第 3 章

交换机工作原理

第 1 章详细介绍了 TCP/IP，按由下往上的顺序协议栈 5 层分别是：物理层、链路层、网络层、传输层和应用层。物理层基本上是物理网卡，在实际工作中需要配置的地方不多。

需要配置的网络设备绝大部分位于链路层和网络层，因此下面将重点介绍链路层、网络层相关的协议。

本章介绍链路层相关的内容，包括交换机工作原理、VLAN 原理、STP 原理、RSTP&MSTP 原理等 4 个部分。

3.1 交换机工作原理

工作于链路层的网络设备有 Hub、网桥、交换机，其中 Hub、网桥目前已基本不用，被交换机所替代。出于学习的目的，这里比较一下 Hub 和交换机的差异。Hub 简单地将各个端口连在一起，如图 3.1 所示。

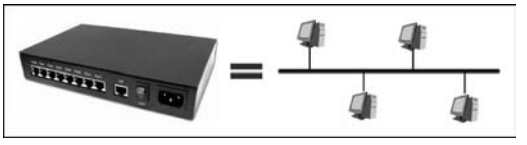


图 3.1 Hub 设备工作原理

连在 Hub 上的设备共享冲突域，同一时间只能有一台设备发送数据，带宽受到很大限制。

与 Hub 相比，交换机就聪明多了，不同设备互相隔离，避免冲突。如图 3.2 所示，连在交换机 SWA 上的 4 台主机可以同时两两互相通信，主机 A 和主机 B 通信，主机 C 和主机 D 通信，互不影响。

那么，当主机 A 的数据发到 SWA 的时候，SWA 怎么知道这个数据应该发给主机 B，而不是主机 C 或者 D 呢？

如图 3.3 所示，SWA 上有一张数据转发表，里面有 MAC 地址和接口的对应关系，主机

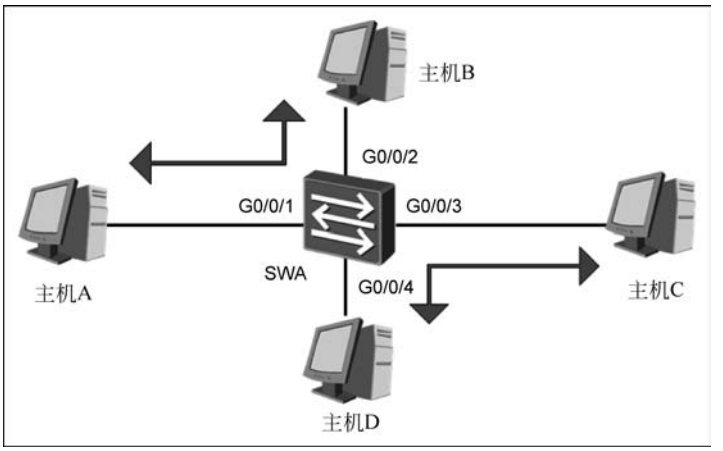


图 3.2　交换机工作原理

A 给 SWA 发以太网帧的时候，目标 MAC 地址填的是主机 B 的 MAC 地址，SWA 根据收到的帧里面的目标 MAC 地址查表，得知这个以太网帧应该从 G0/0/2 转发出去，这样就可以正确送达主机 B 了。

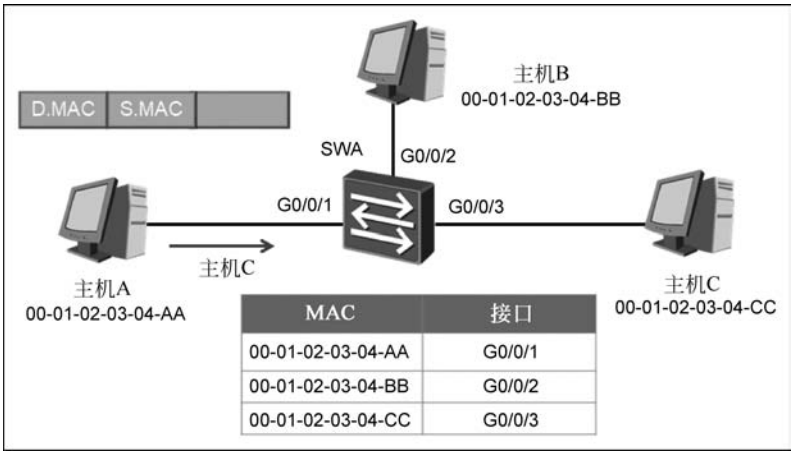


图 3.3　交换机数据转发机制

那么这个 MAC 地址表是怎么来的？交换机刚上电的时候这个表是空的，如图 3.4 所示。

主机 A、主机 B、主机 C 在互相通信前，首先要做的就是获得对方的 MAC 地址。最开始的时候需要通过 ARP 协议获得 MAC 地址，例如主机 A 要获得主机 C 的 MAC 地址时，要先发送 ARP 请求，然后等待主机 C 回应该 ARP 请求。

交换机通过分析 ARP 报文来更新 MAC 地址表。注意 MAC 地址表的更新过程：首先，

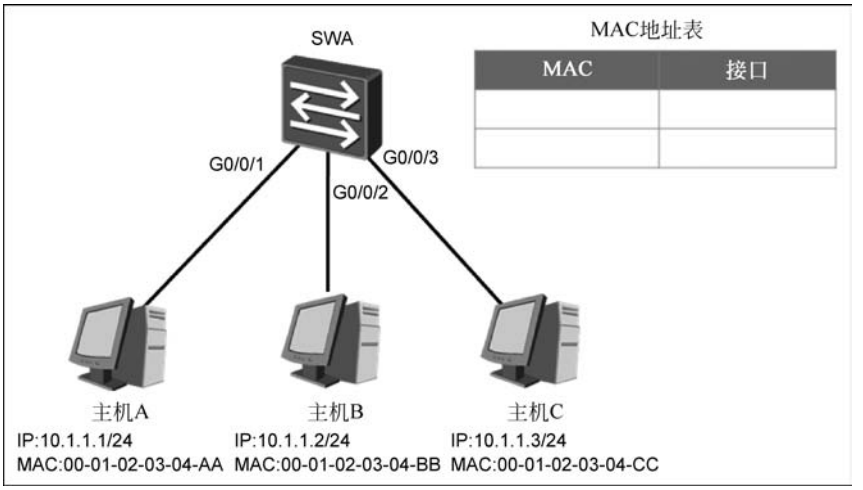
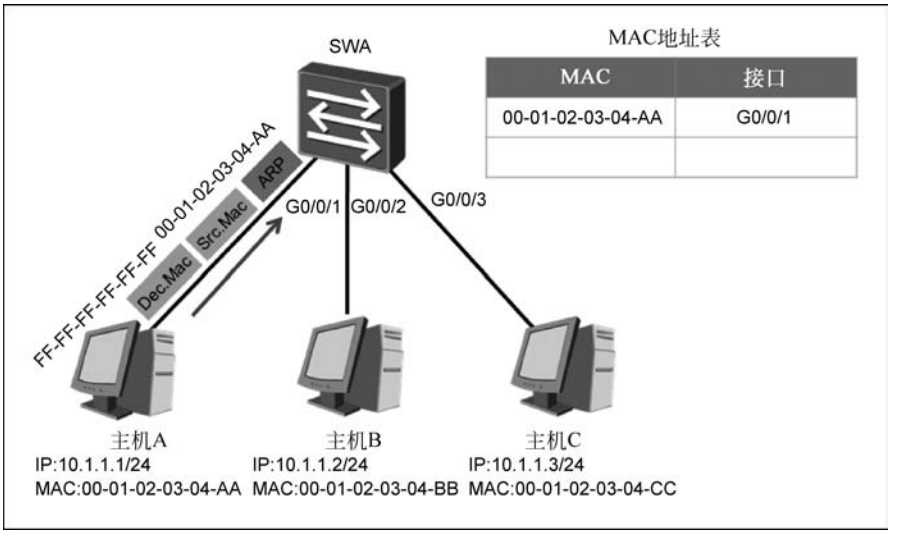


图 3.4　初始状态的 MAC 地址表

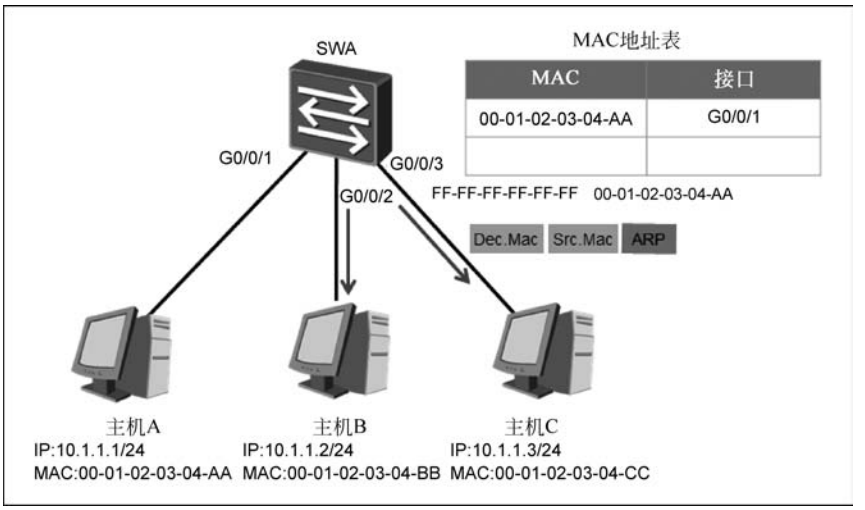
交换机根据来自主机 A 的 ARP 请求在 MAC 地址表中添加主机 A 的信息，如图 3.5(a)所示；然后，交换机将该 ARP 请求转发给主机 B 和主机 C，如图 3.5(b)所示；最后，交换机根据来自主机 C 的 ARP 回应在 MAC 地址表中添加主机 C 的信息，如图 3.5(c)所示。

