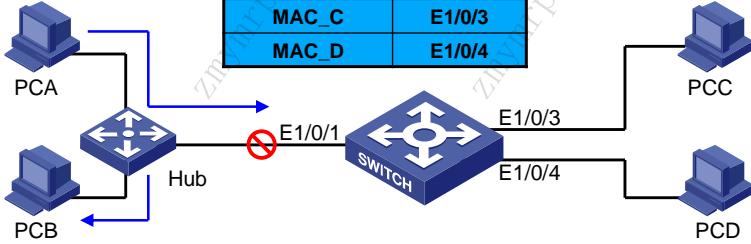
[www.h3c.com](http://www.h3c.com)

交换机需要把广播、组播帧从所有的端口转发出去（除了源端口）。因为广播和组播的目的就是要让网络中其他的成员收到这些数据帧。

未知单播帧是指帧的目的 MAC 地址在交换机 MAC 地址表中无相应表项的数据帧。由于 MAC 地址表中无相关表项，所以交换机也要把未知单播帧从其他端口转发出去，以便网络中其他主机能收到。

## 数据帧的过滤

MAC Address Table	
MAC Address	Port
MAC_A	E1/0/1
MAC_B	E1/0/1
MAC_C	E1/0/3
MAC_D	E1/0/4



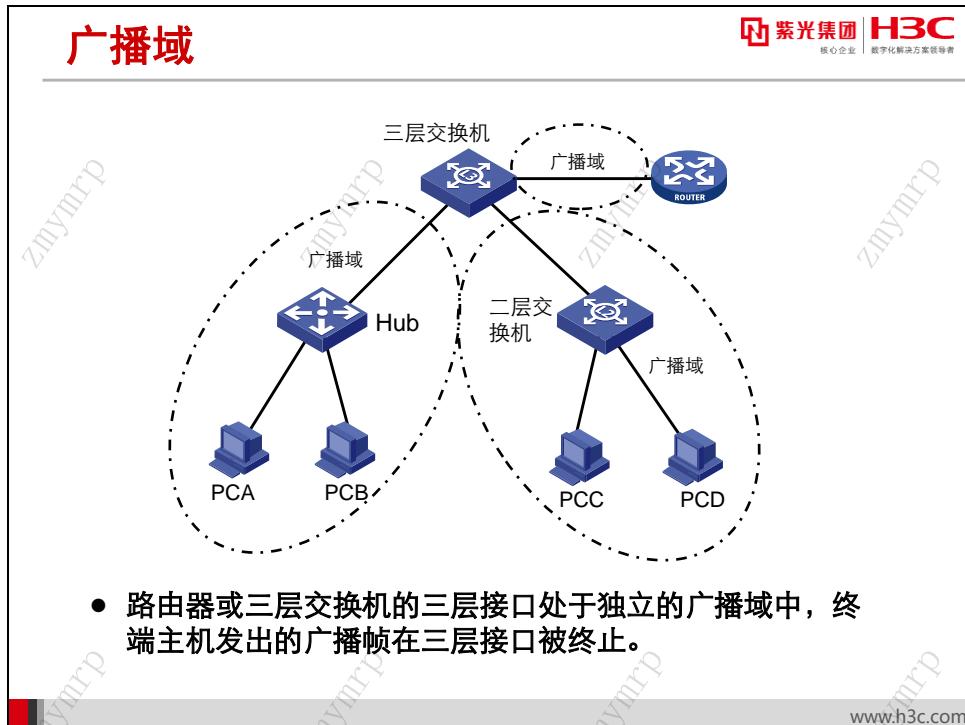
- PCA发出目的到PCB的数据帧
- 接收到帧时，交换机查找MAC地址表，如果发现帧的目的地址关联到接收端口上，则丢弃此帧

www.h3c.com

为了杜绝不必要的帧转发，交换机对符合特定条件的帧进行过滤。无论是单播、组播、广播帧，如果帧目的 MAC 地址在 MAC 地址表中有表项存在，且表项所关联的端口与接收到帧的端口相同时，则交换机对此帧进行过滤，即不转发此帧。

通常，帧过滤发生在一个端口学习到多个 MAC 地址的情况下。如上图所示，交换机端口 E1/0/1 连接有一个 Hub，所以端口 E1/0/1 上会同时学习到 PCA 和 PCB 的 MAC 地址。此时，PCA 和 PCB 之间进行数据通信时，尽管这些帧能够到达交换机的 E1/0/1 端口，交换机也不会转发这些帧到其它端口，而是将其丢弃了。

## 11.5 广播域



广播帧是指目的 MAC 地址是 FFFF.FFFF.FFFF 的数据帧，它的目的是要让本地网络中的所有设备都能收到。二层交换机需要把广播帧从除源端口之外的端口转发出去，所以二层交换机不能够隔离广播。

路由器或三层交换机是工作在网络层的设备，对网络层信息进行操作。路由器或三层交换机收到广播帧后，对帧进行解封装，取出其中的 IP 数据包，然后根据 IP 数据包中的 IP 地址进行路由。所以，路由器或三层交换机不会转发广播帧，广播在三层端口上被隔离。

广播域是指广播帧能够到达的范围。如上图中，PCA 发出的广播帧，PCB 能够收到，但 PCC 和 PCD 收不到，PCA 和 PCB 就属于同一个广播域。广播域中的设备数量越少，广播帧流量就越少，网络带宽的无谓消耗也越少。

通过在网络中使用三层交换机或路由器，可以减小广播域，减少网络带宽浪费。

## 11.6 本章总结

### 本章总结

- 共享式以太网中所有终端共享总线带宽，交换式以太网中每个终端处于独立的冲突域
- 交换机根据接收到的数据帧的源地址进行**MAC地址表**的学习
- 交换机根据**MAC地址表**对数据帧进行转发和过滤
- 路由器或三层交换机的三层接口属于独立的广播域

# 第12章 配置 VLAN

VLAN (Virtual Local Area Network, 虚拟局域网) 技术的出现，主要为了解决交换机在进行局域网互连时无法限制广播的问题。这种技术可以把一个物理局域网划分成多个虚拟局域网——VLAN，每个 VLAN 就是一个广播域，VLAN 内的主机间通信就和在一个 LAN 内一样，而 VLAN 间的主机则不能直接互通，这样，广播数据帧被限制在一个 VLAN 内。

## 12.1 本章目标

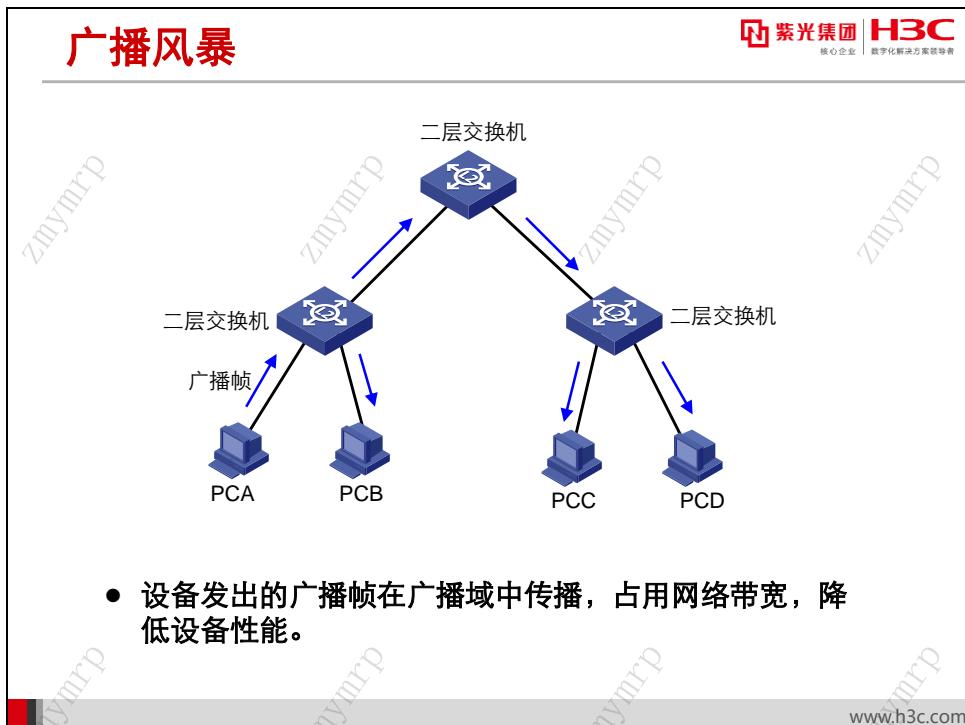
### 课程目标

● 学习完本课程，您应该能够：

- 了解VLAN技术产生的背景
- 掌握VLAN的类型及其相关配置
- 掌握IEEE 802.1Q的帧格式
- 掌握交换机端口的链路类型及其相关配置



## 12.2 VLAN技术简介

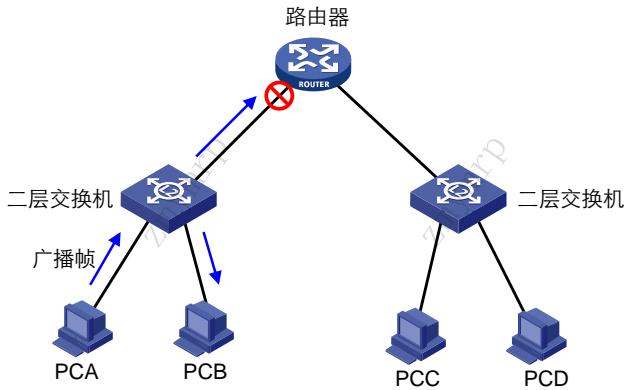


在交换式以太网出现后，同一个交换机下不同的端口处于不同的冲突域中，交换式以太网的效率大大增加。但是，在交换式以太网中，由于交换机所有的端口都处于一个广播域内，导致一台计算机发出的广播帧，局域网中所有的计算机都能够接收到，使局域网中的有限网络资源被无用的广播信息所占用。

上图中，四台终端主机发出的广播帧在整个局域网中广播，假如每台主机的广播帧流量是 100Kbps，则四台主机达到 400Kbps；如果链路是 100Mbps 带宽，则广播帧占用带宽达到 0.4%。但如果网络内主机达到 400 台，则广播流量将达到 40Mbps，占用带宽达到 40%，网络上到处充斥着广播流，网络带宽资源被极大的浪费。另外，过多的广播流量会造成网络设备及主机的 CPU 负担过重，系统反应变慢甚至死机。

如何降低广播域的范围，提升局域网的性能，是急需解决的问题。

## 用路由器来隔离广播



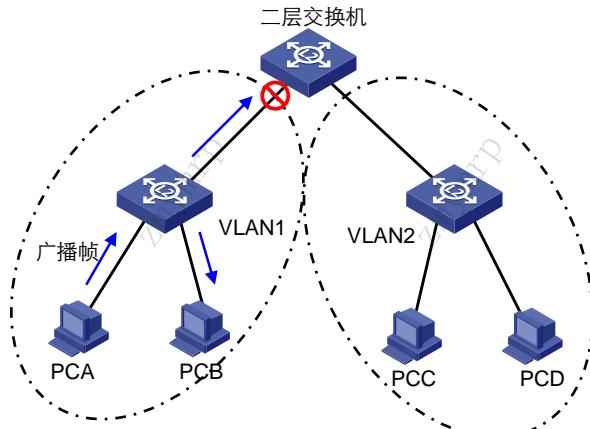
- 路由器能够隔离广播，减小广播域范围。

[www.h3c.com](http://www.h3c.com)

路由器的各个接口处于独立的广播域中，终端主机发出的广播帧在接口被终止。所以，在局域网中使用路由器能够隔离广播，减小广播域范围。

但是，路由器的价格比交换机要高，使用路由器提高了局域网的部署成本。另外，大部分中低端路由器使用软件转发，转发性能不高，容易在网络中造成性能瓶颈。所以，在局域网中使用路由器来隔离广播是一个高成本、低性能的方案。

## 用VLAN隔离广播



- 二层交换机使用VLAN隔离广播，减小广播域范围。

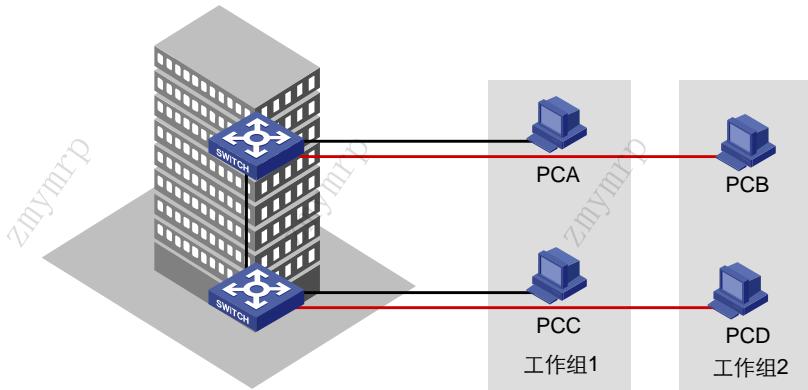
[www.h3c.com](http://www.h3c.com)

VLAN 技术的出现，就是为了解决交换机在进行局域网互连时无法限制广播的问题。这种技术可以把一个 LAN 划分多个逻辑的 LAN——VLAN，每个 VLAN 是一个广播域，不同 VLAN 间的设备不能直接互通，只能通过路由器等三层设备而互通。这样，广播数据帧被限制在一个 VLAN 内。

目前，绝大多数以太网交换机都能够支持 VLAN。使用 VLAN 来减小广播域的范围，减少 LAN 内的广播流量，是高效率、低成本的方案。

## VLAN的优点

 紫光集团 | **H3C**  
核心企业 | 数字化解决方案领导者



- **有效控制广播域范围**
- **增强局域网的安全性**
- **灵活构建虚拟工作组**

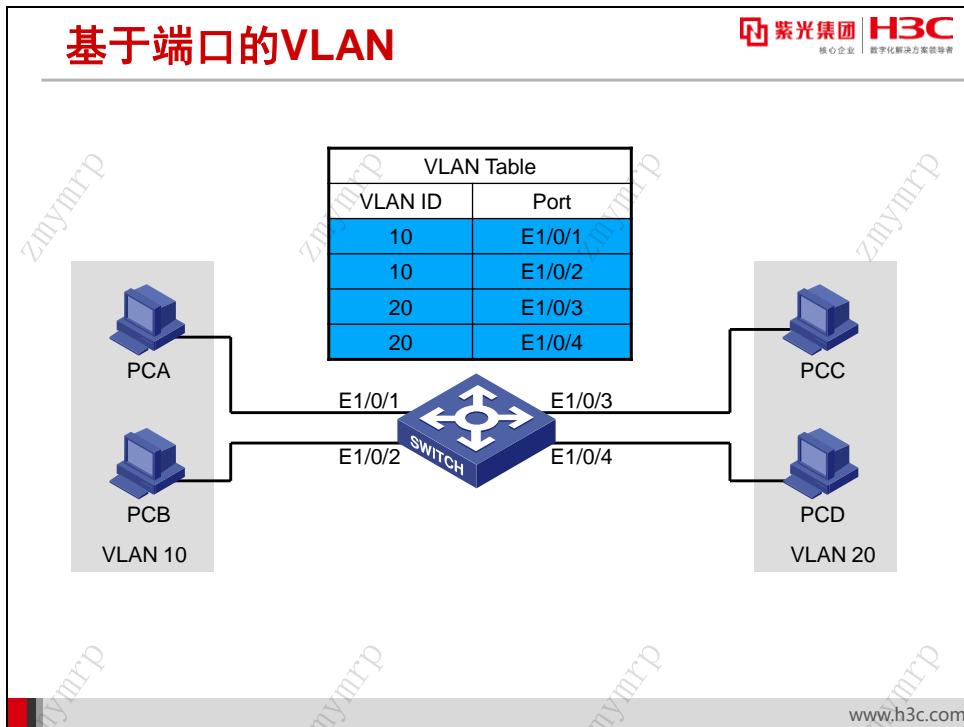
[www.h3c.com](http://www.h3c.com)

VLAN 的划分不受物理位置的限制。不在同一物理位置范围的主机可以属于同一个 VLAN；一个 VLAN 包含的用户可以连接在同一个交换机上，也可以跨越交换机，甚至可以跨越路由器。