交换机获取 MAC 地址与接口的映射信息之后，就可以指导报文转发。如果主机和交换机的连接断开，例如图 3.5(c)中主机 C 离开 SWA，SWA 检测到链路断开，会马上更新 MAC 地址表，删除主机 C 对应的表项。可以用 display mac-address 查询交换机 MAC 地址表的变化，如图 3.6 所示。

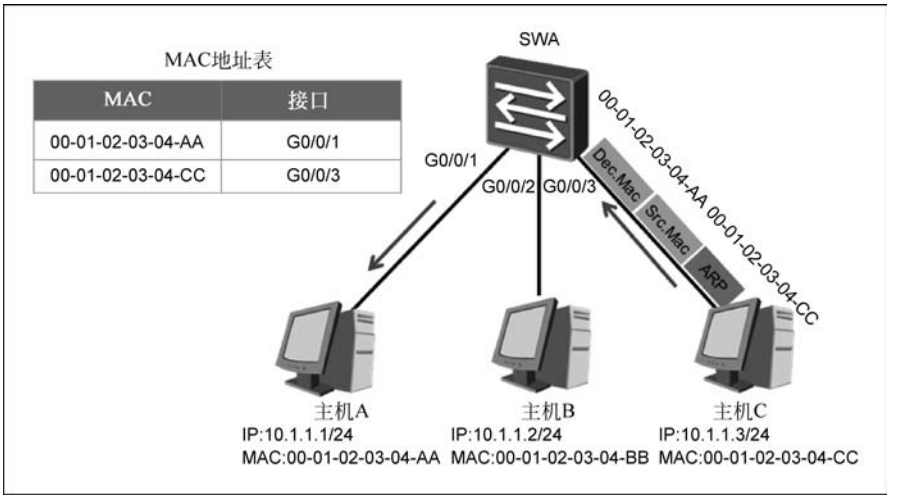


(a) 添加主机 A 信息

图　3.5



（b）转发 ARP 请求



（c）添加主机 C 信息

图 3.5 （续）

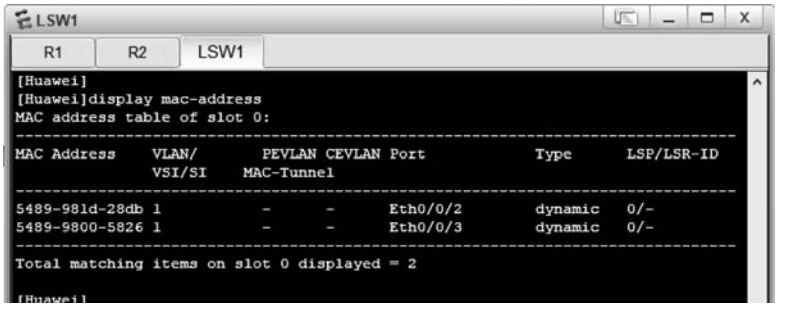


图 3.6　MAC 地址表查询

SWA 已经删除了主机 C 对应的表项，此时主机 A 并不知道主机 C 已经离开，ARP 缓存表里还有主机 C 对应的 MAC 地址，如果主机 A ping 主机 C，SWA 会怎么处理呢？

SWA 收到主机 A 发来的报文，根据目标 MAC 查表时，发现找不到主机 C 对应的表项，这个情况对于 SWA 来说是未知单播。交换机对未知单播的处理方法是将其广播给各个端口。这里可以自己做实验验证一下。

还有一种 MAC 地址是广播 MAC，交换机的处理方法也是发给各个端口，这个过程叫作泛洪。

前面介绍 MAC 地址的时候，还提到一种特殊 MAC——组播 MAC，组播 MAC 有一个专门的组播 MAC 地址表，交换机会根据这个表给对应组播成员精确转发。

另外还有一种特殊情况——丢弃，例如 SWA 接口 Ethernet 0/0/0 配置的 VLAN 是 1，但是收到的帧携带的 VLAN 是 2，这个帧就会被丢弃；或者是帧已经进入交换机，但是不符合转发规则，也会被丢弃。

以上各种转发情况汇总如图 3.7 所示。

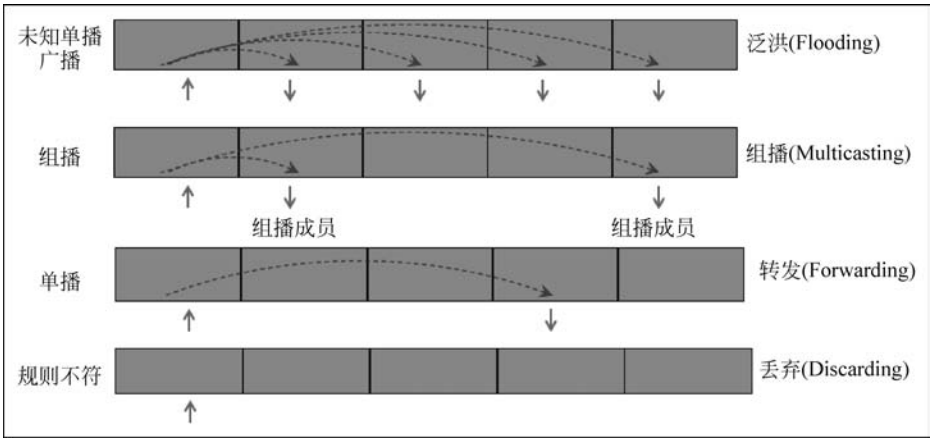


图 3.7　交换机转发规则

3.2　VLAN 原理与配置

3.2.1　VLAN 基本原理

交换机可以让不同的主机互相之间同时通信，但是不能隔离广播。在网络规模比较大的时候，广播报文会耗费很多网络资源，如图 3.8 所示，每一个广播报文都会发给所有的主机。

为了避免全网广播问题，可以用 VLAN(Virtual Local Area Network，虚拟局域网)技

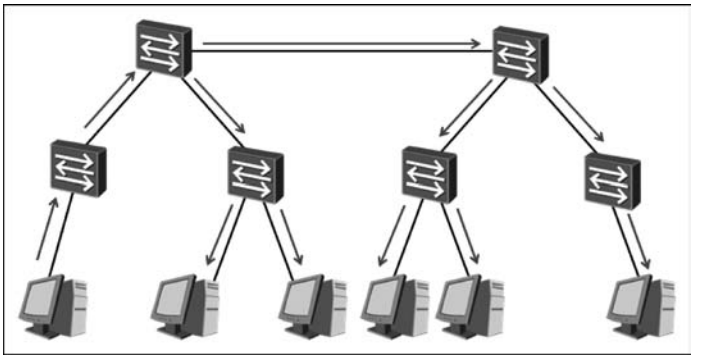


图 3.8　广播报文转发

术将一个物理的局域网在逻辑上划分成多个广播域。这样既能够隔离广播域，又能够提升网络的安全性，如图 3.9 所示。

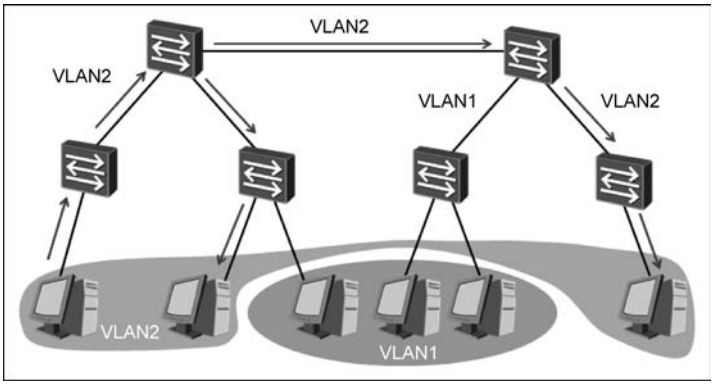


图 3.9　VLAN 基本概念

VLAN 技术是如何实现的呢？或者说交换机如何知道哪个报文属于 VLAN 1，哪个报文属于 VLAN 2 呢？实际上是通过 VLAN 标签识别的。如图 3.10 所示，这是前面介绍过的普通以太网帧。

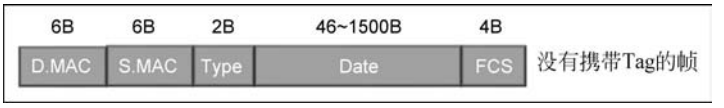


图 3.10　普通以太网帧

携带 VLAN 标签的以太网帧是什么样的呢？如图 3.11 所示，VLAN 标签总共 4B，在 S. MAC 后面，Type 前面。

4B 的 VLAN 标签前 2B 是 TPID(Tag Protocol Identifier，标签协议标识)，华为设备取固定值 0x8100，后面 2B 分 3 个部分，具体如下：

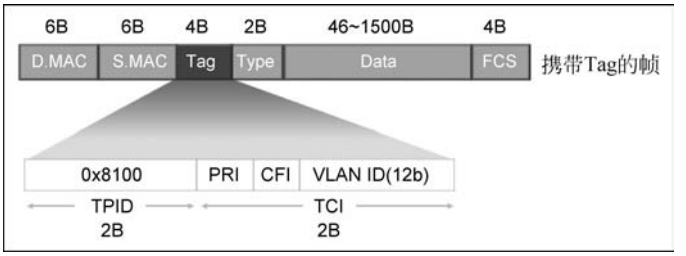


图 3.11　带 VLAN 标签的帧

PRI：Priority，优先级，3 比特，可以表示的范围是 0～7，用来标识帧的优先级，和 IP 头部的优先级类似，不同的是这个优先级用来指导交换机，IP 头的优先级用来指导路由器。