VLAN 技术的优点如下：

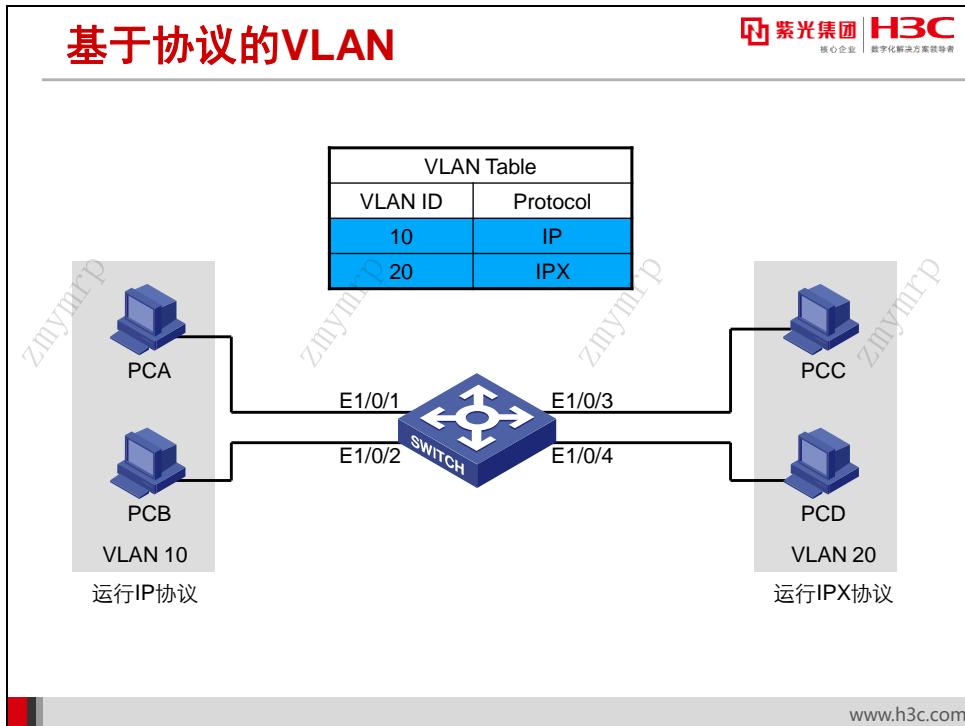
- **有效控制广播域范围：**广播域被限制在一个 VLAN 内，广播流量仅在 VLAN 中传播，节省了带宽，提高了网络处理能力。
- **增强局域网的安全性：**不同 VLAN 内的报文在传输时是相互隔离的，即一个 VLAN 内的用户不能和其它 VLAN 内的用户直接通信，如果不同 VLAN 要进行通信，则需要通过路由器或三层交换机等设备。
- **灵活构建虚拟工作组：**用 VLAN 可以划分不同的用户到不同的工作组，同一工作组的用户也不必局限于某一固定的物理范围，网络构建和维护更方便灵活。

## 12.3 VLAN类型



基于端口的 VLAN 是最简单、最有效的 VLAN 划分方法，它按照设备端口来定义 VLAN 成员。将指定端口加入到指定 VLAN 中之后，该端口就可以转发指定 VLAN 的数据帧。

上图中，交换机端口 **E1/0/1** 和 **E1/0/2** 被划分到 VLAN10 中，端口 **E1/0/3** 和 **E1/0/4** 被划分到 VLAN20 中，则 PCA 和 PCB 处于 VLAN10 中，可以互通；PCC 和 PCD 处于 VLAN20 中，可以互通。但 PCA 和 PCC 处于不同 VLAN，它们之间不能互通。



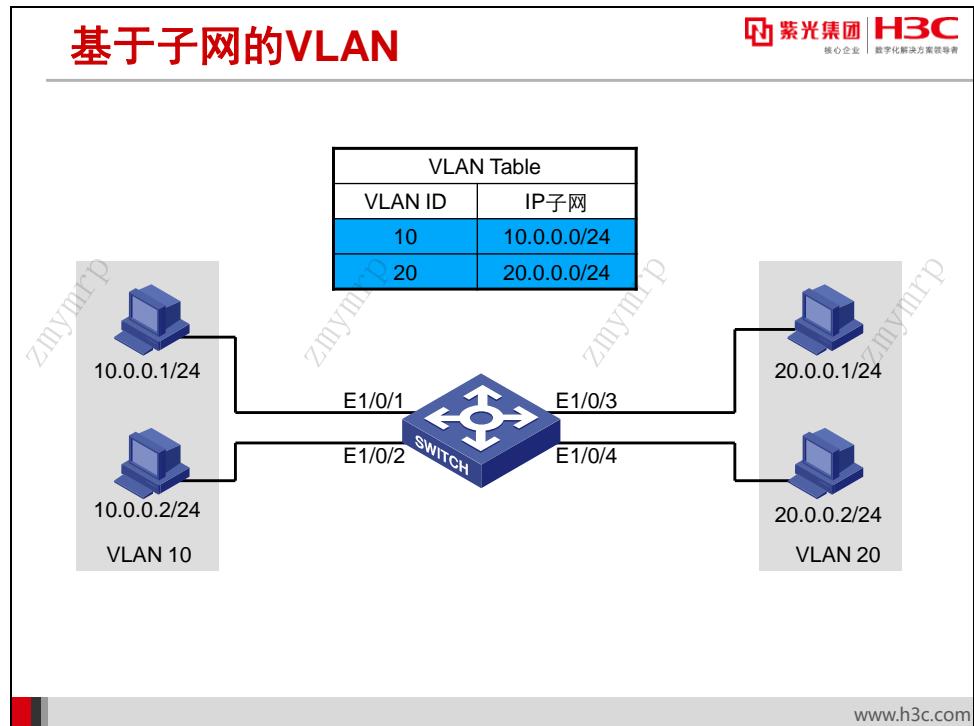
基于协议的 VLAN 是根据端口接收到的报文所属的协议（族）类型来给报文分配不同的 VLAN ID。可用来划分 VLAN 的协议族有 IP、IPX。

交换机从端口接收到以太网帧后，会根据帧中所封装的协议类型来确定报文所属的 VLAN，然后将数据帧自动划分到指定的 VLAN 中传输。

此特性主要应用于将网络中提供的协议类型与 VLAN 相绑定，方便管理和维护。

## 基于子网的VLAN

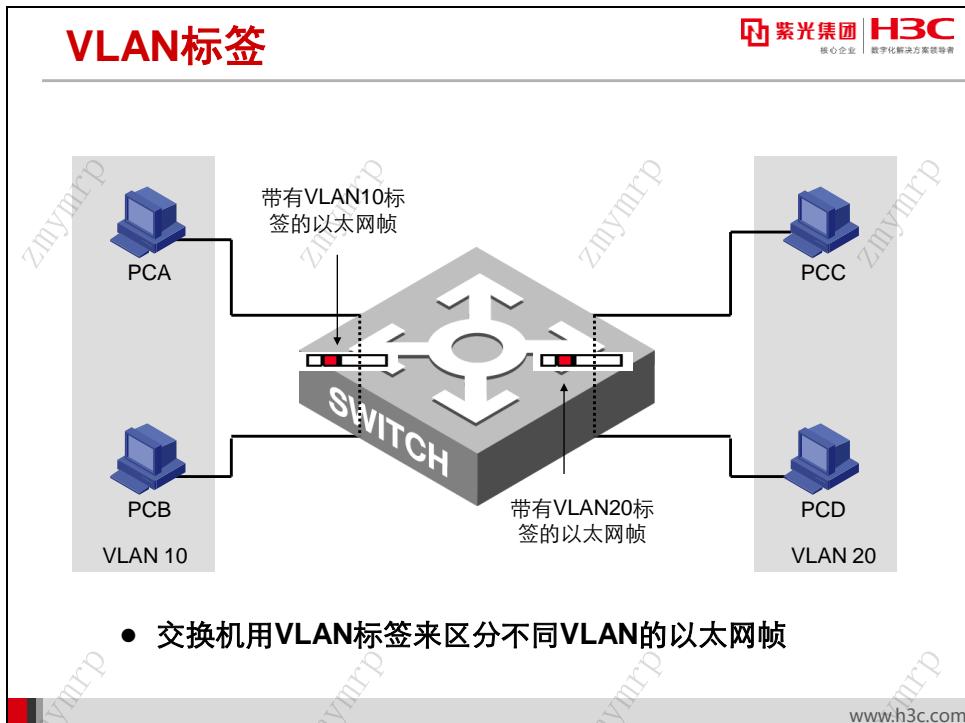
紫光集团 **H3C**  
核心企业 | 数字化解决方案领导者



基于 IP 子网的 VLAN 是根据报文源 IP 地址及子网掩码作为依据来进行划分的。设备从端口接收到报文后，根据报文中的源 IP 地址，找到与现有 VLAN 的对应关系，然后自动划分到指定 VLAN 中转发。

此特性主要用于将指定网段或 IP 地址发出的数据在指定的 VLAN 中传送。

## 12.4 VLAN技术原理

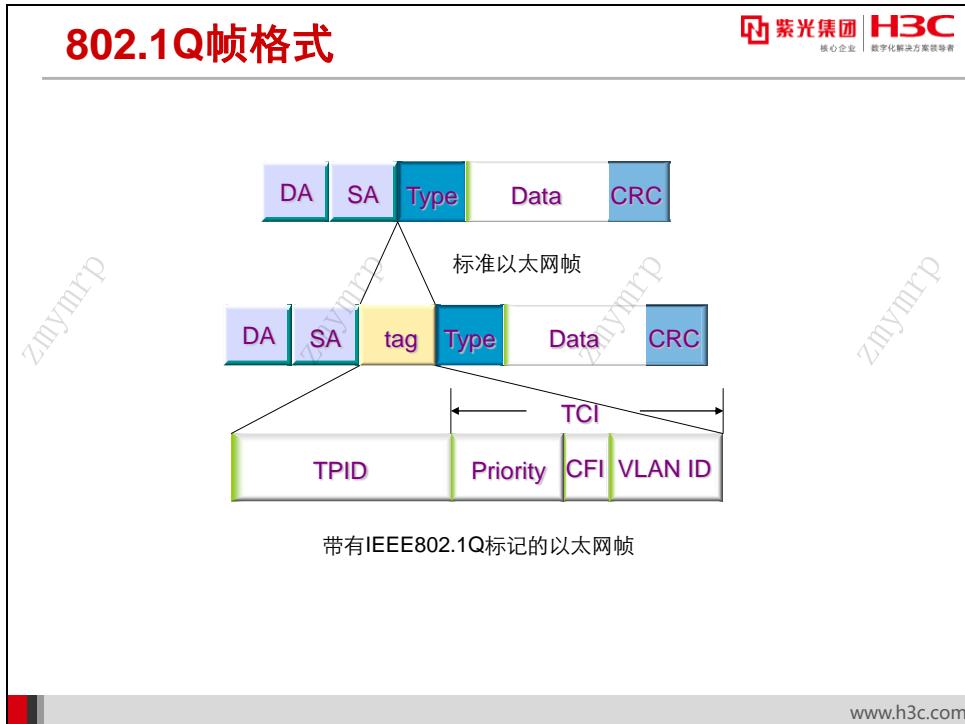


我们知道，以太网交换机根据 MAC 地址表来转发数据帧。MAC 地址表中包含了端口和端口所连接终端主机 MAC 地址的映射关系。交换机从端口接收到以太网帧后，通过查看 MAC 地址表来决定从哪一个端口转发出去。如果端口收到的是广播帧，则交换机把广播帧从除源端口外的所有端口转发出去。

在 VLAN 技术中，通过给以太网帧附加一个标签（Tag）来标记这个以太网帧能够在哪个 VLAN 中传播。这样，交换机在转发数据帧时，不仅要查找 MAC 地址来决定转发到哪个端口，还要检查端口上的 VLAN 标签是否匹配。

在上图中，交换机给主机 PCA 和 PCB 发来的以太网帧附加了 VLAN10 的标签，给 PCC 和 PCD 发来的以太网帧附加 VLAN20 的标签，并在 MAC 地址表中增加关于 VLAN 标签的记录。这样，交换机在进行 MAC 地址表查找转发操作时，会查看 VLAN 标签是否匹配；如果不匹配，则交换机不会从端口转发出去。这样相当于用 VLAN 标签把 MAC 地址表里的表项区分开来，只有相同 VLAN 标签的端口之间能够互相转发数据帧。

IEEE 在 802.1Q 中定义了在以太网帧中所附加标签的格式。



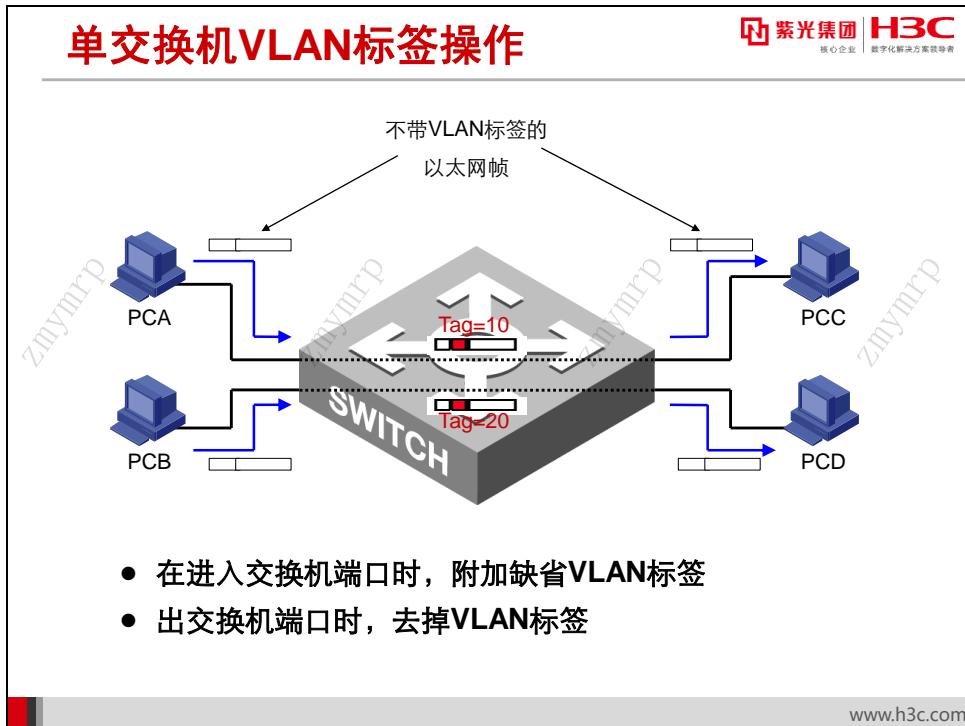
在传统的以太网帧中添加了 4 个字节的 802.1Q 标签后，成为带有 VLAN 标签的帧（Tagged Frame）。而传统的不携带 802.1Q 标签的数据帧称为未打标签的帧（Untagged Frame）。

802.1Q 标签头包含了 2 个字节的标签协议标识（TPID）和 2 个字节的标签控制信息（TCI）。

TPID（Tag Protocol Identifier）是 IEEE 定义的新的类型，表明这是一个封装了 802.1Q 标签的帧。TPID 包含了一个固定的值 0x8100。

TCI（Tag control Information）包含的是帧的控制信息，它包含了下面的一些元素：

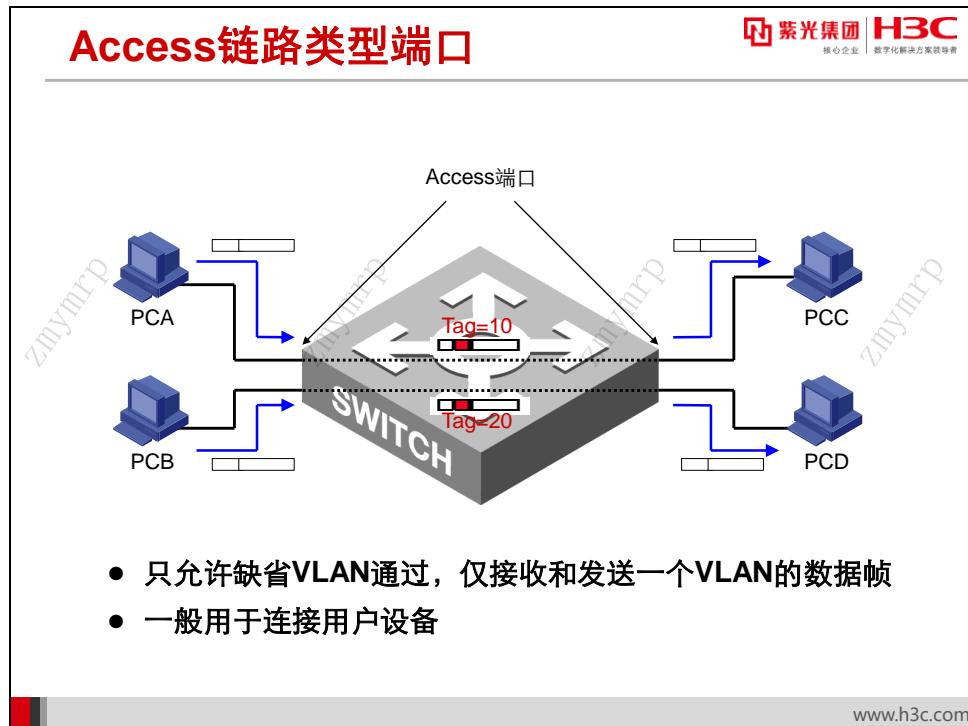
- **Priority:** 这 3 位指明数据帧的优先级。一共有 8 种优先级，0—7。
- **CFI (Canonical Format Indicator):** CFI 值为 0 说明是规范格式，1 为非规范格式。它被用在令牌环/源路由 FDDI 介质访问方法中来指示封装帧中所带地址的比特次序信息。
- **VLAN ID (VLAN Identifier):** 共 12 比特，指明 VLAN 的编号。VLAN 编号一共 4096 个，每个支持 802.1Q 协议的交换机发送出来的数据帧都会包含这个域，以指明自己属于哪一个 VLAN。