CFI：Canonical Format Indicator，规范格式指示器，1 比特，0 表示以太网，1 表示令牌环网。通常取值 0。

VLAN ID：VLAN Identifier，12 比特，可以表示的范围是 0～4095，其中 0 和 4095 这两个 ID 保留，不能使用，因此实际应用中，可用的 VLAN ID 范围是 1～4094。

这个 VLAN 标签由交换机来添加，如图 3.12 所示。主机发出来的以太网帧不携带 VLAN Tag，通常也称为 Untagged 帧，这个帧到达 SWA 接口后，SWA 会添加一个 VLAN 标签。

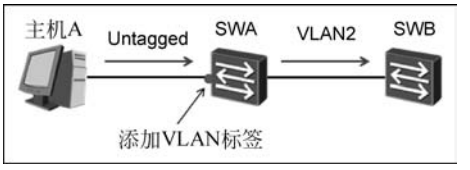
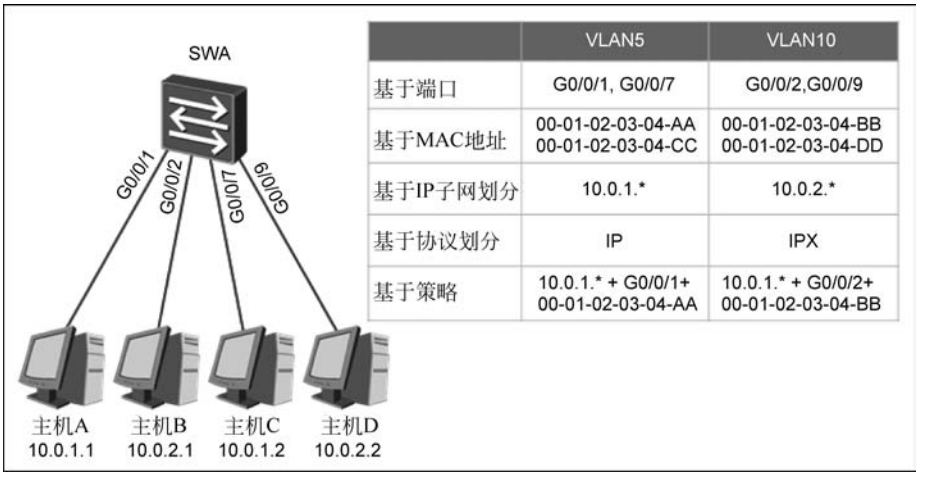


图 3.12　添加 VLAN 标签

交换机根据什么添加 VLAN 标签呢？如图 3.13 所示，有多种不同的方式：



	VLAN5	VLAN10
基于端口	G0/0/1, G0/0/7	G0/0/2,G0/0/9
基于MAC地址	00-01-02-03-04-AA 00-01-02-03-04-CC	00-01-02-03-04-BB 00-01-02-03-04-DD
基于IP子网划分	10.0.1.*	10.0.2.*
基于协议划分	IP	IPX
基于策略	10.0.1.* + G0/0/1+ 00-01-02-03-04-AA	10.0.1.* + G0/0/2+ 00-01-02-03-04-BB

图 3.13　VLAN 标签添加的不同方法

第 1 种：基于端口，图中 SWA 端口 G0/0/1 和 G0/0/7 收到的报文都打上 VLAN5，G0/0/2 和 G0/0/9 收到的报文都打上 VLAN10，这是最常用的方式。

第 2 种：基于 MAC 地址，SWA 根据收到报文的 S.MAC 添加 VLAN 标签，这种方式需要对每个接入 SWA 的 PC 都要配置 MAC 和 VLAN 的映射关系，管理很不方便，实际中很少用。

第 3 种：基于 IP 网段，SWA 收到报文之后需要分析源 IP 所处的网段，因为要额外分析 IP 头，消耗交换机资源，实际中很少用。

第 4 种：基于协议划分，应用面很窄，实际中很少用。

第 5 种：基于策略，也就是将前面的 4 种进行组合控制，可以精准控制，但是配置复杂，消耗资源多，实际中用得更少。

通常都是基于端口分配 VLAN 标签，其他几种方式很少用到。

基于端口分配 VLAN 标签时，用 PVID(Port VLAN ID，端口 VLAN ID)配置接口的 VLAN ID，如图 3.14 所示。

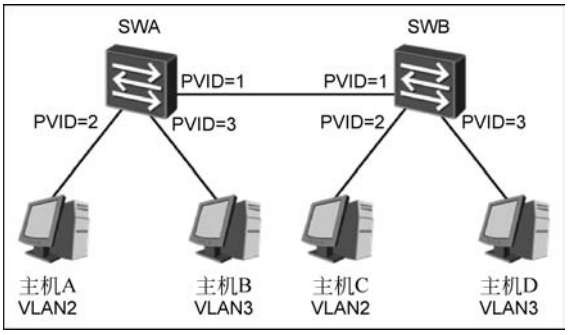


图 3.14　配置接口 VLAN ID

VLAN 标签是在交换机接口上处理的，交换机接收主机发过来的报文时会添加 VLAN 标签，往主机发送报文的时候需要剥离 VLAN 标签，这是因为主机不能识别带 VLAN 标签的报文。然而如果对端是交换机，发送报文的时候，又需要带 VLAN 标签。为了更好地控制 VLAN 标签的处理，交换机的接口可以工作于不同模式。

如图 3.15 所示，交换机连接主机的接口通常用 Access(接入)模式，两交换机之间的接口通常用 Trunk(骨干)模式。

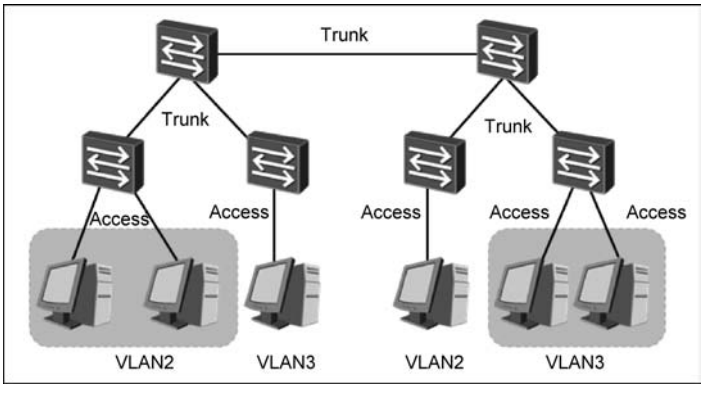


图 3.15　交换机接口类型

3.2.2 Access 口工作原理

Access 口和 Trunk 口的工作机制有什么区别呢？首先看 Access 口的工作机制。如图 3.16 所示，方框表示交换机，端口 PVID=5，主机发往交换机的报文通常是 Untagged，有些情况下也可以发 Tagged 报文。

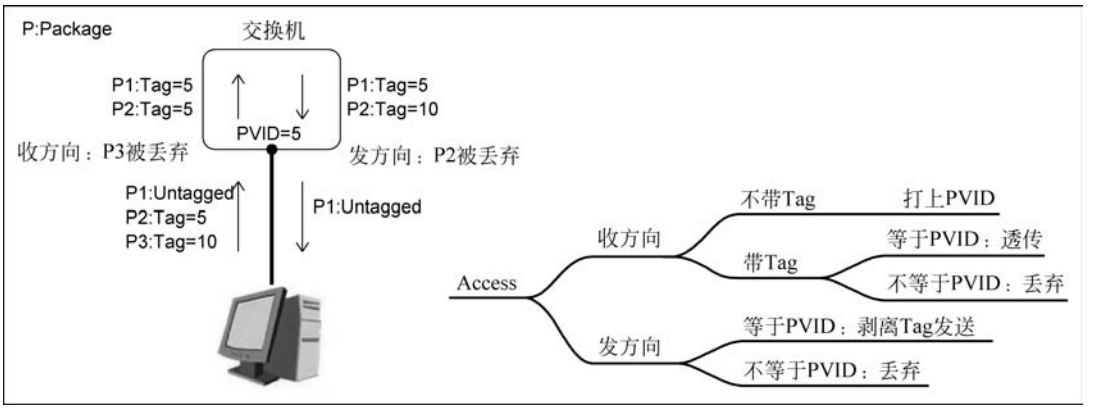


图 3.16 Access 口工作机制

收方向，指交换机接收主机发来的报文，分 3 种情况：

第 1 种：Untagged 报文，交换机会添加一个 VLAN Tag=5，见方框左边 P1 报文处理；

第 2 种：Tag=5 的报文，交换机直接放行，不再添加 VLAN Tag，见方框左边 P2 报文处理；

第 3 种：Tag=10 的报文，因为和 PVID 不一致，直接丢弃，见方框左边 P3 报文处理。

发方向，指交换机发往主机的报文，分 2 种情况：

第 1 种：VLAN 与 PVID 一致，Tag=5，剥离 VLAN Tag，还原成 Untagged 报文，见方框右边 P1 报文处理；

第 2 种：VLAN 与 PVID 不一致，例如 Tag=10，直接丢弃，见方框右边 P2 报文处理。

记忆技巧：与 PVID 不一致的报文全部丢弃。Access 口只允许一种 VLAN Tagged 报文通过。

Access 口举例说明：如图 3.17 所示，主机 A 发 Untagged 报文到 SWA 的 G0/0/1 接口，SWA 添加 VLAN Tag =10，报文进入交换机之后，携带 VLAN Tag=10，然后交换机会从 G0/0/3 接口转发出去，因为 G0/0/3 接口的 VLAN Tag =10，发出去的报文不带 VLAN Tag。报文不会从 G0/0/2 接口转发出去，因为 G0/0/2 接口的 PVID =2，与报文本身携带的 VLAN Tag 不一致，根据规则，直接丢弃。

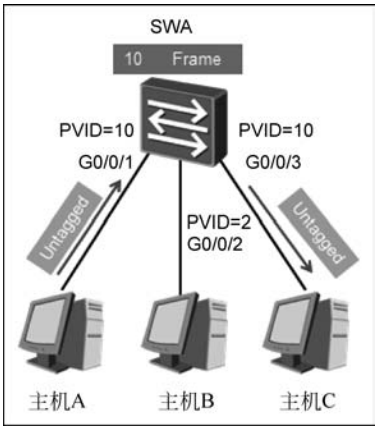


图 3.17 Access 口工作示例

Access 口配置如图 3.18 所示。

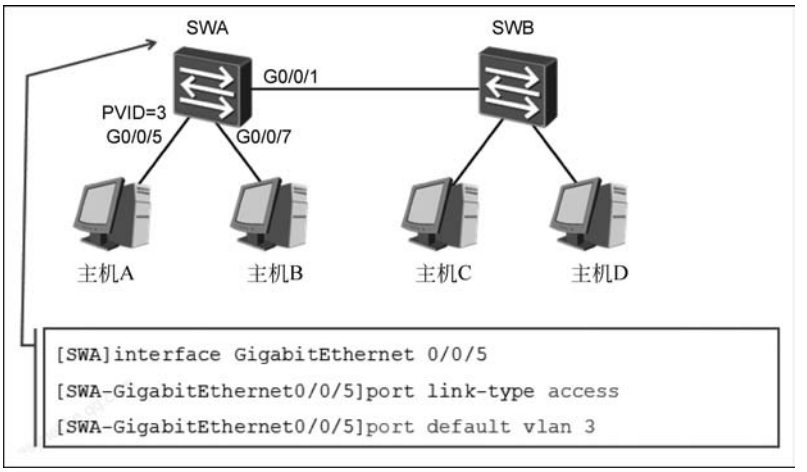


图 3.18　Access 口配置

进到接口模式，将接口配置为 Access 口：port link-type access。

给接口指定 PVID：port default vlan 3。

3.2.3　Trunk 口工作原理

Trunk 口的工作机制与 Access 口最大的不同点就是交换机之间允许多种不同 VLAN tagged 报文通过，因此相比 Access 口来说，多了一个 allow pass 控制列表，用来控制让哪几个 VLAN Tag 通过，实际配置的命令是 port trunk allow vlan 5 10，指定让 VLAN Tag 5、10 通过。

如图 3.19 所示，左边上下两个方框表示交换机，上方交换机的接口 PVID＝5，从它的角度看收发报文的处理情况。

收方向：指上方交换机接收下方交换机发来的报文，分 3 种情况：

第 1 种：收到 Untagged 报文，添加 VLAN Tag＝5，见方框左边 P1 报文处理。

第 2 种：收到在 allow pass 范围内的 VLAN Tag 报文，直接通过，见方框左边 P2、P3 报文。

第 3 种：收到带有 VLAN Tag 报文，但是该 VLAN Tag 不在 allow pass 范围内，丢弃，见方框左边 P4 报文。

发方向：指上方交换机发报文给下方交换机，分 3 种情况：

第 1 种：报文带的 VLAN Tag 在 allow pass 范围内，同时又和 PVID 相同，剥掉 VLAN Tag，发送 Untagged 报文，见方框右边 P1 报文处理。

第 2 种：报文带的 VLAN Tag 在 allow pass 范围内，但是和 PVID 不一样，直接发走，见方框右边 P2 报文处理。

第 3 种：报文带的 VLAN Tag 不在 allow pass 范围内，直接丢弃，见方框右边 P3 报文的处理。

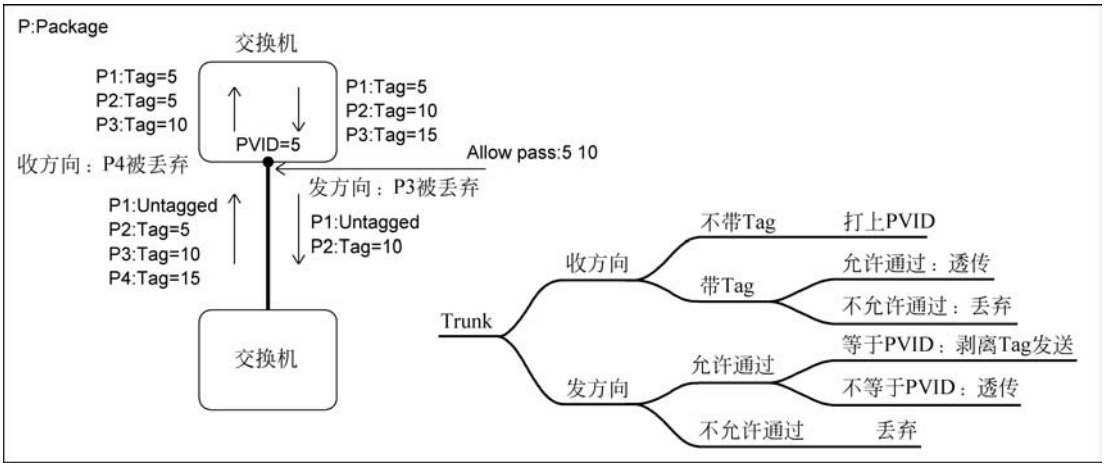


图 3.19　Trunk 口工作机制

Trunk 口处理方式和 Access 有点类似，不同点就是多了一个 allow pass 控制列表。

Trunk 口举例说明，如图 3.20 所示，主机 A、C 属于 VLAN 1，主机 B、D 属于 VLAN 20，交换机连接主机的接口都是 Access 口，SWA 与 SWB 之间的接口是 Trunk 口，双方配置的 VLAN 控制列表都是 allow pass vlan 1 20。

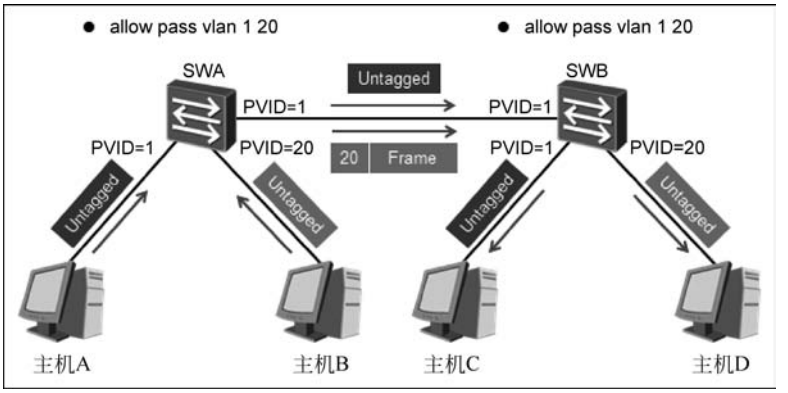


图 3.20　Trunk 口工作示例

主机 A 发送 Untagged 报文给 SWA，SWA 添加 VLAN 1，这个报文会发往 SWB，VLAN 1 在 SWA 的 allow pass 范围内，同时又和接口的 PVID 一致，根据 Trunk 口的规则，会剥掉 VLAN 标签，变成 Untagged 报文发往 SWB。SWB 收到 Untagged 报文时会添加 VLAN Tag=1，VLAN Tag=1 在 SWB 的 allow pass 范围内，所以会放行并转发给主机 C。

主机 B 发送 Untagged 报文给 SWA，SWA 添加 VLAN Tag＝20。SWA 将报文发送给 SWB 时，因为其 VLAN Tag 20 在 allow pass 范围内，而且与接口 PVID 不一致，所以将其直接发给 SWB。SWB 收到这个报文时判断 VLAN Tag 20 在它的 allow pass 范围内，所以会转发给主机 D。

通常 Trunk 口的 PVID 都使用默认的 VLAN 1，如果要修改 Trunk 口的 PVID，需要确保链路两边接口的 PVID 一致，如果不一致会导致报文无法被正确转发。可以尝试分析一下，如果 SWA 右边接口的 PVID＝1，SWB 左边接口的 PVID＝20，报文是否能正确被转发到目标主机。

Trunk 口配置如图 3.21 所示。

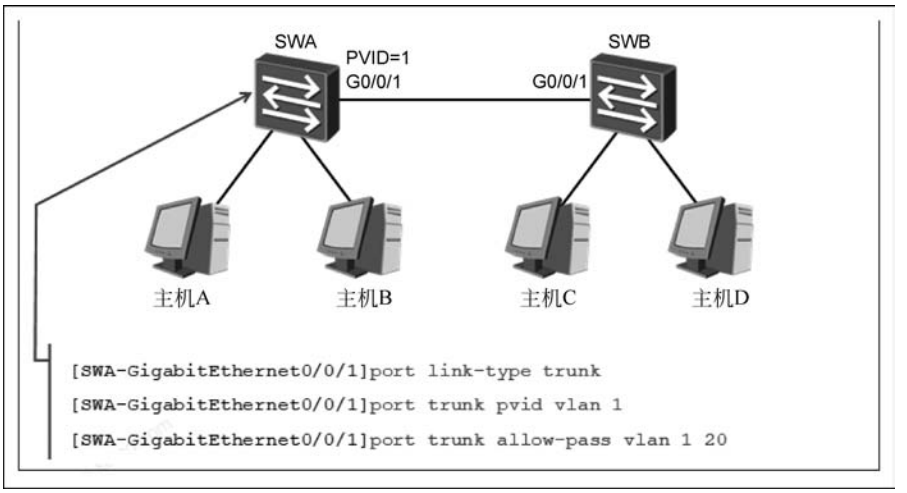


图 3.21　Trunk 口配置命令

3.2.4　Hybrid 口工作原理

除了 Access 口和 Trunk 口外，华为交换机还支持一种混合接口，称之为 Hybrid 口，如图 3.22 所示，Hybrid 口可以用在交换机和主机之间，还可以用于两个交换机之间。