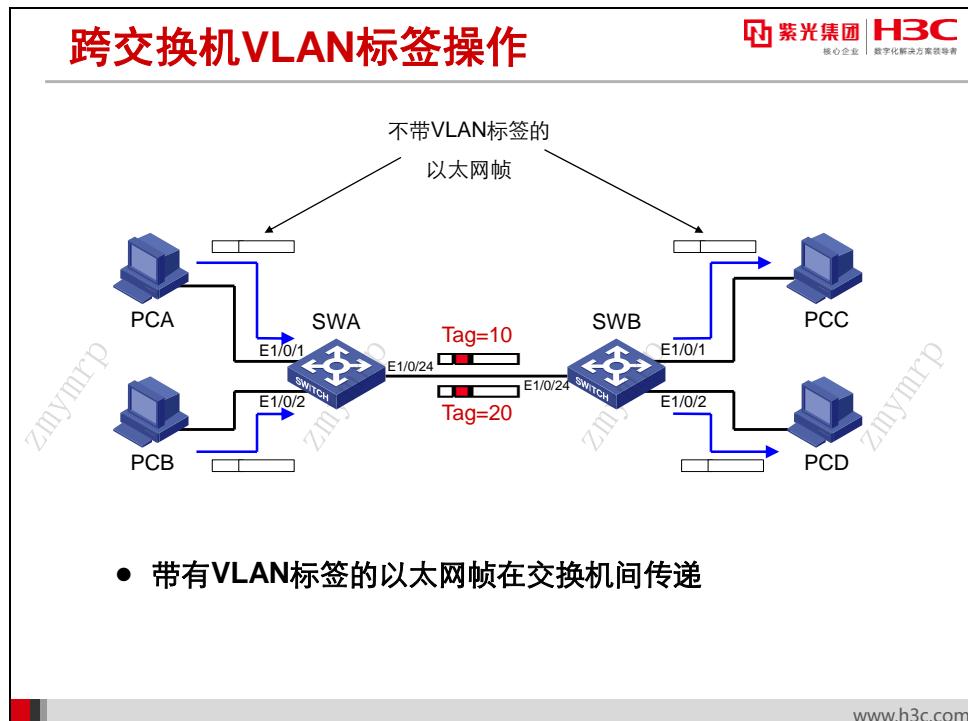
交换机根据数据帧中的标签来判定数据帧属于哪一个 VLAN，那么标签是从哪里来的呢？VLAN 标签是由交换机端口在数据帧进入交换机时添加的。这样做的好处是，VLAN 对终端主机是透明的，终端主机不需要知道网络中 VLAN 是如何划分的，也不需要识别带有 802.1Q 标签的以太网帧，所有的相关事情由交换机负责。

当终端主机发出的以太网帧到达交换机端口时，交换机检查端口所属的 VLAN，然后给进入端口的帧打相应的 802.1Q 标签。端口所属的 VLAN 称为端口默认 VLAN，又称为 PVID (Port VLAN ID)。

同样，为保持 VLAN 技术对主机透明，交换机负责剥离出端口的以太网帧的 802.1Q 标签。



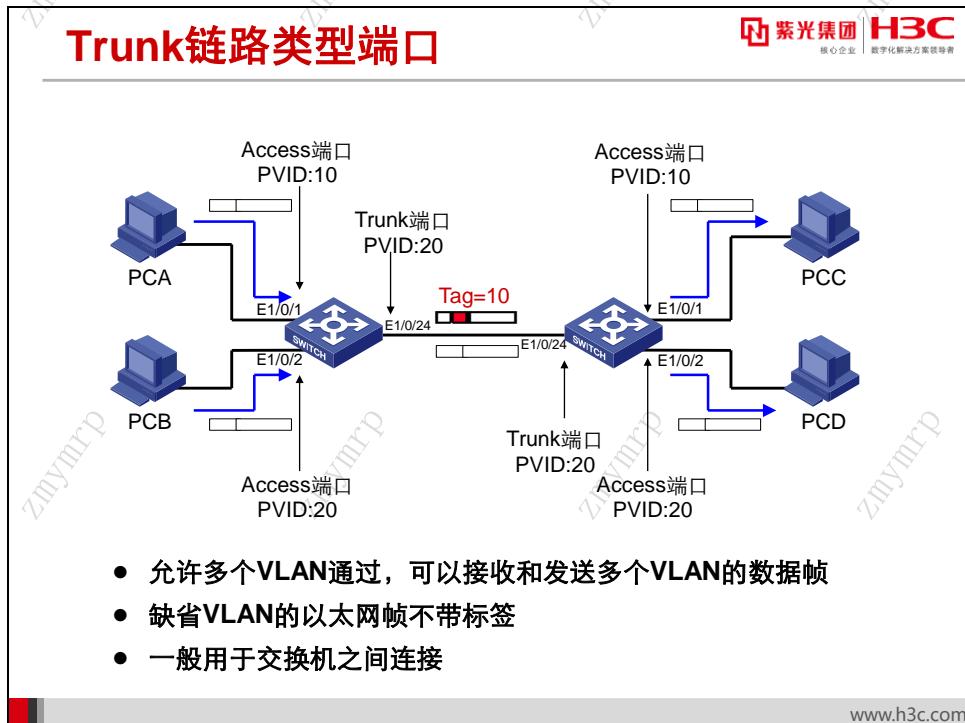
这种只允许默认 VLAN 的以太网帧通过的端口称为 Access 链路类型端口。Access 端口在收到以太网帧后打 VLAN 标签，转发出端口时剥离 VLAN 标签，对终端主机透明，所以通常用来连接不需要识别 802.1Q 协议的设备，如终端主机、路由器等。



VLAN 技术的很重要的功能是在网络中构建虚拟工作组，划分不同的用户到不同的工作组，同一工作组的用户也不必局限于某一固定的物理范围。通过在网络中实施跨交换机的 VLAN，能够实现虚拟工作组。

VLAN 跨越交换机时，需要交换机之间传递的以太网数据帧带有 802.1Q 标签。这样，数据帧所属的 VLAN 信息才不会丢失。

在上图中，PCA 和 PCB 所发出的数据帧分别打有 VLAN10 和 VLAN20 的标签，SWA 的端口 E1/0/24 负责对这些带 802.1Q 标签的数据帧进行转发，并不对其中的标签进行剥离操作。

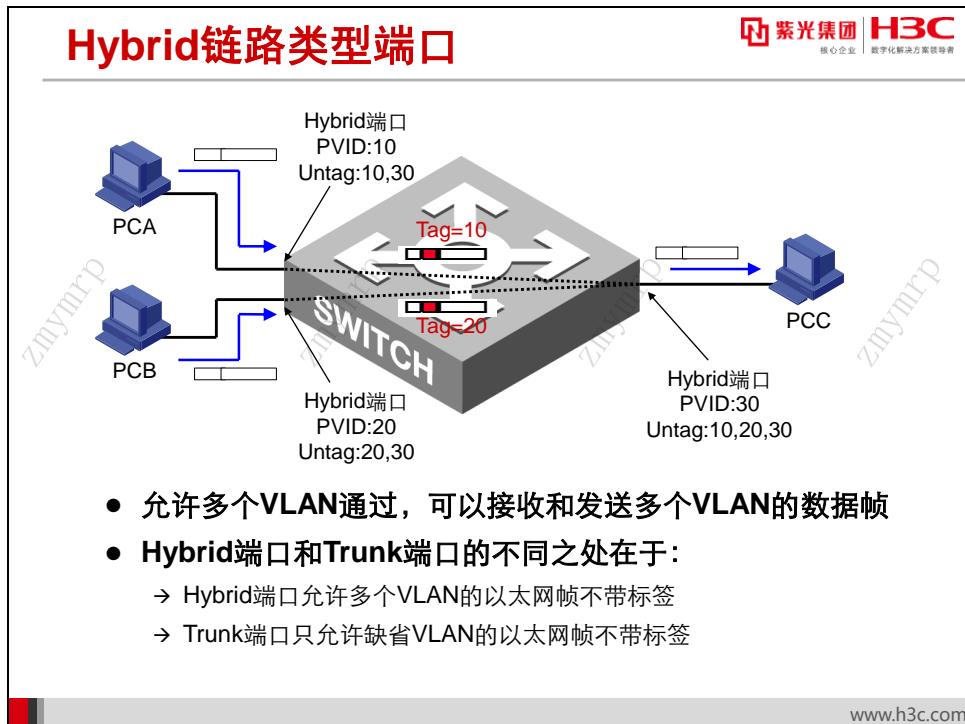


允许多个 VLAN 帧通过的端口称为 Trunk 链路类型端口。Trunk 端口可以接收和发送多个 VLAN 的数据帧，且在接收和发送过程中不对帧中的标签进行任何操作。

不过，默认 VLAN 帧是一个例外。在发送帧时，Trunk 端口要剥离默认 VLAN 帧中的标签；同样，交换机从 Trunk 端口接收到不带标签的帧时，要打上默认 VLAN 标签。

Trunk 端口一般用于在交换机之间互连。

图示为 PCA 至 PCC、PCB 至 PCD 的标签操作流程。PCA 发出以太网帧，到达 SWA 的 E1/0/1 端口，端口的默认 VLAN 是 10，所以以太网帧被打上 VLAN10 标签；E1/0/24 端口是 Trunk 端口，VLAN10 标签的帧从端口转发至 SWB；SWB 从帧中的标签得知它属于 VLAN10，于是转发至端口 E1/0/1，经剥离标签后到达 PCC。PCB 发出的帧在 E1/0/2 端口被打上 VLAN20 的标签；E1/0/24 端口是 Trunk 端口且默认 VLAN 是 20，所以数据帧被剥离标签后转发；当未带标签的数据帧到达 SWB 的 E1/0/24 端口后，端口给它打上 VLAN20 的标签再转发到端口 E1/0/2，端口 E1/0/2 剥离标签后转发至 PCD。



除了 Access 链路类型和 Trunk 链路类型端口外，交换机还支持第三种链路类型端口，称为 Hybrid 链路类型端口。Hybrid 端口可以接收和发送多个 VLAN 的数据帧，同时还能够指定对任何 VLAN 帧进行剥离标签操作。

当网络中大部分主机之间需要隔离，但这些隔离的主机又需要与另一台主机互通时，可以使用 Hybrid 端口。

在上图中，PCA 发出的以太网帧进入端口时打上 VLAN10 的标签，在到达连接 PCC 的端口时，端口根据设定（Untag: 10, 20, 30）将数据帧中的标签剥离后发送给 PCC，所以 PCA 与 PCC 能够通信；同理，PCB 也能够与 PCC 通信。但 PCA 发出的以太网帧到达连接 PCB 的端口时，端口上的设定（Untag: 20, 30）表明只对 VLAN20、VLAN30 的数据帧转发且剥离标签，而不允许 VLAN10 的帧通过，所以 PCA 与 PCB 不能互通。

## 12.5 VLAN的基本配置

**VLAN基本配置**

● 创建VLAN并进入VLAN视图

**[Switch] vlan *vlan-id***

● 将指定端口加入到当前VLAN中

**[Switch-vlan10] port *interface-list***

www.h3c.com

默认情况下，交换机只有 VLAN1，所有的端口都属于 VLAN1 且是 Access 链路类型端口。进行 VLAN 配置的基本步骤如下。

**第1步：**在系统视图下创建 VLAN 并进入 VLAN 视图。配置命令为：

**vlan *vlan-id***

**第2步：**在 VLAN 视图下将指定端口加入到 VLAN 中。配置命令为：

**port *interface-list***

## 配置Trunk端口

- 配置端口的链路类型为Trunk类型

```
[Switch-Ethernet1/0/1] port link-type trunk
```

- 允许指定的VLAN通过当前Trunk端口

```
[Switch-Ethernet1/0/1] port trunk permit vlan  
{ vlan-id-list | all }
```

- 设置Trunk端口的缺省VLAN

```
[Switch-Ethernet1/0/1] port trunk pvid vlan vlan-id
```

www.h3c.com

Trunk 端口能够允许多个 VLAN 的数据帧通过，通常用于在交换机之间互连。配置某个端口成为 Trunk 端口的步骤如下。

**第1步：**在以太网端口视图下指定端口链路类型为 Trunk。配置命令为：

**port link-type trunk**

**第2步：**默认情况下，Trunk 端口只允许默认 VLAN 即 VLAN1 的数据帧通过。所以，需要在以太网端口视图下指定哪些 VLAN 帧能够通过当前 Trunk 端口。配置命令为：

**port trunk permit vlan { vlan-id-list | all }**

**第3步：**必要时，可以在以太网端口视图下设定 Trunk 端口的默认 VLAN。配置命令为：

**port trunk pvid vlan vlan-id**

### 注意

默认情况下，Trunk 端口的默认 VLAN 是 VLAN1。可以根据实际情况进行修改默认 VLAN，以保证两端交换机的默认 VLAN 相同为原则，否则会发生同一 VLAN 内的主机跨交换机不能够通信的情况。

## 配置 Hybrid 端口

- 配置端口的链路类型为 Hybrid 类型

```
[Switch-Ethernet1/0/1] port link-type hybrid
```

- 允许指定的 VLAN 通过当前 Hybrid 端口

```
[Switch-Ethernet1/0/1] port hybrid vlan vlan-id-list  
{ tagged | untagged }
```

- 设置 Hybrid 端口的缺省 VLAN

```
[Switch-Ethernet1/0/1] port hybrid pvid vlan vlan-id
```

www.h3c.com

在某些情况下，需要用到 Hybrid 端口。Hybrid 端口也能够允许多个 VLAN 帧通过，并且还可以指定哪些 VLAN 数据帧被剥离标签。配置某个端口成为 Hybrid 端口的步骤如下。

**第1步：**在以太网端口视图下指定端口链路类型为 Hybrid。配置命令为：

**port link-type hybrid**

**第2步：**默认情况下，所有 Hybrid 端口只允许 VLAN1 通过。所以，需要在以太网端口视图下指定哪些 VLAN 数据帧能够通过 Hybrid 端口，并指定是否剥离标签。配置命令为：

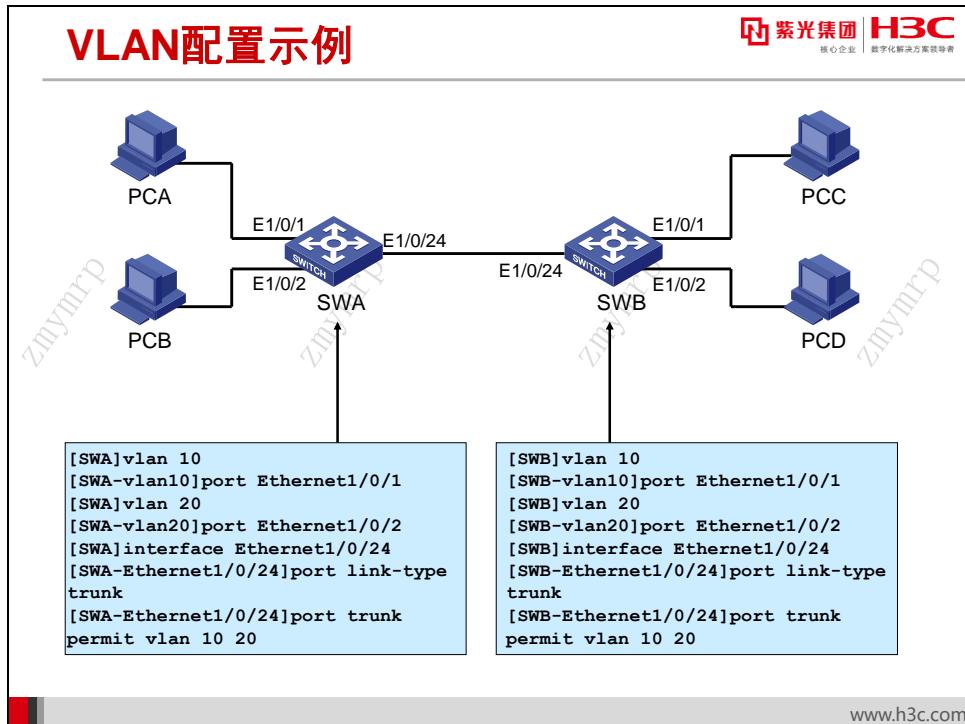
**port hybrid vlan vlan-id-list { tagged | untagged }**

**第3步：**在以太网端口视图下设定 Hybrid 端口的默认 VLAN。配置命令为：

**port hybrid pvid vlan vlan-id**

### 注意

Trunk 端口不能直接被设置为 Hybrid 端口，只能先设为 Access 端口，再设置为 Hybrid 端口。



上图是 VLAN 的基本配置示例。图中 PCA 与 PCC 属于 VLAN10, PCB 与 PCD 属于 VLAN20, 交换机之间使用 Trunk 端口相连, 端口的默认 VLAN 是 VLAN1。

#### 配置 SWA:

```
[SWA]vlan 10
[SWA-vlan10]port Ethernet1/0/1
[SWA]vlan 20
[SWA-vlan20]port Ethernet1/0/2
[SWA]interface Ethernet1/0/24
[SWA-Ethernet1/0/24]port link-type trunk
[SWA-Ethernet1/0/24]port trunk permit vlan 10 20
```

#### 配置 SWB:

```
[SWB]vlan 10
[SWB-vlan10]port Ethernet1/0/1
[SWB]vlan 20
[SWB-vlan20]port Ethernet1/0/2
[SWB]interface Ethernet1/0/24
[SWB-Ethernet1/0/24]port link-type trunk
[SWB-Ethernet1/0/24]port trunk permit vlan 10 20
```

配置完成后, PCA 与 PCC 能够互通, PCB 与 PCD 能够互通; 但 PCA 与 PCB, PCC 与 PCD 之间不能够互通。

**VLAN显示及维护**

紫光集团 | H3C  
核心企业 | 数字化解决方案领导者

```

<Switch> display vlan
VLAN function is enabled.
Total 3 VLAN exist(s).
Now, the following VLAN exist(s):
1 (default), 2, 10 ← 当前交换机存在的VLAN

<Switch> display vlan 2
VLAN ID: 2
VLAN Type: static
Route interface: not configured
Description: VLAN 0002
Tagged Ports: none ← VLAN中哪些端口打标签
Untagged Ports:
Ethernet1/0/1 Ethernet1/0/3 Ethernet1/0/4 ← VLAN中哪些端口不打标签

<Switch> display interface ethernet 1/0/1
.....
PVID: 2 ← 当前端口的缺省VLAN
Mdi type: auto
Port link-type: access ← 当前端口链路类型为access
Tagged VLAN ID : none
Untagged VLAN ID : 2
Port priority: 0
.....

```

www.h3c.com

在任意视图下可以使用 **display vlan** 命令来查看交换机当前启用的 VLAN。

### **display vlan**

由图中可以看到，目前交换机上有 VLAN1、VLAN2、VLAN10 存在，VLAN1 是默认 VLAN。

如果要查看某个具体 VLAN 所包含的端口，可以使用 **display vlan vlan-id** 命令。

### **display vlan vlan-id**

由图中可以看到，VLAN2 中包含了 Ethernet1/0/1、Ethernet1/0/3 和 Ethernet1/0/4 等 3 个端口，且 VLAN 数据帧离开这些端口时需要剥离标签。

如果要查看具体端口的 VLAN 信息，可以使用 **display interface** 命令。

### **display interface interface-type interface-number**

由图中可知，端口 Ethernet1/0/1 的端口链路类型为 Access，默认 VLAN(Pvid)是 VLAN1。如果是 Trunk 或 Hybrid 端口，则还会显示哪些 VLAN 帧是携带标签通过，哪些 VLAN 帧需要剥离标签。

## 12.6 本章总结

### 本章总结

- **VLAN**的作用是限制局域网中广播传送的范围；
- 通过对以太网帧进行打标签操作，交换机区分不同**VLAN**的数据帧；
- 交换机的端口链路类型分为**Access**、**Trunk**和**Hybrid**。

# 第13章 生成树协议

一个局域网通常由多台交换机互连而成，为了避免广播风暴，我们需要保证在网络中不存在路径回环，也就是说所有链路应该组成一棵无回环的树，交换机上的 STP（生成树协议）就实现了这样的功能。在本章中我们首先会学习有关 STP 协议的一些基本概念，以及 STP 协议是如何通过实现冗余链路的闭塞和开启从而实现一棵动态的生成树；最后我们还会介绍一下 RSTP（快速生成树协议）、PVST 和 MSTP（多生成树协议），以及如何在交换机上对生成树进行配置。

## 13.1 本章目标

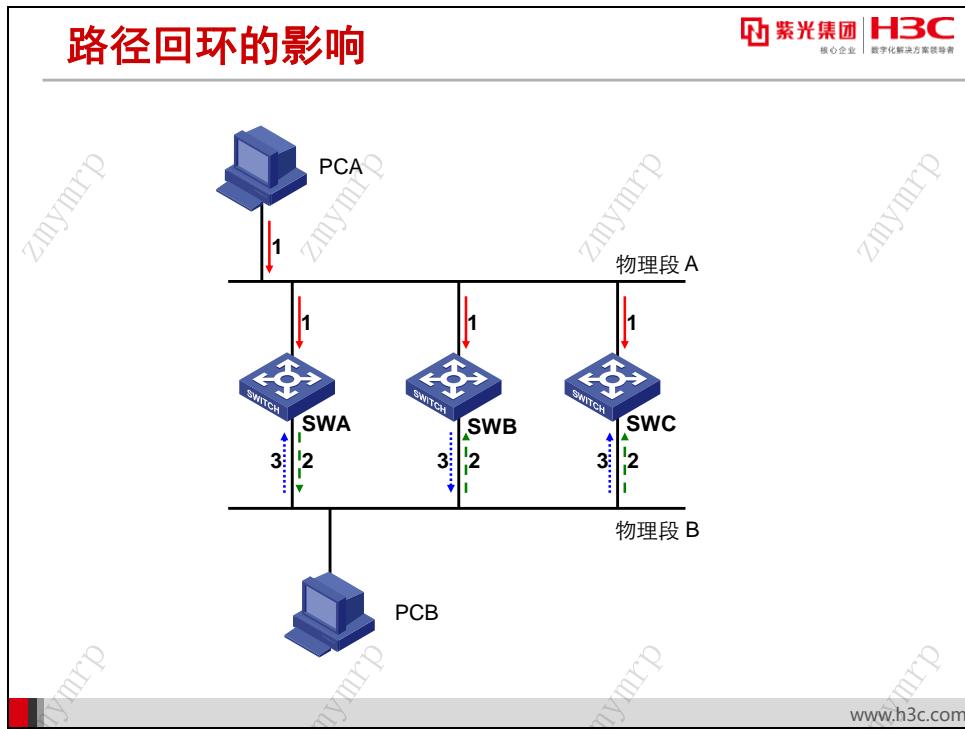
### 课程目标

学习完本课程，您应该能够：

- 了解STP产生的背景
- 掌握STP基本工作原理
- 掌握RSTP、PVST和MSTP基本原理
- 掌握生成树协议的配置



## 13.2 STP产生背景



在桥接网络中，网桥不会对以太网数据帧做任何修改，帧中也不会记录到底经过了几个网桥。如果网络中存在环路，帧有可能在环路中不断循环和增生，造成网络带宽被大量重复帧占据，导致网络拥塞。

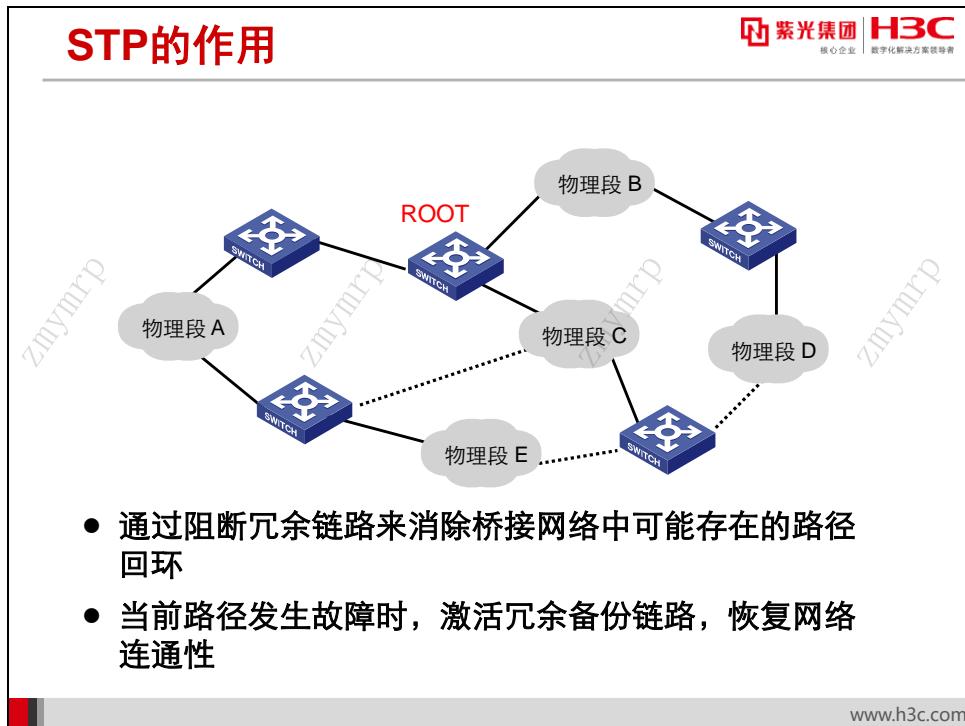
上图中是一个由于环路造成数据帧循环和增生的例子。

开始，假定 PCA 还没有发送过任何帧，因此网桥 SWA、SWB 和 SWC 的地址表中都没有 PCA 的地址记录。

当 PCA 发送了一个帧，最初三个网桥都接收了这个帧，记录 PCA 的地址在物理段 A 上，并将这个帧转发到物理段 B 上。

网桥 SWA 会将此帧转发到物理段 B 上，从而 SWB 和 SWC 将会再次接收到这个帧。因为 SWA 对于 SWB 和 SWC 来说是透明的，这个帧就好像是 PCA 在物理段 B 上发送的一样。于是 SWB 和 SWC 记录 PCA 在物理段 B 上，并将这个新帧转发到物理段 A 上。

同样的道理，SWB 会将最初的帧转发到物理段 B 上，那么 SWA 和 SWC 都接收到这个帧。SWC 认为 PCA 仍然在物理段 B 上，而 SWA 又发现 PCA 已经转移到物理段 B 上了，然后 SWA 和 SWC 都会转发新帧到物理段 A 上。如此下去，帧就在环路中不断循环，更糟糕的是每次成功的帧发送都会导致网络中出现两个新帧。



尽管透明网桥存在这个隐患，但是它的应用还是相当有诱惑力的，因为透明网桥在无回路的网络中发挥的作用是无可指摘的。那么是不是就认为我们不能组建有回路的网络呢？这显然是不合适的，因为回路的存在可以在拓扑结构的某条链路断开之后，仍然保证网络的连通性。

为此，我们找到了一种很好的算法，它通过阻断冗余链路将一个有回路的桥接网络修剪成一个无回路的树型拓扑结构，这样既解决了回路问题，又能在某条活动（active）的链路断开时，通过激活被阻断的冗余链路重新修剪拓扑结构以恢复网络的连通。

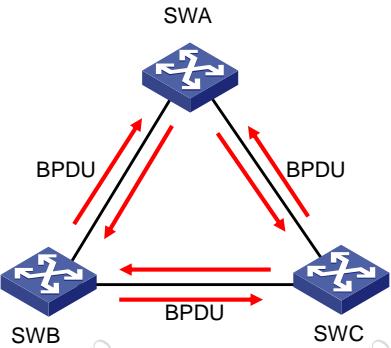
上面的图中给出了一个应用生成树的桥接网络的例子，其中字符 ROOT 所标识的网桥是生成树的树根，实线是活动的链路，也就是生成树的枝条，而虚线则是被阻断的冗余链路，只有在活动链路断开时才会被激活。

## 13.3 STP

**生成树协议**


  
 紫光集团 H3C  
核心企业 | 数字化解决方案领导者

- **STP (Spanning Tree Protocol, 生成树协议)** 是用于在局域网中消除数据链路层物理环路的协议。
- 通过在桥之间交换**BPDU (Bridge Protocol Data Unit, 桥协议数据单元)**，来保证设备完成生成树的计算过程。



www.h3c.com

STP (Spanning Tree Protocol, 生成树协议) 是由 IEEE 协会制定的，用于在局域网中消除数据链路层物理环路的协议，其标准名称为 802.1D。运行该协议的设备通过彼此交互信息发现网络中的环路，并有选择的对某些端口进行阻塞，最终将环路网络结构修剪成无环路的树型网络结构，从而防止报文在环路网络中不断增生和无限循环，避免设备由于重复接收相同的报文造成报文处理能力下降的问题发生。

STP 包含了两个含义，狭义的 STP 是指 IEEE 802.1D 中定义的 STP 协议，广义的 STP 是指包括 IEEE 802.1D 定义的 STP 协议以及各种在它的基础上经过改进的生成树协议，如 RSTP、MSTP。

STP 采用的协议报文是 BPDU (Bridge Protocol Data Unit, 桥协议数据单元)，BPDU 中包含了足够的信息来完成生成树的计算。

BPDU 在 STP 协议中分为两类：

- 配置 BPDU (Configuration BPDU): 用来进行生成树计算和维护生成树拓扑的报文。
- TCN BPDU (Topology Change Notification BPDU): 当拓扑结构发生变化时，用来通知相关设备网络拓扑结构发生变化的报文。

## 配置BPDU的生成和传递

- 配置BPDU包含以下重要信息，完成生成树计算
  - 根桥ID (RootID)
  - 根路径开销 (RootPathCost)
  - 指定桥ID (DesignatedBridgeID)
  - 指定端口ID (DesignatedPortID)
- 各台设备的各个端口在初始时生成以自己为根桥 (**Root Bridge**) 的配置消息，向外发送自己的配置消息
- 网络收敛后，根桥向外发送配置BPDU，其他的设备对该配置BPDU进行转发

[www.h3c.com](http://www.h3c.com)

STP 协议的配置 BPDU 报文携带了如下几个重要信息：

- 根桥 ID (RootID)

由根桥的优先级和 MAC 地址组成。通过比较 BPDU 中的根桥 ID，STP 最终决定谁是根桥。

- 根路径开销 (RootPathCost)

到根桥的路径开销。根端口选举时，开销最小的端口被选举为根端口；指定桥选举时，开销最小的桥被选举为指定桥。

- 指定桥 ID (DesignatedBridgeID)

根端口选举时，所连桥 ID 最小的端口被选举为根端口。指定桥选举时，桥 ID 最小的桥被选举为指定桥。

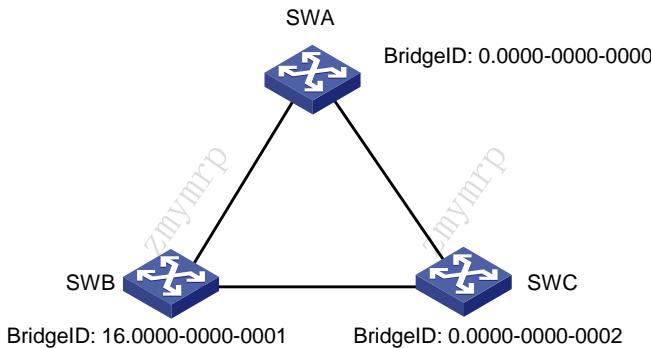
- 指定端口 ID (DesignatedPortID)

根端口选举时，所连端口 ID 最小的端口被选举为根端口。

各台设备的各个端口在初始时会生成以自己为根桥的配置消息，根路径开销为 0，指定桥 ID 为自身设备 ID，指定端口为本端口。各台设备都向外发送自己的配置消息，同时也会收到其他设备发送的配置消息。通过比较这些配置消息，交换机进行生成树计算，选举根桥，决定端口角色。

网络收敛后，根桥会按照一定的时间间隔产生并向外发送配置 BPDU，其他的设备对该配置 BPDU 进行转发，从而保证拓扑的稳定。

## 根桥的选举



- 桥 ID 由桥优先级 (BridgePriority) 和桥 MAC 地址 (BridgeMacAddress) 组成
- 桥 ID 小的桥被选举为根桥

[www.h3c.com](http://www.h3c.com)

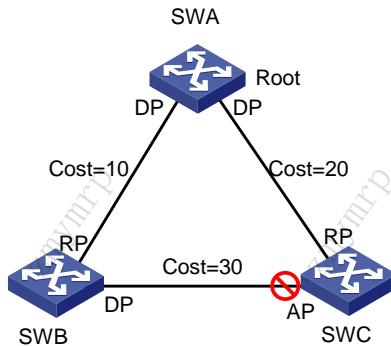
树形的网络结构，必须要有树根，于是 STP 引入了根桥 (Root Bridge) 的概念。

网络中每台设备都有自己的桥 ID，桥 ID 由桥优先级 (BridgePriority) 和桥 MAC 地址 (BridgeMacAddress) 两部分组成。因为桥 MAC 地址在网络中是唯一的，所以能够保证桥 ID 在网络中也是唯一的。在进行桥 ID 比较时，先比较优先级，优先级值小者为优；在优先级相等的情况下，再用 MAC 地址来进行比较，MAC 地址小者为优。

网络初始化时，网络中所有的 STP 设备都认为自己是“根桥”。设备之间通过交换配置 BPDU 而比较桥 ID，网络中桥 ID 最小的设备被选为根桥。根桥会按照一定的时间间隔产生并向外发送配置 BPDU，其他的设备对该配置 BPDU 进行转发，从而保证拓扑的稳定。