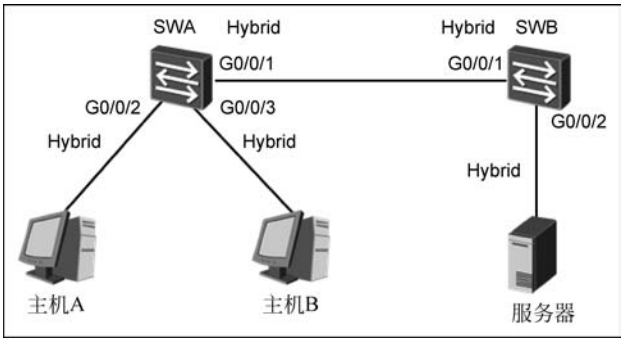


图 3.22　Hybrid 口应用场景

Hybrid 口的工作机制与 Trunk 口最大的不同点就是 Hybrid 口有 2 个控制列表，一个是 Tagged VLAN 列表，另一个是 Untagged VLAN 列表，用来更精准地控制报文。

如图 3.23 所示，左边上下两个方框表示交换机，上方交换机的接口 PVID=5，从它的角度看收发报文的处理情况。

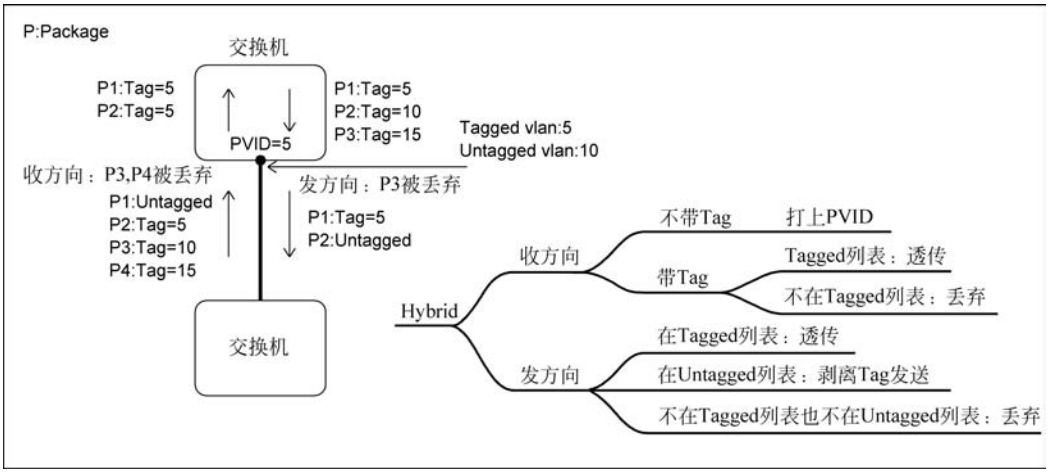


图 3.23　Hybrid 口工作机制

收方向：指上方交换机接收下方交换机发来的报文，分 3 种情况：

第 1 种：收到 Untagged 报文，添加 VLAN Tag=5，见左边 P1 报文处理。

第 2 种：收到带有 VLAN Tag 报文，且 VLAN Tag 在 Tagged VLAN 列表里，见左边 P2 报文处理。

第 3 种：收到带有 VLAN Tag 报文，但该 VLAN Tag 不在 Tagged VLAN 列表里，丢弃，见左边 P3、P4 报文处理。

发方向：指上方交换机发报文给下方交换机，分 3 种情况：

第 1 种：报文带的 VLAN Tag 在 Tagged VLAN 列表里，透传，见右边 P1 报文处理。

第 2 种：报文带的 VLAN Tag 在 Untagged VLAN 列表里，剥掉 VLAN Tag 发出去，见右边 P2 报文处理。

第 3 种：报文带的 VLAN Tag 不在 Tagged VLAN 列表又不在 Untagged VLAN 列表里，直接丢弃，见右边 P3 报文的处理。

Hybrid 口举例说明，如图 3.24 所示，交换机所有接口都是 Hybrid 口，主机 A 属于 VLAN2，主机 B 属于 VLAN3，服务器属于 VLAN100，主机 A 和主机 B 属于不同部门不能互相通信，但是都可以访问服务器。交换机端口配置如下：

SWA 端口 1 配置 tagged list vlan 2 3 100，untagged 列表随意，不配置也可以；

SWA 端口 2 配置 untagged list vlan 2 100，tagged 列表随意；

SWA 端口 3 配置 untagged list vlan 3 100，tagged 列表随意；

SWB 端口 1 配置 tagged list vlan 2 3 100，untagged 列表随意；

SWB 端口 2 配置 untagged list vlan 2 3 100，tagged 列表随意。

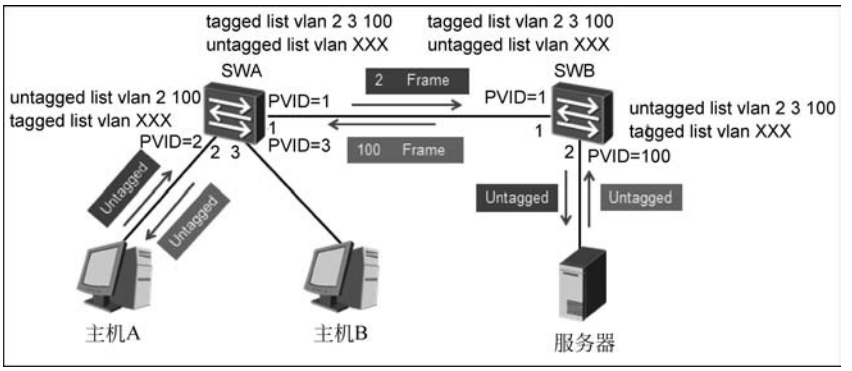


图 3.24　Hybrid 口工作示例

主机 A 发送 Untagged 报文给 SWA，SWA 的 2 号端口会添加 VLAN2，VLAN2 在该端口的 Untagged 列表里，所以继续转发（如果不在列表里，会直接丢弃）。SWA 的 1 号端口会将报文发出去，因为 VLAN2 在该端口的 Tagged 列表里，所以发出去的报文带有 VLAN2。该报文到达 SWB 后，因为 VLAN2 也在 SWB 的 Tagged 列表里，所以继续转到 SWB 的 2 号端口。因为 VLAN2 在 2 号端口的 Untagged 列表里，所以 SWB 剥掉 VLAN 之后将其发给服务器。

反方向：服务器回一个 Untagged 报文到达 SWB 的 2 号端口时会加上 VLAN100，VLAN100 在该端口的 Untagged 到表里，所以继续从 1 号端口发出去，因为 VLAN100 在 1 号端口的 Tagged 列表里，所以 SWB 发给 SWA 的报文带有 VLAN100，到达 SWA 的 1 号端口时，因为 VLAN100 在该端口的 Tagged 列表里，所以继续转发给 SWA 的 2 号端口，VLAN100 又在 SWA 的 2 号端口的 Untagged 列表里，所以会剥掉 VLAN100 变成 Untagged 报文发给主机 A。

主机 B 和服务器的通信过程与主机 A 相似，可以试着分析一下。相关配置命令如图 3.25 所示。

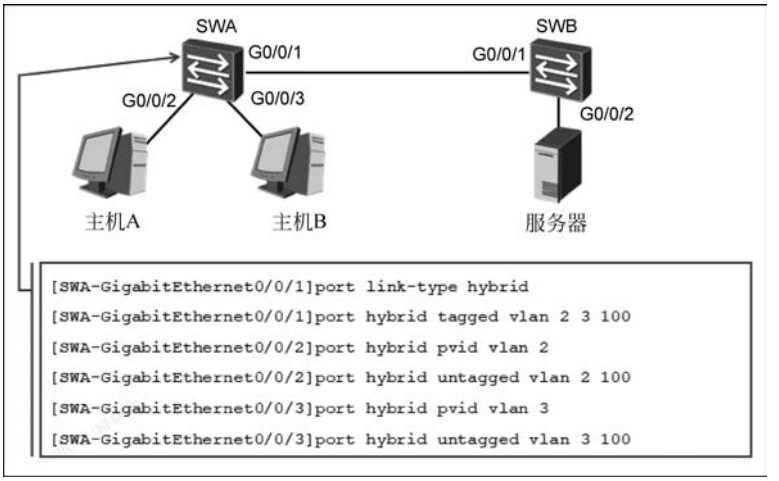


图 3.25　Hybrid 口配置命令

华为交换机的接口支持 3 种模式，分别是 Access、Trunk、Hybrid。出厂默认的接口模式是 Hybrid，默认的 VLAN Tag=1。

初次接触 VLAN 配置会容易搞混，因为规则太多，最好的解决办法就是多做实验，多练习，也可以自己设计一些场景，然后做实验验证。

3.2.5 实验演示

做实验之前先介绍几个小技巧，方便问题定位解决。

实验小技巧：

① 查看当前接口的配置情况，可以检查当前接口是不是漏配、错配，如图 3.26 所示；

```
[Huawei]inter e0/0/1
[Huawei-Ethernet0/0/1]display this
#
interface Ethernet0/0/1
 port hybrid pvid vlan 100
 port hybrid untagged vlan 2 to 3 100
#
return
```

图 3.26 检查当前接口配置

② 端口模式切换之前，要删除接口下的所有配置，如果不删除，系统会提示错误，删除的时候用 undo 命令，如图 3.27 所示；

```
[Huawei-Ethernet0/0/1]undo port hybrid pvid vlan
[Huawei-Ethernet0/0/1]undo port hybrid untagged vlan 2 3 100
[Huawei-Ethernet0/0/1]
```