在上图中，因为 SWA 的桥 ID 最小，所以 SWA 被选举为根桥。

## 端口角色的确定



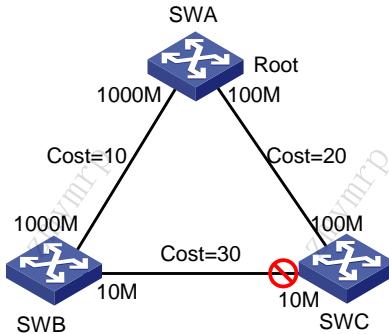
- 根桥上的所有端口为指定端口（Designated Port）
- 在非根桥上选举根路径开销（RootPathCost）最小的端口为根端口（Root Port）
- 每个物理段选出根路径开销最小的桥作为指定桥（Designated Bridge），连接指定桥的端口为指定端口
- 不是根端口和指定端口的其余端口被STP置为阻塞状态

[www.h3c.com](http://www.h3c.com)

STP 的作用是通过阻断冗余链路使一个有回路的桥接网络修剪成一个无回路的树型拓扑结构。它通过将环路上的某些端口置为阻塞状态，不允许数据帧通过而做到这一点。下面是确定哪些端口是阻塞状态的过程：

- 1) 根桥上的所有端口为指定端口（Designated Port）；
- 2) 为每个非根桥选择根路径开销（RootPathCost）最小的那个端口作为根端口（Root Port），该端口到根桥的路径是此网桥到根桥的最佳路径；
- 3) 为每个物理段选出根路径开销最小的那个网桥作为指定桥（Designated Bridge），该指定桥到该物理段的端口作为指定端口，负责所在物理段上的数据转发；
- 4) 既不是指定端口也不是根端口的端口置于阻塞状态。

## 根路径开销



- 根路径开销（RootPathCost）是到达根的路径上所有链路开销（Cost）的代数和
- 非根桥进行根端口选举时，根路径开销最小的端口为根端口
- 物理段进行指定桥选举时，路径开销最小的桥为指定桥

[www.h3c.com](http://www.h3c.com)

根路径开销（RootPathCost）是生成树协议中用来判定到达根的距离的参数。它是到达根的路径上所有链路开销（Cost）的代数和。

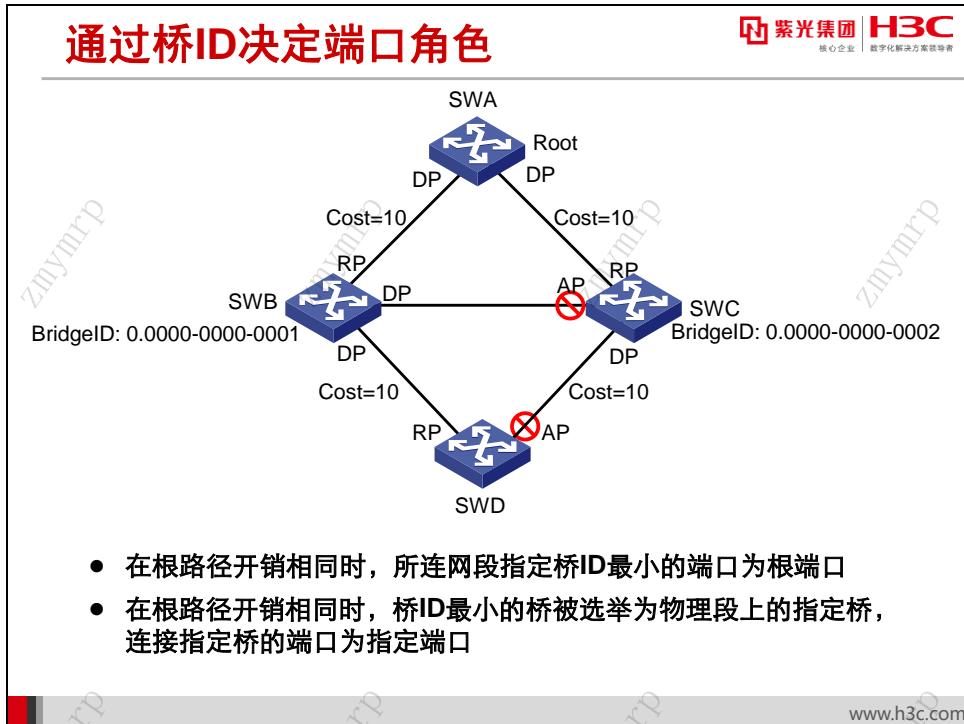
非根桥进行根端口选举时，会首先比较各端口的根路径开销，开销最小的端口被选举为根端口；物理段进行指定桥选举时，也会首先比较各桥的根路径开销，开销最小的桥被选举为指定桥。

通常情况下，链路的开销与物理带宽成反比。带宽越大，表明链路通过能力越强，则路径开销越小。

IEEE 802.1D 和 802.1t 定义了不同速率和工作模式下的以太网链路（端口）开销，H3C 则根据实际的网络运行状况优化了开销的数值定义，制定了私有标准。上述三种标准的常用定义如表所示。其他细节定义请参照相关标准文档及设备手册。

链路速率	802.1D-1998	802.1t	私有标准
0	65535	200,000,000	200,000
10Mbps	100	2,000,000	2,000
100Mbps	19	200,000	200
1000Mbps	4	20,000	20
10Gbps	2	2,000	2

H3C 交换机默认采用私有标准定义的链路开销。交换机端口的链路开销可手工设置，以影响生成树的选路。



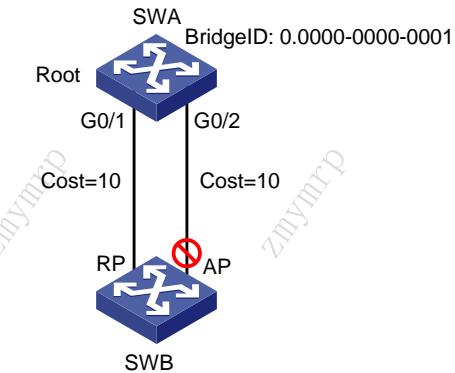
在根路径开销相同的情况下，生成树协议根据桥 ID 来决定端口角色。

当一个非根桥上有多个端口经过不同的上游桥到达根桥，且这些路径的根路径开销相同时，协议会比较各端口的上游指定桥 ID，所连上游指定桥 ID 最小的端口被选举为根端口。当一个物理段有多个网桥到根桥的路径开销相同，进行指定桥选举时，也比较这些网桥的桥 ID，桥 ID 最小的桥被选举为指定桥。

上图中，SWD 有 2 个端口能到达根，且根路径开销是相同的。但因 SWB 的桥 ID 小于 SWC 的桥 ID，所以连接 SWB 的端口为根端口。同样，SWB 被选举为 SWB 和 SWC 之间物理网段的指定桥，相连端口为指定端口。

因为桥 ID 是唯一的，所以通过比较桥 ID 可以对经过多个桥到达根桥的路径好坏进行最终判定。

## 通过端口ID决定端口角色



- 在根路径开销、指定桥 ID 都相同的情况下，所连指定端口 ID 小的端口为根端口

[www.h3c.com](http://www.h3c.com)

在根路径开销和指定桥 ID 都相同的情况下，生成树协议根据端口 ID 来决定端口角色。

如果非根桥上多个端口经过相同的上游桥到达根，且根路径开销相同，则协议会比较端口所连上游桥的端口 ID，所连端口 ID 最小的端口被选举为根端口。

端口 ID 由端口索引号和端口优先级两部分组成。在进行比较时，先比较端口优先级，优先级小的端口优先；在优先级相同时，再比较端口索引号，索引号小的端口优先。

上图中，SWB 上的 2 个端口连接到 SWA，这 2 个端口的根路径开销相同，上游指定桥 ID 也相同，协议根据上游指定端口 ID 来判定，所连指定端口 ID 小的端口为根端口。

在通常情况下，端口索引号无法改变，用户可通过设置端口优先级来影响生成树的选路。

## 端口状态

紫光集团 | H3C  
核心企业 | 数字化解决方案领导者

端口角色	端口状态	端口行为
未启用STP功能的端口	<b>Disabled</b>	不收发BPDU报文，接收或转发数据
非指定端口或根端口	<b>Blocking</b>	接收但不发送BPDU，不接收或转发数据
--	<b>Listening</b>	接收并发送BPDU，不接收或转发数据
--	<b>Learning</b>	接收并发送BPDU，不接收或转发数据
指定端口或根端口	<b>Forwarding</b>	接收并发送BPDU，接收并转发数据

www.h3c.com

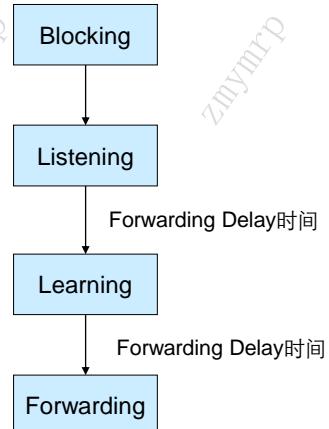
事实上，在 802.1D 的协议中，端口共有五种状态：

- **Disabled:** 表示该端口处于失效状态，不接收和发送任何报文。这种状态可以是由于端口的物理状态(比如端口物理层没有 up)导致的，也可能是管理者手工将端口关闭。
- **Blocking:** 处于这个状态的端口不能够参与转发数据报文，但是可以接收配置消息，并交给 CPU 进行处理。不过不能发送配置消息，也不进行地址学习。
- **Listening:** 处于这个状态的端口也不参与数据转发，不进行地址学习；但是可以接收并发送配置消息。
- **Learning:** 处于这个状态的端口同样不能转发数据，但是开始地址学习，并可以接收、处理和发送配置消息。
- **Forwarding:** 一旦端口进入该状态，就可以转发任何数据了，同时也进行地址学习和配置消息的接收、处理和发送。

以上五种状态中，**Listening** 和 **Learning** 是不稳定的中间状态。

## 端口状态迁移

- 端口被选为指定端口或根端口后，需要从 Blocking 状态经 Listening 和 Learning 才能到 Forwarding 状态
- 默认的 Forwarding Delay 时间是 15 秒



[www.h3c.com](http://www.h3c.com)

在一定条件下，端口状态之间是可以互相迁移的。

当一个端口由于拓扑发生改变不再是根端口或指定端口了，就会立刻迁移到 Blocking 状态。

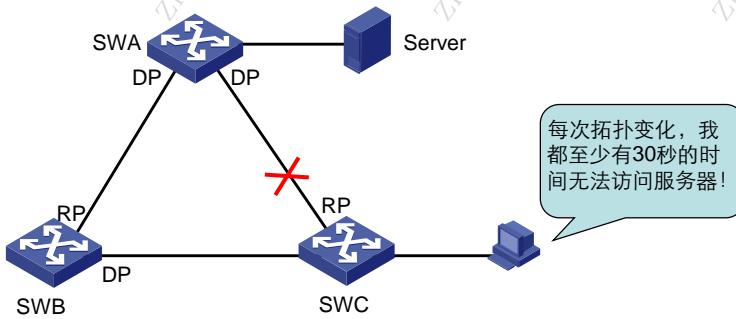
当一个端口被选为根端口或指定端口，就会从 Blocking 状态迁移到一个中间状态 Listening 状态；经历 Forwarding Delay 时间，迁移到下一个中间状态 Learning 状态；再经历一个 Forwarding Delay 时间，迁移到 Forwarding 状态。

从 Listening 迁移到 Learning，或者从 Learning 迁移到 Forwarding 状态，都需要经过 Forwarding Delay 时间，通过这种延时迁移的方式，能够保证当网络的拓扑发生改变时，新的配置消息能够传遍整个网络，从而避免由于网络未收敛而造成临时环路。

在 802.1D 中，默认的 Forwarding Delay 时间是 15 秒。

## 生成树的不足

- 端口从阻塞状态进入转发状态必须经历两倍的 **Forwarding Delay** 时间
- 如果网络中的拓扑结构变化频繁，网络会频繁地失去连通性



www.h3c.com

在前面我们介绍了有关 STP 的一些特性。在实际的应用中，STP 也有很多美中不足的地方。最主要的缺点是端口从阻塞状态到转发状态需要两倍的 **Forwarding Delay** 时间，导致网络的连通性至少要几十秒的时间之后才能恢复。如果网络中的拓扑结构变化频繁，网络会频繁的失去连通性，这样用户就会无法忍受。

为了在拓扑变化后网络尽快恢复连通性，交换机在 STP 的基础上发展出 RSTP（快速生成树）。

## 13.4 RSTP

### RSTP



- **RSTP (Rapid Spanning Tree Protocol, 快速生成树协议) 是 STP 协议的优化版**
- **RSTP 具备 STP 的所有功能**
- **RSTP 可以实现快速收敛**

→ 在某些情况下，端口进入转发状态的延时大大缩短，从而缩短了网络最终达到拓扑稳定所需要的时间。

[www.h3c.com](http://www.h3c.com)

RSTP (Rapid Spanning Tree Protocol, 快速生成树协议) 是 STP 协议的优化版。IEEE 802.1w 定义了 RSTP。RSTP 是从 STP 算法的基础上发展而来，承袭了它的基本思想，即也是通过配置消息来传递生成树信息，并通过优先级比较来进行计算。

RSTP 能够完成生成树的所有功能，不同之处就在于：在某些情况下，当一个端口被选为根端口或指定端口后，RSTP 减小了端口从阻塞到转发的时延，尽可能快的恢复网络连通性，提供更好的用户服务。

## RSTP的改进

	STP行为	RSTP行为
端口被选为根端口	默认情况下，2倍的Forwarding Delay的时间延迟。	存在阻塞的备份根端口情况下，仅有数毫秒延迟。
端口被选为指定端口	默认情况下，2倍的Forwarding Delay的时间延迟。	在指定端口是非边缘端口的情况下，延迟取决因素较多。
		在指定端口是边缘端口的情况下，指定端口可以直接进入转发状态，没有延迟。

www.h3c.com

RSTP 从三个方面实现“快速”功能：

### 1) 端口被选为根端口

交换机上原来有两个端口能够到达根桥，其中一个端口为根端口，另外一个为备用的端口（处于阻塞状态）。因某种原因，原根端口不再是根端口，而原来是阻塞状态的端口被选为根端口时，故障恢复的时间就是根端口的切换时间，无需延时，无需传递 BPDU，只是一个 CPU 处理的延时，约几毫秒。

### 2) 端口被选为非边缘指定端口

此时情况较复杂。“非边缘”的意思是这个端口连接着其他的交换机，而不是只连接到终端设备。此时如果交换机之间是点对点链路，则交换机需要发送握手报文到其它交换机进行协商，只有对端返回一个赞同报文后，端口才能进入转发状态。

可见点对点链路对 RSTP 的性能有很大的影响，下面列举了点对点链路的几种情况：

- 该端口是一个链路聚合端口；（请参考相关章节的描述）
- 该端口支持自协商功能，并通过协商工作在全双工模式；（请参考相关章节的描述）
- 管理者将该端口配置为一个全双工模式的端口。

如果是非点对点链路，则恢复时间与 STP 无异，是两倍的 Forwarding Delay 时间，默认情况下是 30 秒。

在 RSTP 握手协商时，总体收敛时间取决于网络直径。最坏的情况是，握手从网络的一边开始扩散到网络的另一边。比如网络直径为 7 的情况，最多要经过六次握手，网络的连通性才能被恢复。

### 3) 端口被选为边缘指定端口

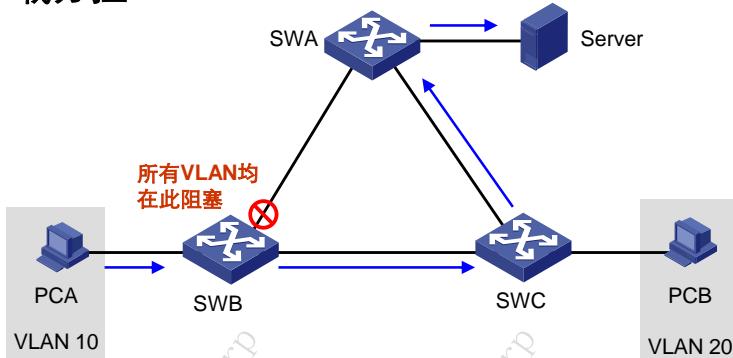
“边缘端口”是指那些直接和终端设备相连，不再连接任何交换机的端口。这些端口无需参与生成树计算，端口可以无时延的快速进入转发状态。此时不会造成任何的环路。

## 13.5 STP、RSTP 的不足

### STP、RSTP 的问题

紫光集团 | H3C  
核心企业 | 数字化解决方案领导者

- Trunk 链路上实际上运行着多个 VLAN
- 所有 VLAN 共用一棵生成树
- 无法实现不同 VLAN 在多条 Trunk 链路上的负载分担



www.h3c.com

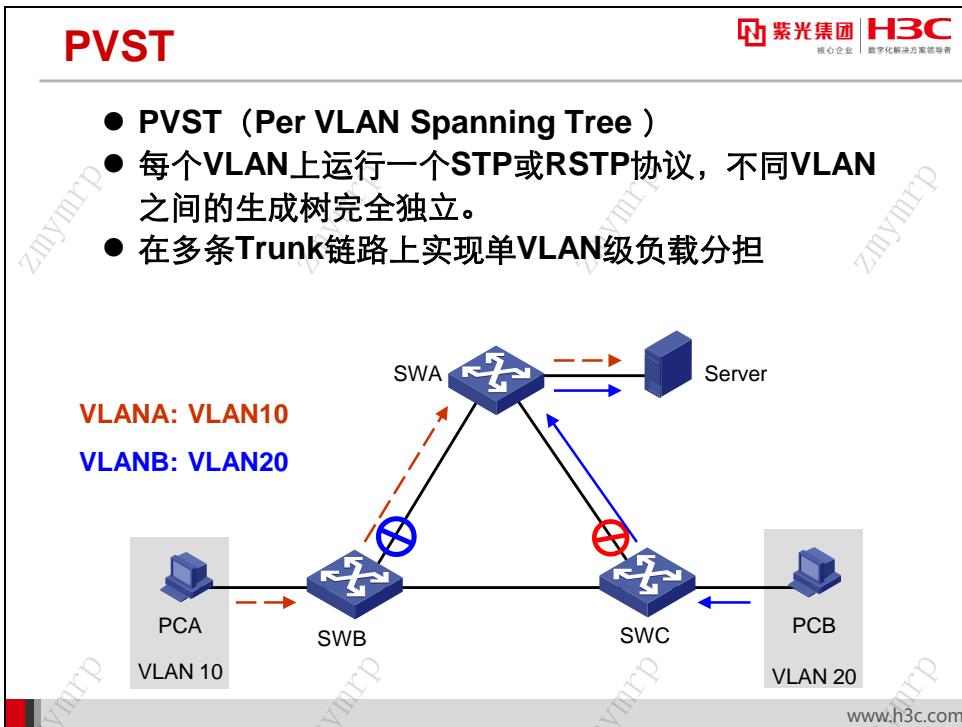
STP 使用生成树算法，能够在交换网络中避免环路造成的故障，并实现冗余路径的备份功能。RSTP 则进一步提高了交换网络拓扑变化时的收敛速度。

然而当前的交换网络往往工作在多 VLAN 环境下。在 802.1q 封装的 Trunk 链路上，同时存在多个 VLAN，每个 VLAN 实质上是一个独立的 2 层交换网络。为了给所有的 VLAN 提供环路避免和冗余备份功能，就必须为所有的 VLAN 都提供生成树计算。

传统 STP/RSTP 采用的方法是使用统一的生成树。所有的 VLAN 共享一棵生成树（CST, Common spanning tree），其拓扑结构也是一致的。因此在一条 Trunk 链路上，所有的 VLAN 要么全部处于转发状态，要么全部处于阻塞状态。

在图示情况下，SWB 到 SWA 的端口被阻塞，则从 PCA 到 Server 的所有数据都需要经过 SWB 至 SWC 至 SWA 的路径传递。SWB 至 SWA 之间的带宽完全浪费了。

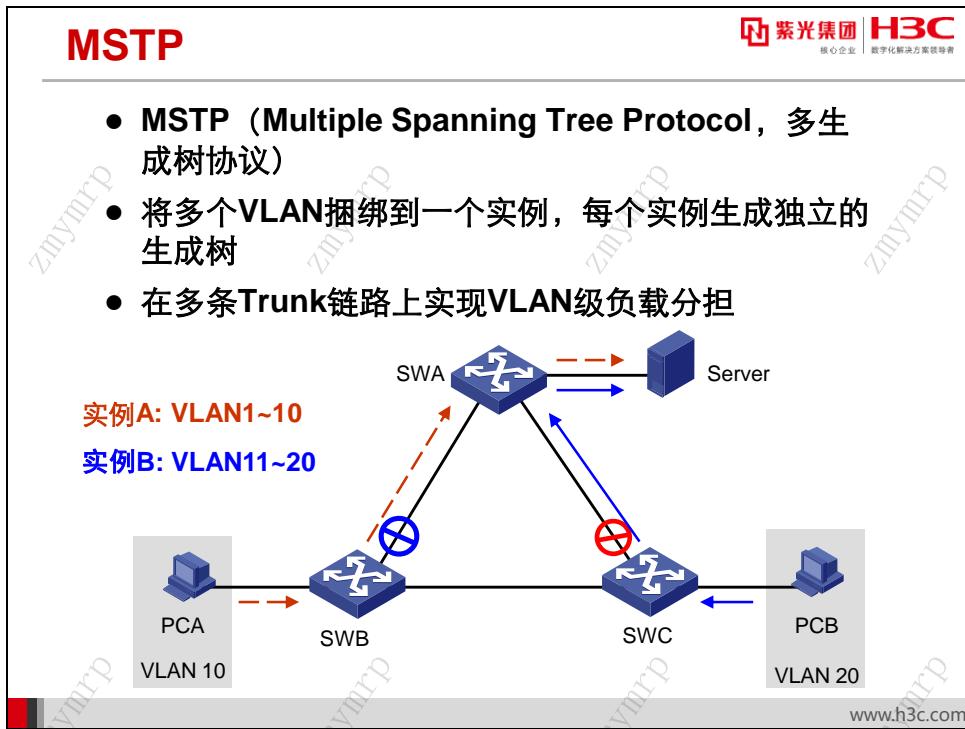
## 13.6 PVST



STP 和 RSTP 在局域网内的所有网桥都共享一棵生成树，不能按 VLAN 阻塞冗余链路，所有 VLAN 的报文都沿着一棵生成树进行转发。而 PVST 则可以在每个 VLAN 内都拥有一棵生成树，能够有效地提高链路带宽的利用率。PVST 可以简单理解为在每个 VLAN 上运行一个 STP 或 RSTP 协议，不同 VLAN 之间的生成树完全独立。

运行 PVST 的 H3C 设备可以与运行 Rapid PVST 或 PVST 的友商设备互通。当运行 PVST 的 H3C 设备之间互联，或运行 PVST 的 H3C 设备与运行 Rapid PVST 的友商设备互联时，H3C 设备支持像 RSTP 一样的快速收敛。

## 13.7 MSTP



IEEE 802.1s 定义的 MSTP (Multiple Spanning Tree Protocol) 可以实现 VLAN 级负载均衡。

通过 MSTP 协议，我们可以在网络中定义多个生成树实例，每个实例对应多个 VLAN，每个实例维护自己的独立生成树。这样既避免了为每个 VLAN 维护一棵生成树的巨大资源消耗，又可以使不同的 VLAN 具有完全不同的生成树拓扑，不同 VLAN 在同一端口上可以具有不同的状态，从而可以实现 VLAN 一级的负载分担。

图中，PCA 属于 VLAN10，VLAN10 绑定到实例 A 中； SWB 至 SWA 之间的链路在实例 A 中是连通的，所以 PCA 到 Server 的数据帧就经过 SWB 至 SWA 之间的路径传递。同理，PCB 属于 VLAN20，VLAN20 绑定到实例 B 中； PCB 到 Server 的数据帧就经过 SWC 至 SWA 之间的路径传递。可以看出，此网络通过 MSTP 而实现不同 VLAN 的数据流有不同的转发路径。

## 13.8 四种生成树协议的比较

### 四种生成树协议特性的比较

 紫光集团 | H3C  
核心企业 | 数字化解决方案领导者

特性列表	STP	RSTP	PVST	MSTP
解决环路故障并实现冗余备份	Y	Y	Y	Y
快速收敛	N	Y	Y	Y
形成多棵生成树实现负载分担	N	N	Y	Y

- MSTP、PVST 具有 RSTP 的快速收敛，同时又具有负载分担机制
- MSTP 兼容 STP 和 RSTP

www.h3c.com

STP 可以在交换网络中形成一棵无环路的树，解决环路故障并实现冗余备份。

RSTP 在 STP 功能基础上，通过使根端口快速进入转发状态、采用握手机制和设置边缘端口等方法，提供了更快的收敛速度。

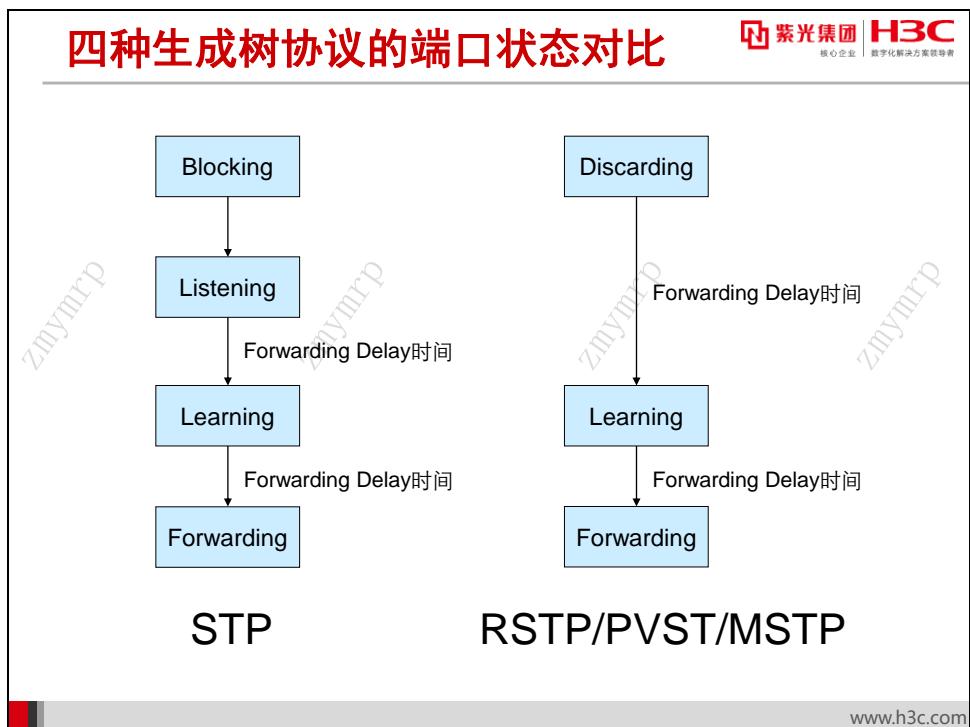
MSTP、PVST 则可以在大规模、多 VLAN 环境下形成多个生成树实例，从而高效地提供多 VLAN 负载均衡。

PVST 模式与其他模式的兼容性如下：

- 对于 Access 端口：PVST 模式在任意 VLAN 中都能与其他模式互相兼容。
- 对于 Trunk 端口或 Hybrid 端口：PVST 模式仅在 VLAN 1 中能与其他模式互相兼容。

MSTP 同时兼容 STP、RSTP。STP、RSTP 两种协议报文都可以被运行 MSTP 的设备识别并应用于生成树计算。

## 四种生成树协议的端口状态对比


[www.h3c.com](http://www.h3c.com)

另外，RSTP/PVST/MSTP 与 STP 的端口状态也有所不同，从 STP 的 5 种变成 3 种，其对应关系如下：

STP 端口状态	RSTP/PVST/MSTP 端口状态
Disabled	Discarding
Blocking	Discarding
Listening	Discarding
Learning	Learning
Forwarding	Forwarding

在 RSTP/PVST/MSTP 中，取消了 Listening 这个中间状态，并且把 Disabled、Blocking、Listening 三种状态合并为一种 Discarding，减少状态数量，简化生成树计算，加快收敛速度。

## 13.9 生成树协议的基本配置

**STP基本配置**

**● 开启设备STP特性**

**[Switch] stp global enable**

**● 关闭端口的STP特性**

**[Switch-Ethernet1/0/1] undo stp enable**

**● 配置STP的工作模式**

**[Switch] stp mode { stp | rstp | pvst | mstp }**

www.h3c.com

交换机的生成树功能在默认情况下是处于关闭状态的。如果组网中需要通过环路设计来提供网络的冗余容错的能力，而同时又需要防止路径回环的产生，就需要用到生成树的概念。我们可以在系统视图下开启生成树功能：

**[Switch] stp global enable**

如果不需要生成树，则可以在系统视图下关闭生成树功能：

**[Switch] undo stp global enable**

如果用户在系统视图下启用了生成树，那么所有端口都默认参与生成树计算。如果用户可以确定某些端口连接的部分不存在回路，则可以通过一条在端口视图下的命令关闭特定端口上的生成树功能：

**[Switch -Ethernet0/1] undo stp enable**

MSTP 和 RSTP 能够互相识别对方的协议报文，可以互相兼容。而 STP 无法识别 MSTP 的报文，MSTP 为了实现和 STP 设备的混合组网，同时完全兼容 RSTP，设定了四种工作模式：STP 兼容模式、RSTP 模式、PVST 模式、MSTP 模式。交换机默认工作在 MSTP 模式下，可以通过以下命令在系统视图下设置工作模式：

**[Switch] stp mode { stp | rstp | pvst | mstp }**

## STP 可选配置

- 配置当前设备的优先级

```
[Switch] stp [ instance instance-id ] priority priority
```

- 配置端口为边缘端口

```
[Switch-Ethernet1/0/1] stp edged-port
```

[www.h3c.com](http://www.h3c.com)

默认情况下，所有交换机的优先级是相同的。此时，STP 只能根据 MAC 地址选择根桥，MAC 地址最小的桥为根桥。但实际上，这个 MAC 地址最小的桥并不一定就是最佳的根桥。

所以我们可以配置网桥的优先级来指定根桥。优先级越小，该网桥就越有可能成为根。配置命令为：

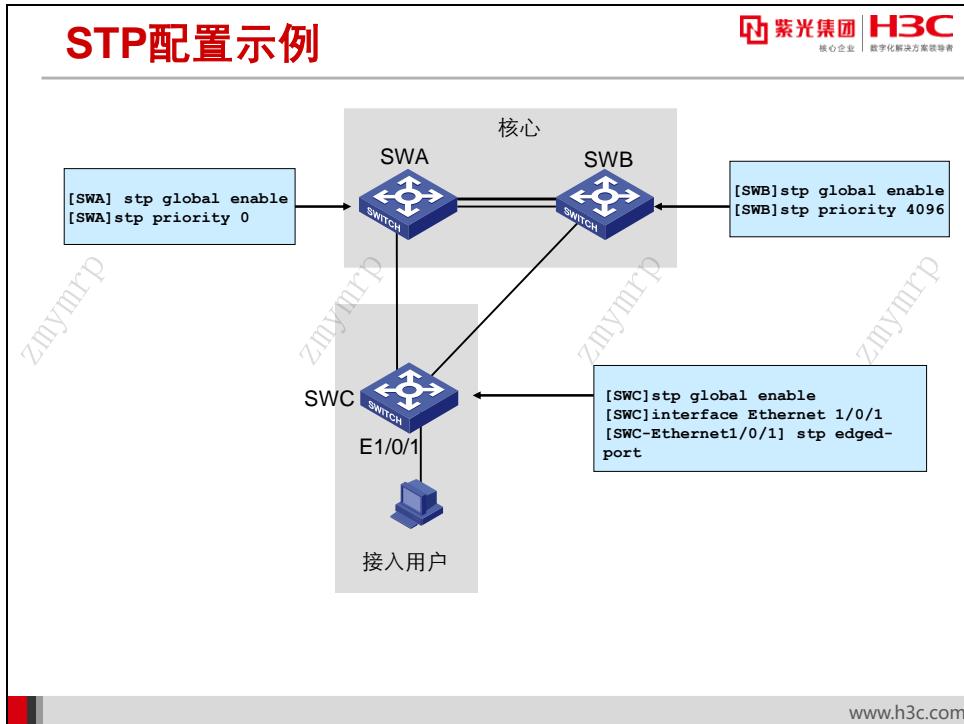
```
[Switch] stp [ instance instance-id ] priority priority
```

在 MSTP 多实例情况下，用 **instance instance-id** 参数来指定交换机在每个实例中的优先级。

当端口直接与用户终端相连，而没有连接到其它设备或共享网段上，则该端口被认为是边缘端口。网络拓扑变化时，边缘端口不会产生临时环路。用户如果将某个端口指定为边缘端口，那么当该端口由堵塞状态向转发状态迁移时，这个端口可以实现快速迁移，而无需等待延迟时间。因此，如果管理员确定某端口是直接与终端相连，可以配置其为边缘端口，可以极大的加快 STP 收敛速度。

在端口视图下配置某端口为边缘端口：

```
[Switch -Ethernet0/1] stp edged-port
```



www.h3c.com

上图为一个启用 STP 防止环路及实现链路冗余的组网。交换机 SWA 和 SWB 是核心交换机，之间通过两条并行链路互连备份；SWC 是接入交换机，接入用户连接到 SWC 的 E1/0/1 端口上。很显然，为了提高网络的性能，应该使交换机 SWA 位于转发路径的中心位置（即生成树的根），同时为了增加可靠性，应该使 SWB 作为根桥的备份。

我们可以通过下面配置使网络能够满足我们的设计需求。

**第1步：**在所有的交换机上启动生成树协议，命令如下：

```
[SWA] stp global enable
[SWB] stp global enable
[SWC] stp global enable
```

**第2步：**配置 SWA 的优先级为 0（默认值为 32768），使其作为整个桥接网络的根桥；配置 SWB 的优先级为 4096，使其作为根桥的备份。命令如下：

```
[SWA] stp priority 0
[SWB] stp priority 4096
```

**第3步：**设置 SWC 的端口 E1/0/1 为边缘端口，以使其在网络拓扑变化时，能够无时延地从阻塞状态迁移到转发状态。命令如下：

```
[SWC-Ethernet0/1] stp edged-port
```

**STP 监控与维护**

紫光集团 | H3C  
核心企业 | 数字化解决方案领导者

```
[SWA]display stp
-----[CIST Global Info] [Mode MSTP]-----
CIST Bridge      :32768.000f-e23e-f9b0 ← 当前桥ID
Bridge Times     :Hello 2s MaxAge 20s FwDly 15s MaxHop 20 ← 当前工作模式
CIST Root/ERPC   :32768.000f-e23e-f9b0 / 0
CIST RegRoot/IRPC :32768.000f-e23e-f9b0 / 0
CIST RootPortId   :0.0
BPDU-Protection    :disabled
Bridge Config-
Digest-Snooping    :disabled
TC or TCN received  :0
....
```

MSTID	Port	Role	STP State	Protection
0	Ethernet1/0/1	DESI	FORWARDING	NONE
0	Ethernet1/0/2	DESI	FORWARDING	NONE
....				