图 3.27 删除接口下的配置

③ 可以用 display current-configuration 查看当前设备的所有配置，如图 3.28 所示。

```
[Huawei]display current-configuration
#
sysname Huawei
#
vlan batch 2 to 3 100
#
cluster enable
ntdp enable
ndp enable
#
interface Vlanif1
#
interface MEth0/0/1
#
interface Ethernet0/0/1
 port link-type access
#
interface Ethernet0/0/2
#
interface Ethernet0/0/3
 port hybrid tagged vlan 2 to 3 100
#
interface Ethernet0/0/4
#
interface Ethernet0/0/5
```

图 3.28 查询当前所有配置

1. Access 口基本实验

如图 3.29 所示，交换机接口的 PVID＝2，主机的 IP 地址分别是 192.168.1.1/24 和 192.168.1.2/24。

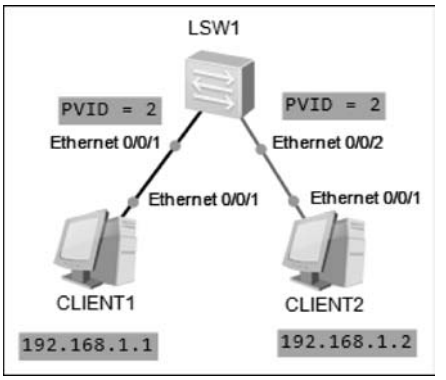


图 3.29　Access 口实验拓扑

步骤 1：配置主机 IP，双击 CLIENT1，配置 IP 地址和子网掩码，如图 3.30 所示。

图 3.30　主机的 IP 配置

CLIENT2 的配置与此相似，把 IP 地址换成 192.168.1.2 即可。

步骤 2：配置交换机，双击 LSW1，配置接口模式和 PVID，如图 3.31 所示。

注：配置之前要先在系统模式下创建 vlan，使用命令 vlan 2。如果不创建 vlan 会导致通信无法建立。

步骤 3：验证实验结果。主机和交换机都配置好后，可以到主机中验证实验是否成功，如图 3.32 所示。

2. Trunk 口实验演示

如图 3.33 所示，PC1 和 PC3 属于 VLAN2，PC2 和 PC4 属于 VLAN3。

步骤 1：配置 LSW1，配置命令如图 3.34 所示。

步骤 2：配置 LSW2，配置命令如图 3.35 所示。

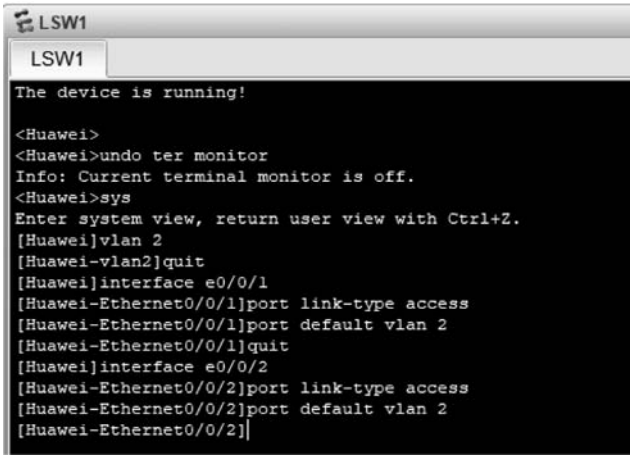


图 3.31　配置交换机接口

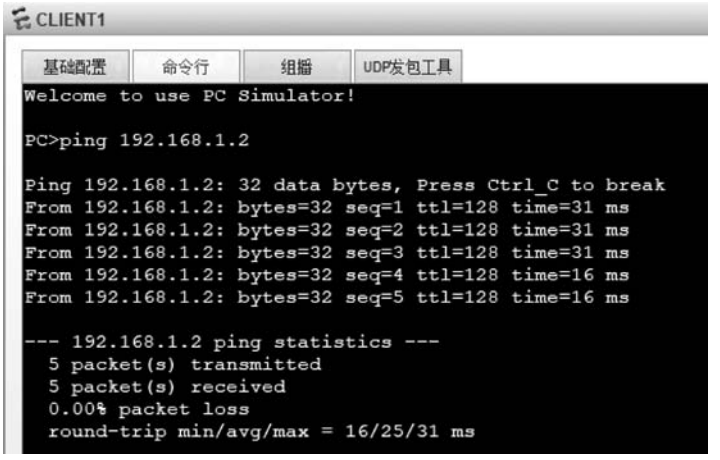


图 3.32　验证配置

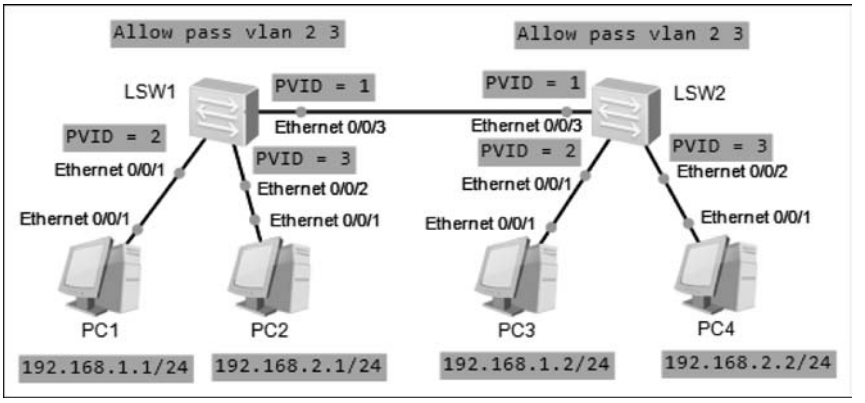


图 3.33　Trunk 口实验拓扑

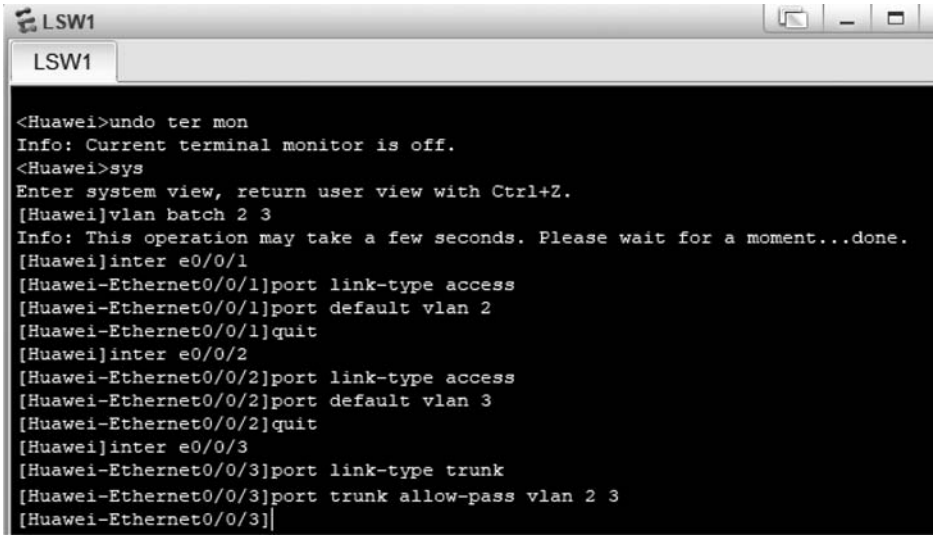


图 3.34　配置 LSW1

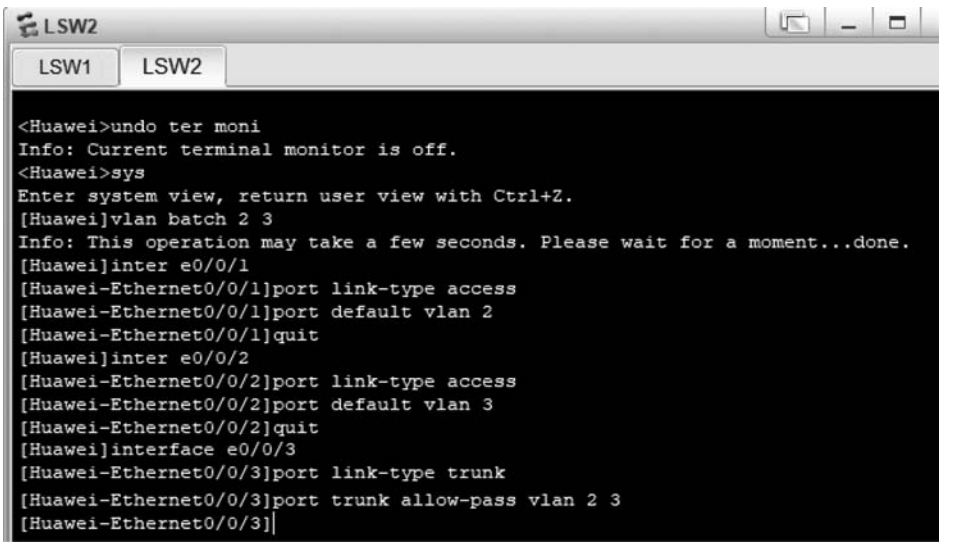


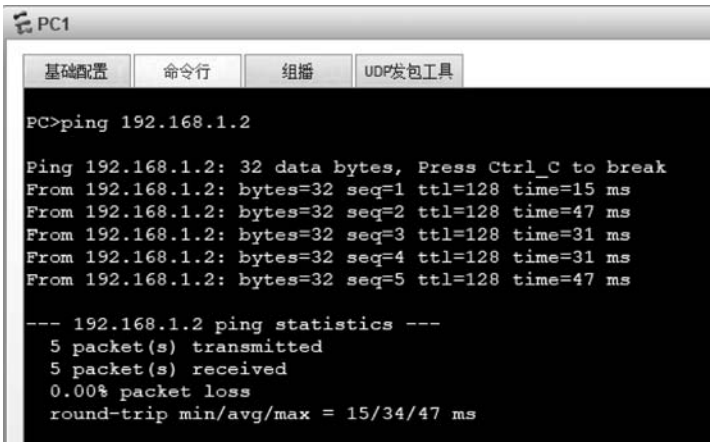
图 3.35　配置 LSW2

步骤 3：配置各个 PC，配置界面如图 3.36 所示。PC2、PC3、PC4 的配置与此相似，修改相应 IP 地址就可以了。

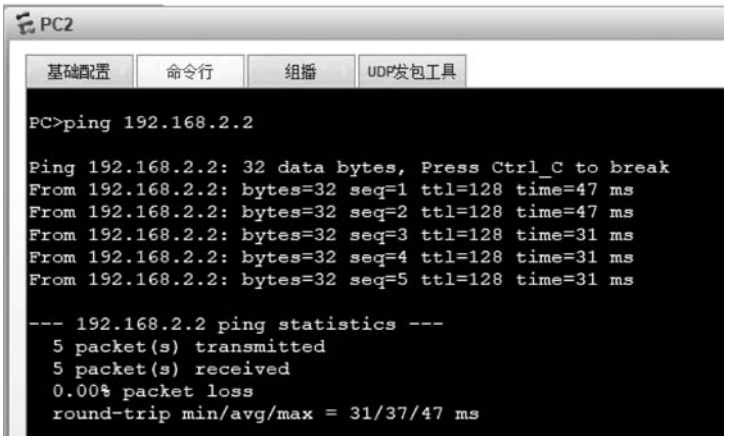
步骤 4：验证配置结果，PC1 可以 ping 通 PC3 的 IP 地址，PC2 可以 ping 通 PC4 的 IP 地址，如图 3.37(a)和图 3.37(b)所示。

实验拓展，自己尝试做以下验证，增加对 Trunk 口转发机制的理解：

图 3.36 配置 PC1



(a) 实验验证——PC1 ping PC3



(b) 实验验证——PC2 ping PC4

图 3.37

1. 在 LSW1 和 LSW2 之间的链路上启动抓包，PC1 ping PC3，PC2 ping PC4，PC1 ping PC4 观察 ping 报文的 vlan 携带情况；

2. 将 LSW1 和 LSW2 的 Trunk 口 PVID 都改成 2，再抓包分析；

3. 将 LSW1 的 Trunk 口 PVID 改成 2，LSW2 的 Trunk 口 PVID 改成 3，再抓包分析。

3. Hybrid 口实验演示

如图 3.38 所示，PC1 属于 VLAN 2，PC2 属于 VLAN 3，PC3 属于 VLAN 100，PC1 和 PC2 不能互通，但是都可以访问 PC3。