↑ 实例ID                   ↑ 端口角色                   ↑ 端口状态

www.h3c.com

默认情况下，交换机未开启 STP 协议。此时如果执行命令查看 STP 全局状态，则有如下输出：

```
<SWA>display stp
Protocol Status   :disabled
Protocol Std.      :IEEE 802.1s
.....
```

开启 STP 以后，再执行命令查看 STP 全局状态，则有如下输出：

```
[SWA]display stp
-----[CIST Global Info] [Mode MSTP]-----
CIST Bridge      :32768.000f-e23e-f9b0
Bridge Times     :Hello 2s MaxAge 20s FwDly 15s MaxHop 20
CIST Root/ERPC   :32768.000f-e23e-f9b0 / 0
CIST RegRoot/IRPC :32768.000f-e23e-f9b0 / 0
CIST RootPortId   :0.0
BPDU-Protection    :disabled
Bridge Config-
Digest-Snooping    :disabled
TC or TCN received  :0
Time since last TC :8 days 21h:14m:50s
....
```

从以上信息可知，目前交换机运行在 MSTP 模式下。MSTP 协议所生成的树称之为 CIST (Common and Internal Spanning Tree, 公共和内部生成树)，所以显示信息中的 CIST Bridge: 32768.000f-e23e-f9b0 就表示交换机的桥 ID 是 32768.000f-e23e-f9b0；交换机的根桥 ID(CIST Root) 也是 32768.000f-e23e-f9b0。桥 ID 和根桥 ID 相同，说明交换机认为自己就是根桥。

如果想查看生成树中各端口的角色和状态，则用如下命令：

```
[SWA]display stp brief
MSTID  Port          Role  STP State  Protection
```

0	Ethernet1/0/1	DESI	FORWARDING	NONE
0	Ethernet1/0/2	DESI	FORWARDING	NONE
.....				

在 MSTP 协议中可配置多个实例进行负载分担。上面的 **MSTID** 就表示实例的 ID。默认情况下，交换机仅有一个实例，ID 值是 0；且所有 **VLAN** 都绑定到实例 0，所有端口角色和状态都在实例 0 中计算。上面 **Ethernet1/0/1** 和 **Ethernet1/0/2** 端口角色都是指定端口（**DESI**），所以都处于转发状态（**FORWARDING**）。

## 13.10 本章总结

### 本章总结

- STP产生的原因是为了消除路径回环的影响
- STP通过选举根桥和阻塞冗余端口来消除环路
- RSTP、PVST和MSTP工作原理
- 生成树协议配置

## 第14章 交换机端口安全技术

随着以太网应用的日益普及，以太网安全成为日益迫切的需求。在没有安全技术应用的以太网中，用户只要能连接到交换机的物理端口，就可以访问网络中的所有资源，局域网的安全无法得到保证。以太网交换机针对网络安全问题提供了多种安全机制，包括地址绑定、端口隔离、接入认证等技术。本文将对这些以太网安全技术的原理与技术进行讲解。

## 14.1 本章目标

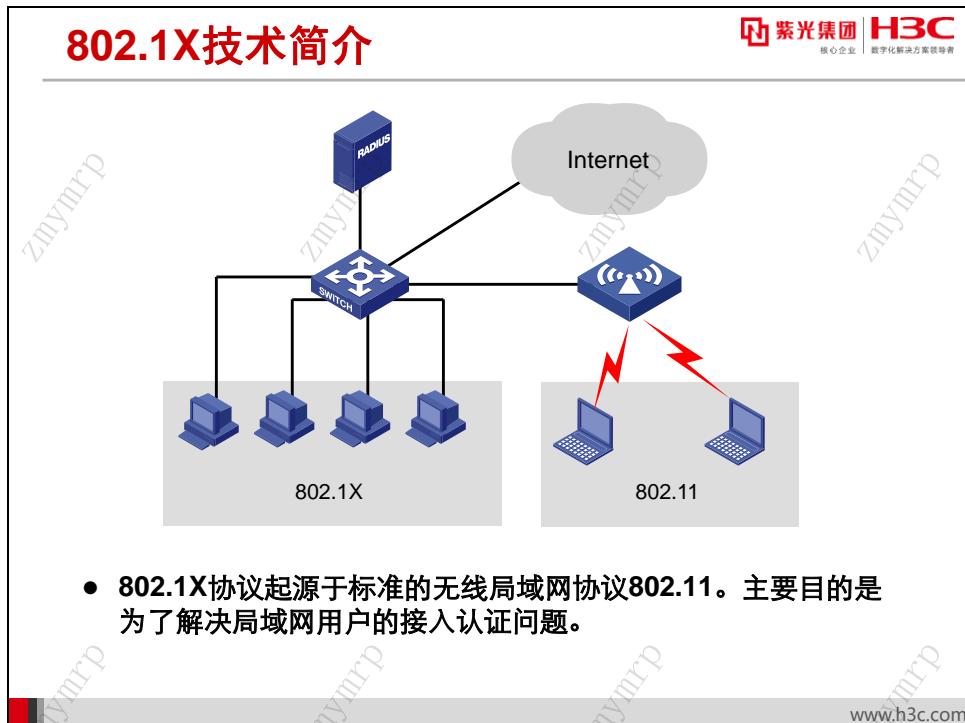
### 课程目标

学习完本课程，您应该能够：

- 掌握802.1X基本原理及其配置
- 掌握端口隔离技术及其配置



## 14.2 802.1X的基本原理和配置

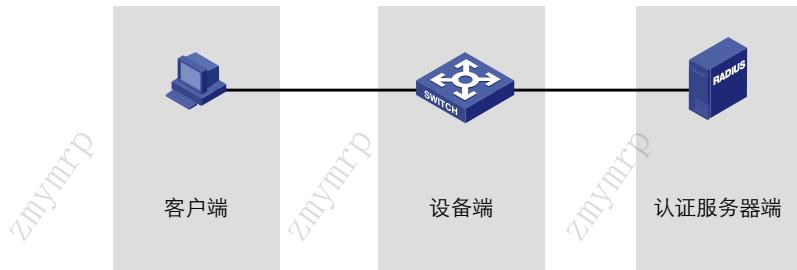


IEEE 802.1X 标准（以下简称 802.1X）是一种基于端口的网络接入控制（Port Based Network Access Control）协议，IEEE 于 2001 年颁布该标准文本并建议业界厂商使用其中的协议作为局域网用户接入认证的标准协议。802.1X 的提出起源于 IEEE 802.11 标准——无线局域网用户接入协议标准，其最初目的主要是解决无线局域网用户的接入认证问题；但由于它的原理对于所有符合 IEEE 802 标准的局域网具有普适性，因此后来它在有线局域网中也得到了广泛的应用。

在符合 IEEE 802 标准的局域网中，只要与局域网接入控制设备（如交换机）相接，用户就可以与局域网连接并访问其中的设备和资源。但是对于诸如电信接入、商务局域网（典型的例子是写字楼中的 LAN）以及移动办公等应用场合，局域网服务的提供者普遍希望能对用户的接入进行控制，为此产生了对“基于端口的网络接入控制”的需求。顾名思义，“基于端口的网络接入控制”是指在局域网接入控制设备的端口这一级对所接入的设备进行认证和控制。连接在端口上的用户设备如果能通过认证，就可以访问局域网中的资源；如果不能通过认证，则无法访问局域网中的资源，相当于物理连接被断开。

## 802.1X的体系结构

紫光集团 | **H3C**  
核心企业 | 数字化解决方案领导者



- **本地认证**

→ 由设备端内置本地服务器对客户端进行认证

- **远程集中认证**

→ 由远程的认证服务器对客户端进行认证

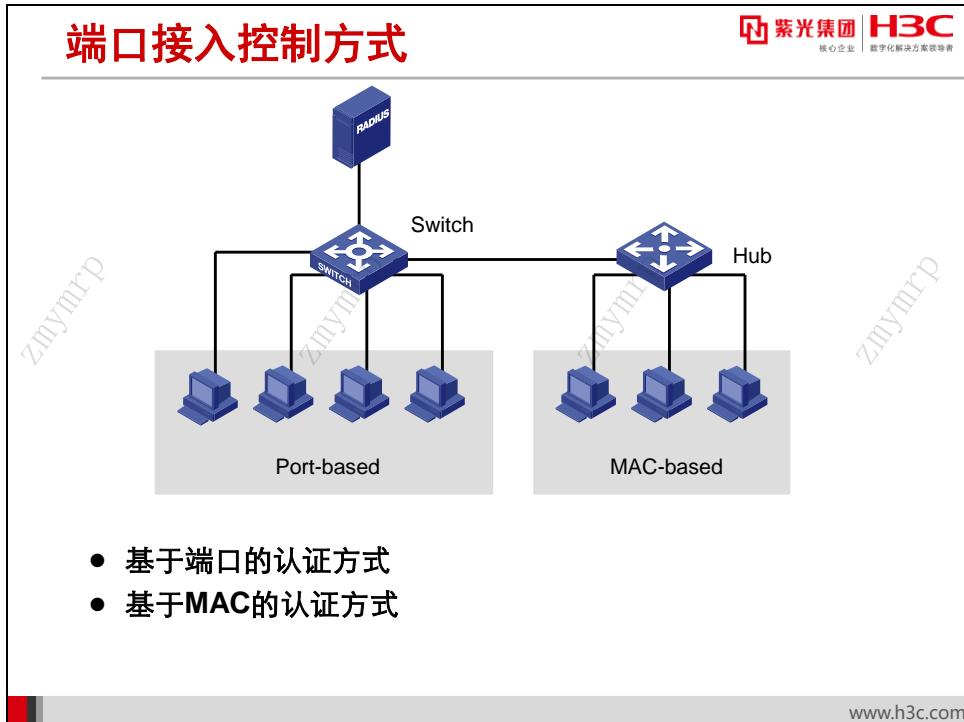
[www.h3c.com](http://www.h3c.com)

使用 802.1X 的系统为典型的客户端/服务器体系结构, 包括三个实体, 如上图所示分别为: 客户端 (Supplicant System)、设备端 (Authenticator System) 以及认证服务器 (Authentication Server System)。

- **客户端:** 安装有 802.1X 客户端软件的用户终端设备。用户通过客户端软件发起 802.1X 认证。客户端必须支持 EAPOL (Extensible Authentication Protocol over LANs, 局域网上的可扩展认证协议)。
- **设备端:** 通常为支持 802.1X 协议的网络设备, 它为客户端提供接入局域网的端口, 如支持 802.1X 协议的交换机。认证通过后, 设备端开放端口, 客户端能够访问网络。
- **认证服务器:** 能够对用户实现进行认证、授权和计费功能。

认证服务器可分为本地认证服务器和远程集中认证服务器, 分别适用于不同场合:

- 本地认证服务器通常集成在设备端上。设备端内置的认证服务器对客户端进行认证, 认证通过后开放端口。本地认证方式适用于网络规模小, 客户端数量不多的情况下; 但用户信息数据库分散在设备本地上, 维护管理不便。
- 远程集中认证服务器通常是一台专门的认证服务器。设备端把客户端信息发送到远程认证服务器, 由服务器查找用户信息数据库后返回消息给设备端。在远程集中认证方式下, 用户信息数据库能够集中管理, 维护管理方便, 适用于较大规模网络中。



交换机端口对用户的接入控制方式包括基于 MAC 地址的认证方式和基于端口的认证方式。

- 基于端口的认证方式

采用基于端口方式时，只要该端口下的第一个用户认证成功后，其它接入用户无须认证就可使用网络资源，但是当第一个用户下线后，其它用户也会被拒绝使用网络。

- 基于 MAC 地址的认证方式

当采用基于 MAC 方式时，该端口下的所有接入用户均需要单独认证，当某个用户下线时，也只有该用户无法使用网络。

默认情况下，802.1X 在端口上进行接入控制方式为基于 MAC 地址的认证方式。

## 802.1X基本配置

- 开启全局的802.1X特性

**[Switch] dot1x**

- 开启端口的802.1X特性

**[Switch-Ten-GigabitEthernet1/0/1] dot1x**

- 添加本地接入用户并设置相关参数

```
[Switch] local-user user-name class network
[Switch- luser-network-localuser] service-type lan-
access
[Switch- luser-network-localuser] password {
cipher | simple } password
```

www.h3c.com

在交换机上配置 802.1X 的基本步骤如下：

**第1步：** 在系统视图下开启全局的 802.1X 特性。其配置命令如下：

**dot1x**

**第2步：** 在接口视图下开启端口的 802.1X 特性。其配置命令如下：

**dot1x**

**第3步：** 添加本地接入用户并设置相关参数。其配置命令如下：

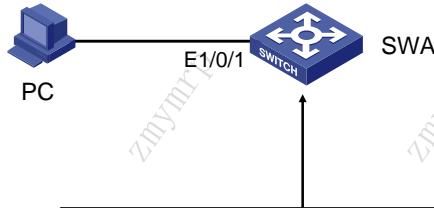
```
local-user user-name class network
service-type lan-access
password { cipher | simple } password
```

**注意：**

必须同时开启全局和端口的 802.1X 特性后，802.1X 的配置才能在端口上生效。

## 802.1X典型配置举例

紫光集团 | H3C  
核心企业 | 数字化解决方案领导者



```

[SWA]dot1x
[SWA]interface GigabitEthernet 1/0/1
[SWA-GigabitEthernet1/0/1]dot1x
[SWA]local-user localuser class network
[SWA-luser-network-localuser]password simple hello
[SWA-luser-network-localuser]service-type lan-access
  
```

[www.h3c.com](http://www.h3c.com)

在上图中，PC 连接到交换机的端口 E1/0/1 上。交换机启用 802.1X 来对 PC 接入进行认证，认证方式为本地认证。

配置 SWA：

```

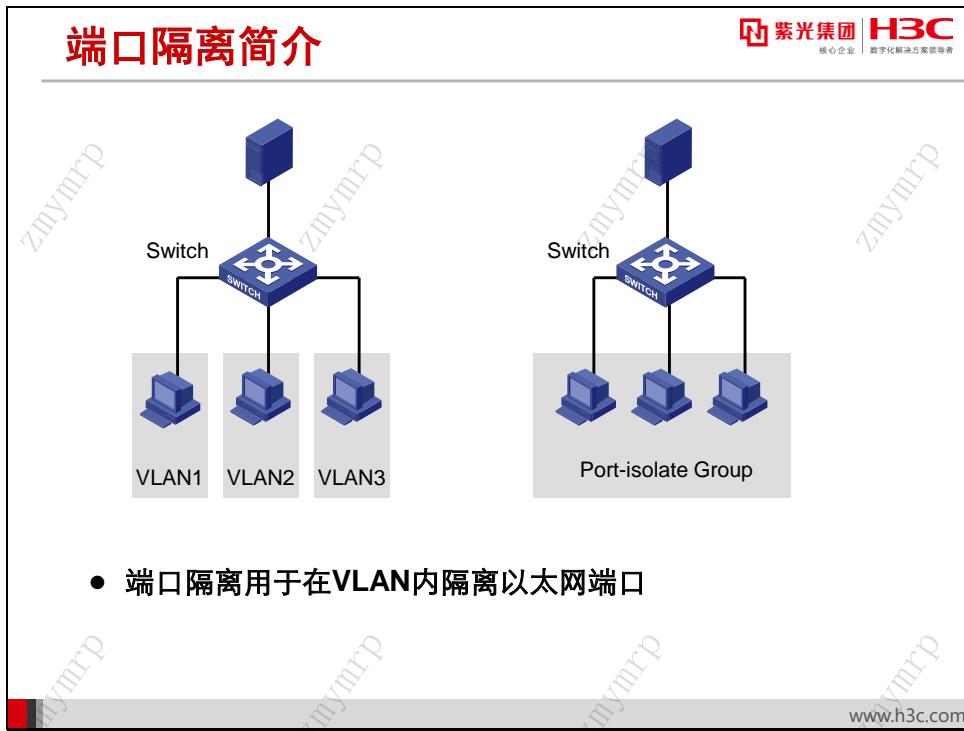
[SWA]dot1x
[SWA]interface GigabitEthernet 1/0/1
[SWA-GigabitEthernet1/0/1]dot1x
[SWA]local-user localuser class network
[SWA-luser-network-localuser]password simple hello
[SWA-luser-network-localuser]service-type lan-access
  