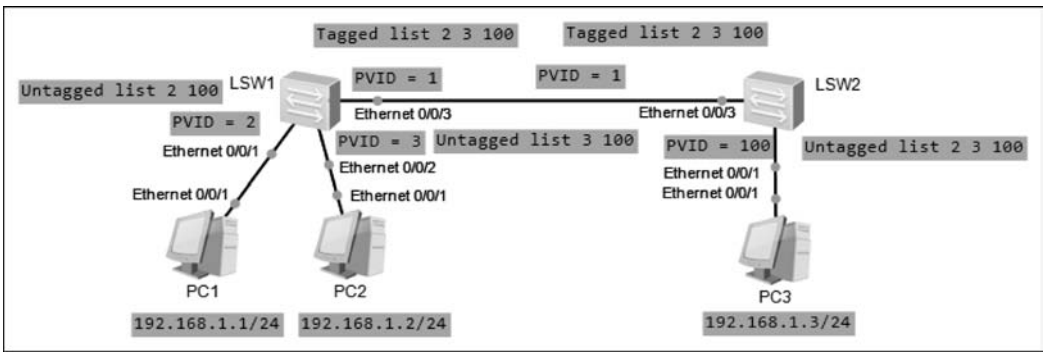


图 3.38　Hybrid 实验拓扑

步骤 1：配置 LSW1，如图 3.39 所示。

```
Enter system view, return user view with Ctrl+Z.
[Huawei]vlan batch 2 3 100
Info: This operation may take a few seconds. Please wait for a moment...done.
[Huawei]inter e0/0/1
[Huawei-Ethernet0/0/1]port link-type hybrid
[Huawei-Ethernet0/0/1]port hybrid pvid vlan 2
[Huawei-Ethernet0/0/1]port hybrid untagged vlan 2 100
[Huawei-Ethernet0/0/1]quit
[Huawei]inter e0/0/2
[Huawei-Ethernet0/0/2]port link-type hybrid
[Huawei-Ethernet0/0/2]port hybrid pvid vlan 3
[Huawei-Ethernet0/0/2]port hybrid untagged vlan 3 100
[Huawei-Ethernet0/0/2]quit
[Huawei]interface e0/0/3
[Huawei-Ethernet0/0/3]port link-type hybrid
[Huawei-Ethernet0/0/3]port hybrid tagged vlan 2 3 100
[Huawei-Ethernet0/0/3]
```

图 3.39　配置 LSW1

步骤 2：配置 LSW2，如图 3.40 所示。

步骤 3：配置 PC1 的 IP 地址和子网掩码，如图 3.41 所示，PC2 和 PC3 的配置与此相似。

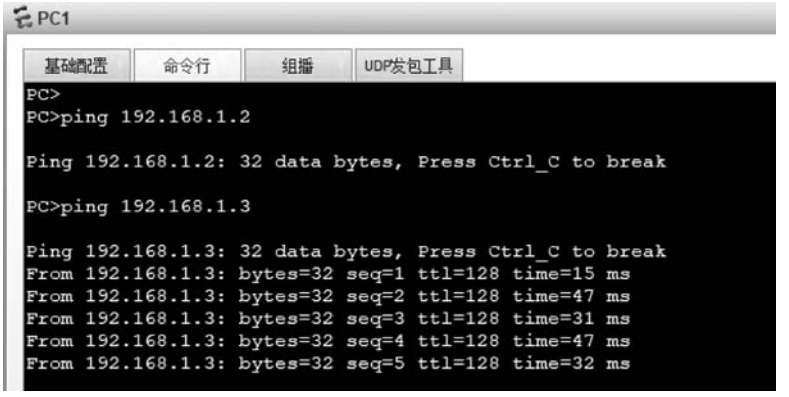
步骤 4：验证 PC1 与 PC3，PC2 与 PC3，PC1 与 PC2 的连通性，如图 3.42(a)和(b)所示。

附加练习：下面是一个综合实验，做完之后可以加深对各种模式的理解，如图 3.43 所示。

```
<Huawei>undo ter mon
Info: Current terminal monitor is off.
<Huawei>sys
Enter system view, return user view with Ctrl+Z.
[Huawei]vlan batch 2 3 100
Info: This operation may take a few seconds. Please wait for a moment...done.
[Huawei]inter e0/0/1
[Huawei-Ethernet0/0/1]port link-type hybrid
[Huawei-Ethernet0/0/1]port hybrid pvid vlan 100
[Huawei-Ethernet0/0/1]port hybrid untagged vlan 2 3 100
[Huawei-Ethernet0/0/1]quit
[Huawei]inter e0/0/3
[Huawei-Ethernet0/0/3]port link-type hybrid
[Huawei-Ethernet0/0/3]port hybrid tagged vlan 2 3 100
[Huawei-Ethernet0/0/3]
```

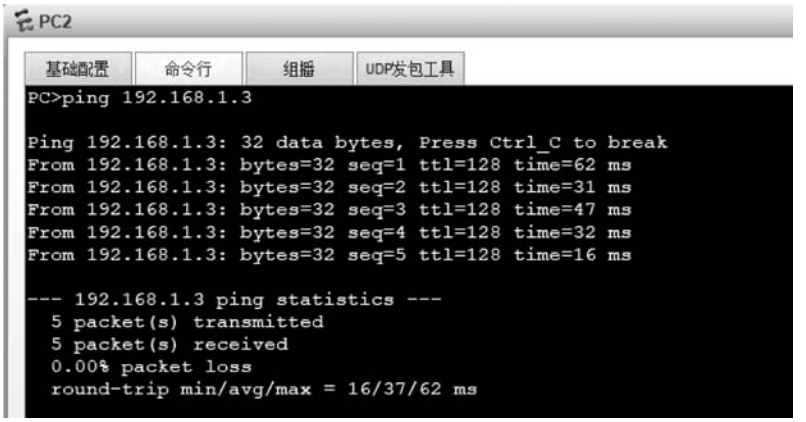
图 3.40　配置 LSW2

图 3.41　配置 PC1 的 IP 地址和子网掩码



(a) 实验验证——PC1 ping PC2/PC3

图　3.42



（b）实验验证——PC2 ping PC3

图 3.42　（续）

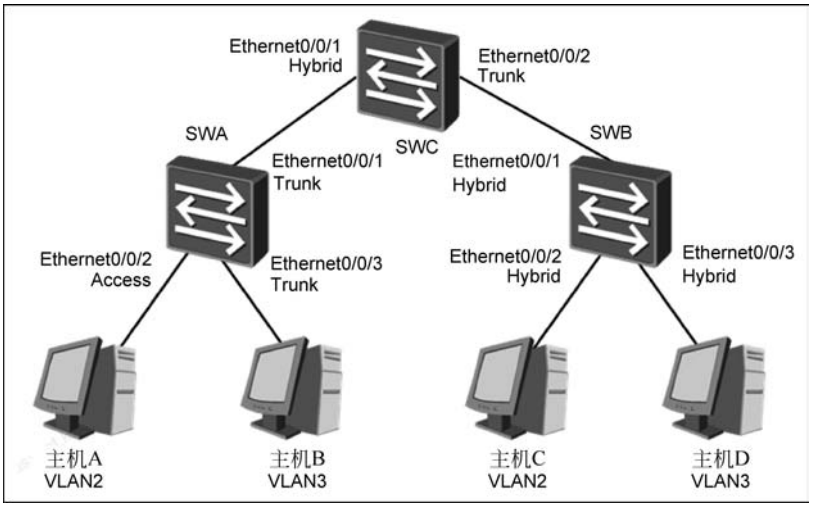


图 3.43　综合实验拓扑

3.2.6　小结

本节介绍了 VLAN 的应用场景，以及 VLAN 的实现方法，还介绍了华为交换机接口的 3 种模式，分别是 Access、Trunk、Hybrid，最后对各种模式做了实验演示。