```

配置完成后，在 PC 上打开 802.1X 认证客户端软件，按照提示输入用户名 localuser 和密码 hello 后，PC 就能够接入网络了。

**注意：**

Windows XP 系统自带客户端，无须安装。连接到启用 802.1X 的交换机端口后，系统会自动弹出对话框，要求输入用户名、密码。

### 14.3 端口隔离技术及其配置



为了实现报文之间的二层隔离，可以将不同的端口加入不同的 VLAN，但会浪费有限的 VLAN 的资源。采用端口隔离特性，可以实现同一 VLAN 内端口之间的隔离。用户只需要将端口加入到隔离组中，就可以实现隔离组内端口之间二层数据的隔离。端口隔离功能为用户提供了更安全、更灵活的组网方案。

用户可以将需要进行控制的端口加入到一个隔离组中，实现隔离组内端口之间二层数据的隔离。

## 端口隔离基本配置



### ● 创建隔离组

```
[switch]port-isolate group 1
```

- 将指定端口加入到隔离组中，端口成为隔离组的普通端口

```
[Switch-Ethernet1/0/1] port-isolate enable group 1
```

www.h3c.com

创建端口隔离组在系统试图下完场。配置创建端口隔离组的命令为：

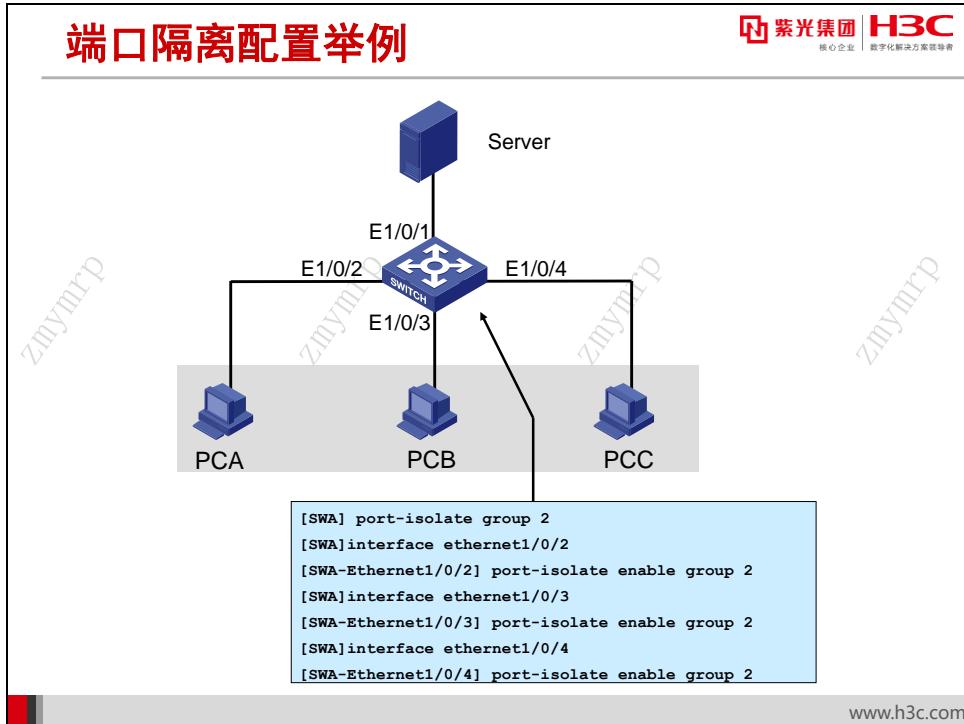
**port-isolate group group-number**

端口隔离的配置在以太网端口视图下完成。配置以太网端口加入隔离组并成为隔离组中普通端口的命令为：

**port-isolate enable group group-number**

**注意：**

系统自动创建了隔离组且其组号为 1。可以创建 8 个隔离组。



图中网络中，PCA、PCB、PCC 分别与交换机的端口 Ethernet1/0/2、Ethernet1/0/3、Ethernet1/0/4 相连，服务器与端口 Ethernet1/0/1 相连。在交换机配置端口隔离。

配置交换机：

```

[SWA] port-isolate group 2
[SWA] interface ethernet1/0/2
[SWA-Ethernet1/0/2] port-isolate enable group 2
[SWA] interface ethernet1/0/3
[SWA-Ethernet1/0/3] port-isolate enable group 2
[SWA] interface ethernet1/0/4
[SWA-Ethernet1/0/4] port-isolate enable group 2

```

配置完成后，网络中 PCA、PCB、PCC 之间被隔离，不能互相访问；但所有的 PC 都能够访问服务器。

## 14.4 本章总结

### 本章总结

- 802.1X是基于端口的网络接入控制协议，对接入用户进行验证；
- 交换机端口隔离技术能够在VLAN内隔离端口

# 第15章 配置链路聚合

在组建局域网的过程中，连通性是最基本的要求。在保证连通性的基础上，有时还要求网络具有高带宽、高可靠性等。链路聚合技术是在局域网中最常见的高带宽和高可靠性技术。

本章介绍了链路聚合的作用，链路聚合中负载分担的原理，以及如何在交换机上配置及维护链路聚合。

## 15.1 本章目标

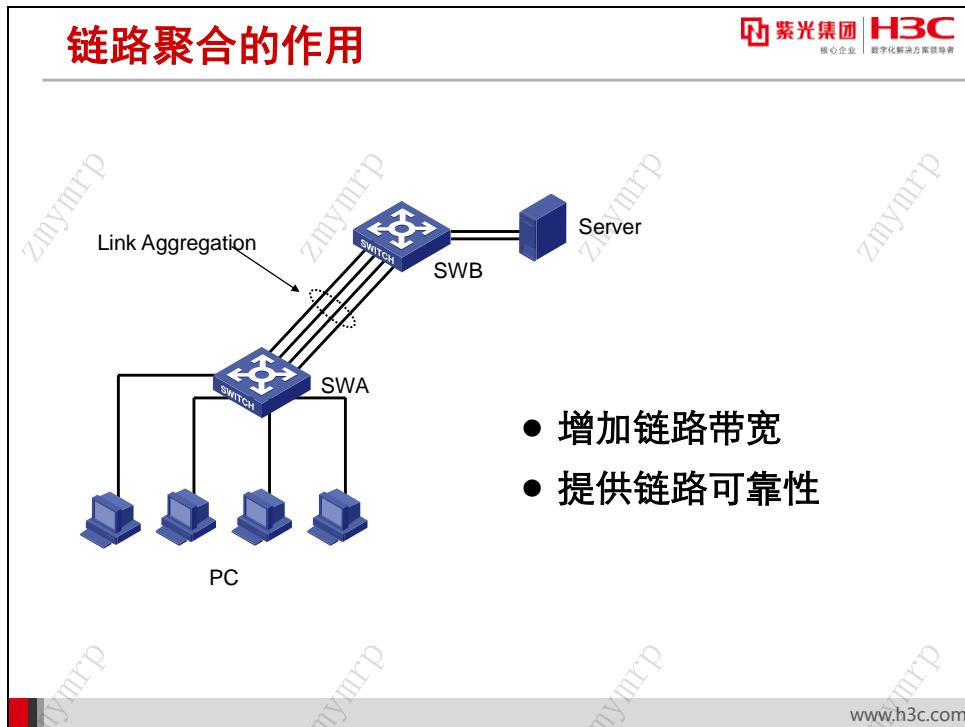
### 课程目标

● 学习完本课程，您应该能够：

- 了解链路聚合的作用
- 掌握链路聚合的分类
- 掌握链路聚合的基本配置



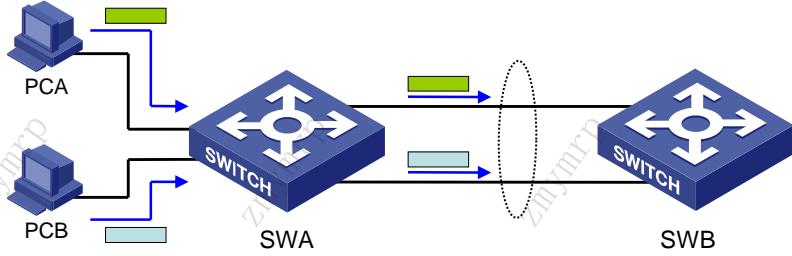
## 15.2 链路聚合简介



链路聚合是以太网交换机所实现的一种非常重要的高可靠性技术。通过链路聚合，多个物理以太网链路聚合在一起形成一个逻辑上的聚合端口组。使用链路聚合服务的上层实体把同一聚合组内的多条物理链路视为一条逻辑链路，数据通过聚合端口组进行传输。链路聚合具有以下优点。

- **增加链路带宽：**通过把数据流分散在聚合组中各个成员端口，实现端口间的流量负载分担，从而有效地增加了交换机间的链路带宽。
- **提供链路可靠性：**聚合组可以实时监控同一聚合组内各个成员端口的状态，从而实现成员端口之间彼此动态备份。如果某个端口故障，聚合组及时把数据流从其他端口传输。

## 聚合链路负载分担原理



- 聚合后链路基于流进行负载分担

[www.h3c.com](http://www.h3c.com)

链路聚合后，上层实体把同一聚合组内的多条物理链路视为一条逻辑链路，系统根据一定的算法，把不同的数据流分布到各成员端口上，从而实现基于流的负载分担。

上图中，因为 PCA 和 PCB 的 MAC 地址不同，系统认为是两条流，所以 SWA 把这两条流分别从聚合组中的两个成员端口向外发送。同理，返回的数据流在 SWB 上也会被分布到两条链路上传输。

## 15.3 链路聚合的分类

**链路聚合分类**

紫光集团 | H3C  
核心企业 | 数字化解决方案领导者

- **静态聚合**
  - 双方系统间不使用聚合协议来协商链路信息
- **动态聚合**
  - 双方系统间使用聚合协议来协商链路信息
  - LACP (Link Aggregation Control Protocol, 链路聚合控制协议) 是一种基于 IEEE802.3ad 标准的、能够实现链路动态聚合的协议

www.h3c.com

按照聚合方式的不同，链路聚合可以分为二大类：

- 静态聚合

在静态聚合方式下，双方设备不需要启用聚合协议，双方不进行聚合组中成员端口状态的交互。

如果一方设备不支持聚合协议或双方设备所支持的聚合协议不兼容，则可以使用静态聚合方式来实现聚合。

- 动态聚合

在动态聚合方式下，双方系统使用 LACP 协议来协商链路信息，交互聚合组中成员端口状态。

LACP (Link Aggregation Control Protocol, 链路聚合控制协议) 是一种基于 IEEE802.3ad 标准的、能够实现链路动态聚合与解聚合的协议。LACP 协议通过 LACPDU (Link Aggregation Control Protocol Data Unit, 链路聚合控制协议数据单元) 与对端交互信息。

## 15.4 链路聚合的基本配置

**静态聚合配置**

**● 创建聚合端口**

**[Switch] interface bridge-aggregation *interface-number***

**● 将以太网端口加入聚合组**

**[Switch-Ethernet1/0/1] port link-aggregation group *number***

www.h3c.com

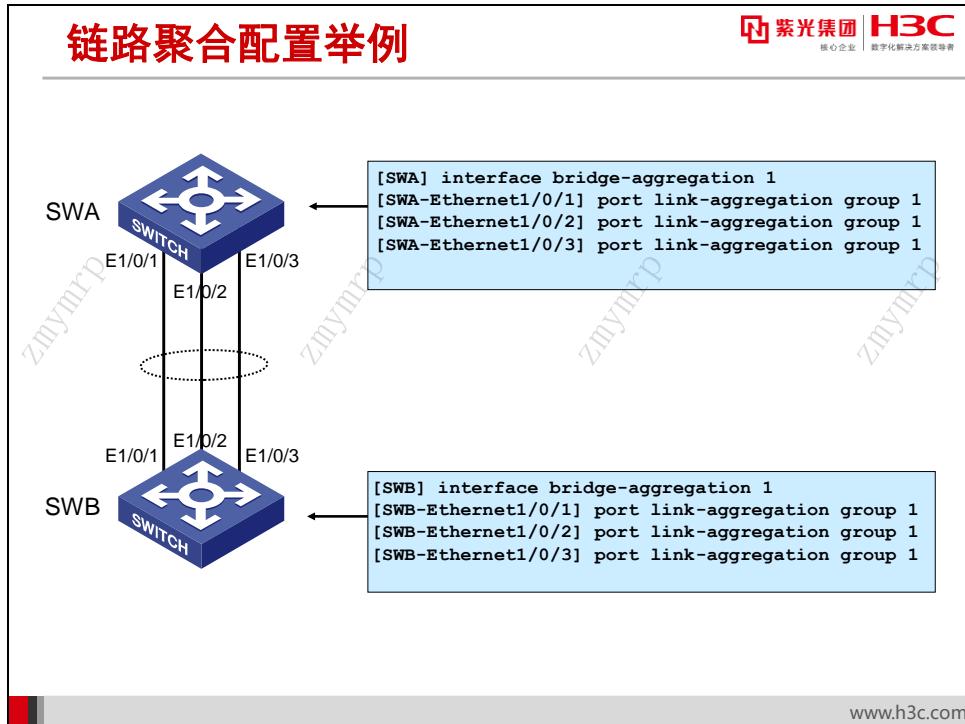
静态聚合的优点是没有聚合协议报文占用带宽，对双方的聚合协议没有兼容性要求。在小型局域网中，最常用的链路聚合方式是静态聚合。配置静态聚合的步骤如下。

**第1步：**在系统视图下创建聚合端口。配置命令为：

```
interface bridge-aggregation interface-number
```

**第2步：**在接口视图下把物理端口加入到创建的聚合组中。配置命令为：

```
port link-aggregation group number
```



上图中,交换机 SWA 使用端口 E1/0/1、E1/0/2 和 E1/0/3 连接到 SWB 的端口 E1/0/1、E1/0/2 和 E1/0/3。在交换机上启用链路聚合以实现增加带宽和可靠性的需求。

#### 配置 SWA:

```

[SWA] interface bridge-aggregation 1
[SWA] interface Ethernet 1/0/1
[SWA-Ethernet1/0/1] port link-aggregation group 1
[SWA] interface Ethernet 1/0/2
[SWA-Ethernet1/0/2] port link-aggregation group 1
[SWA] interface Ethernet 1/0/3
[SWA-Ethernet1/0/3] port link-aggregation group 1

```

#### 配置 SWB:

```

[SWB] interface bridge-aggregation 1
[SWB] interface Ethernet 1/0/1
[SWB-Ethernet1/0/1] port link-aggregation group 1
[SWB] interface Ethernet 1/0/2
[SWB-Ethernet1/0/2] port link-aggregation group 1
[SWB] interface Ethernet 1/0/3
[SWB-Ethernet1/0/3] port link-aggregation group 1

```

紫光集团 | H3C  
核心企业 | 数字化解决方案提供者

```
<Switch>display link-aggregation summary
```

Aggregation Interface Type:  
BAGG -- Bridge-Aggregation, RAGG -- Route-Aggregation  
Aggregation Mode: S -- Static, D -- Dynamic  
Loadsharing Type: Shar -- Loadsharing, NonS -- Non-Loadsharing  
Actor System ID: 0x8000, 000f-e267-6c6a

AGG Interface	AGG Mode	Partner ID	Select Ports	Unselect Ports	Share Type
BAGG1	S	none	3	0	Shar

↑ 聚合端口ID为1      ↑ 聚合方式为静态聚合      ↑ 聚合组中包含有3个端口      ↑ 组中端口是负载分担类型

www.h3c.com

在任意视图下可以用 `display link-aggregation summary` 查看链路聚合的状态。如下所示：

```
<Switch>display link-aggregation summary
```

Aggregation Interface Type:  
BAGG -- Bridge-Aggregation, RAGG -- Route-Aggregation  
Aggregation Mode: S -- Static, D -- Dynamic  
Loadsharing Type: Shar -- Loadsharing, NonS -- Non-Loadsharing  
Actor System ID: 0x8000, 000f-e267-6c6a

AGG Interface	AGG Mode	Partner ID	Select Ports	Unselect Ports	Share Type
BAGG1	S	none	3	0	Shar

以上输出信息表示，这个聚合端口的 ID 是 1，聚合方式为静态聚合，组中包含了 3 个 Selected 端口，处于激活状态并工作在负载分担模式下。

#### 注意：

处于 Selected 状态的端口可以参与转发数据流。Unselected 状态表示端口目前未被选中，不参与数据流转发。比如，端口在物理层 down 的情况下就是 Unselect Ports。

## 15.5 本章总结

### 本章总结

- 链路聚合可以实现链路备份、增加链路带宽及其数据的负载
- 链路聚合按照聚合方式不同分为静态聚合和动态聚合