本节内容相对比较抽象，需要记忆的规则较多，同时又非常重要，日常工作中经常用到，需要掌握到融会贯通的程度。建议多做实验，可以增加命令熟练度，又可以增加对各种不同模式的理解。

3.3 STP 原理与配置

为了提高网络可靠性，二层交换网络中通常会使用冗余链路。然而，冗余链路会带来环路问题。STP 协议(Spanning Tree Protocol，生成树协议)可以避免网络环路带来的问题，还可以在链路故障的时候自动恢复业务。

3.3.1 二层环路带来的问题

如图 3.44 所示，为了提高链路可靠性，交换机之间都采用备份链路，防止链路故障的时候业务中断。

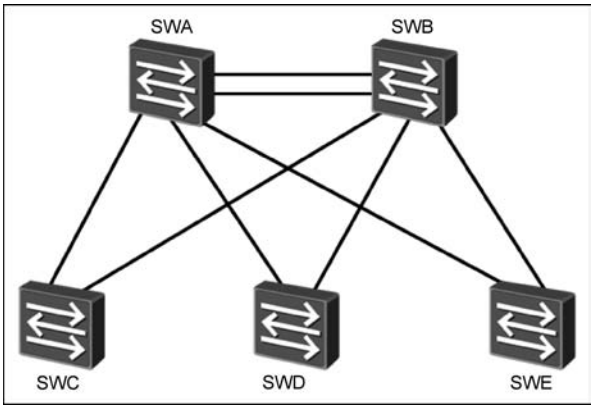


图 3.44 二层冗余链路

广播风暴问题：如图 3.45 所示，交换机 SWA、SWB、SWC 形成环路，主机 A 和主机 B 通信的时候，首先要通过 ARP 协议获得对方的 MAC 地址，ARP 协议的目标 MAC 是广播 MAC 地址 FF-FF-FF-FF-FF-FF。

主机 A 发送一个广播报文给 SWB，SWB 会给 0、1 端口各发一份，SWA 收到之后会给 0 端口发一份，SWC 收到之后会给 1、2 端口各发一份，然后又回到 SWB，此时 SWB 还会继续转发，给 0、2 端口各发一份，如此循环不停，顺时针和逆时针方向都会有环路。

广播报文随着时间推移会不断累加，最终设备端口带宽都被占满，设备崩溃。

MAC 地址表振荡问题：如图 3.46 所示，主机 A 发送 ARP 报文给 SWB，SWB 会分析该报文的源 MAC 地址，并更新到 MAC 地址表，添加表项：00-05-06-07-08-AA 端口 G0/0/3。

因为这是一个广播报文，经过 SWA、SWC 之后又回到 SWB，此时是从 G0/0/2 收到的，因此 SWB 会更新 MAC 地址表，将原来的删掉，更新为：00-05-06-07-08-AA 端口 G0/0/2。

此时如果有其他主机发送报文给主机 A，SWB 查表发现主机 A 在 G0/0/2 端口下，报文发往此端口将无法正确转发给主机 A，导致业务中断。稍后主机 A 再次发出 ARP 报文，SWB 又重新更新到正确的表项，这样就会导致 MAC 地址表振荡，业务时通时不通。

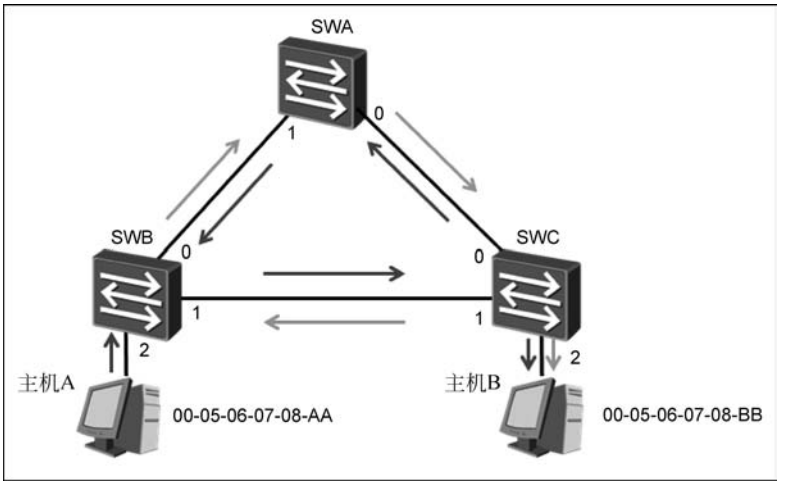


图 3.45　环路广播风暴示意图

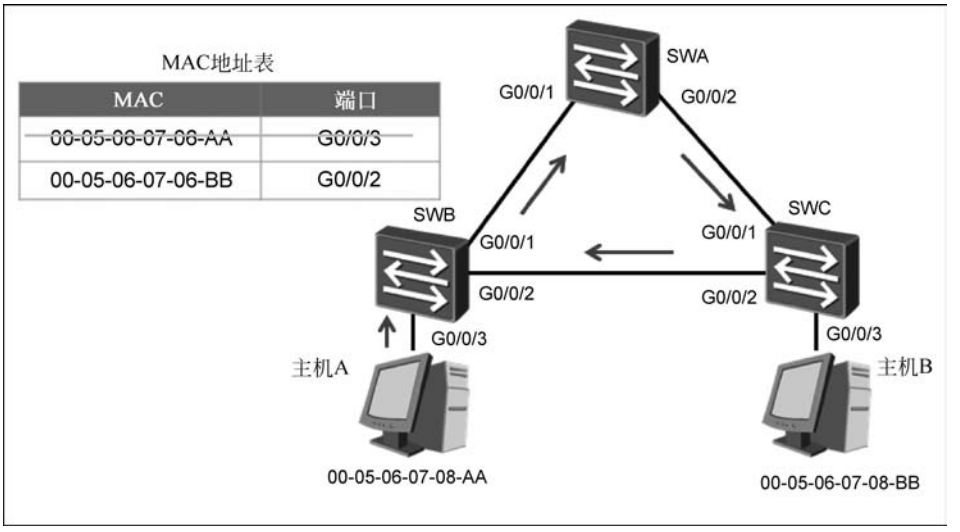


图 3.46　MAC 地址表振荡示意图

因为有广播风暴和 MAC 地址表振荡问题，二层网络不应该存在环路。破除环路有两种办法，一种是手动拔插，但是业务中断时间较长，另外一种是靠协议自动控制，这个协议就是 STP 协议。

3.3.2　STP 基本原理

前面介绍以太网帧结构的时候，介绍了两种帧结构，一种是 Ethernet_II 帧，这是用来封装实际业务报文的，例如 ping 报文，另一种是 IEEE 802.3 帧，这是用来封装二层协议报文的，STP 报文就是用这种格式。

如图 3.47 所示，STP 通过 BPDU（Bridge Protocol Data Unit，桥协议数据单元）交互信息。

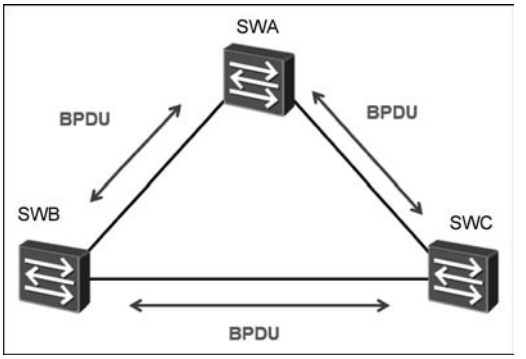


图 3.47　BPDU 交互

STP 使用的帧结构，如图 3.48 所示，里面封装了当前交换机相关的一系列信息，如交换机优先级、MAC 地址、接口数量、各接口带宽、接口优先级等。

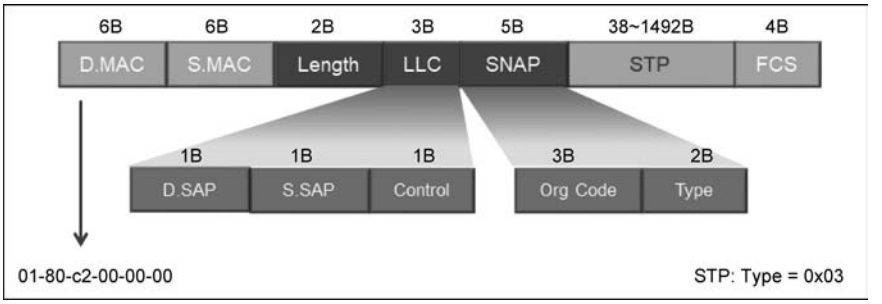


图 3.48　STP 帧结构

交换机收集这些信息之后就可以计算出应该阻塞哪个端口。如图 3.49 所示，SWC 通过计算得知左边端口应该阻塞，该端口被阻塞之后，网络中不存在环路，从而避免了网络风暴和 MAC 表振荡问题。

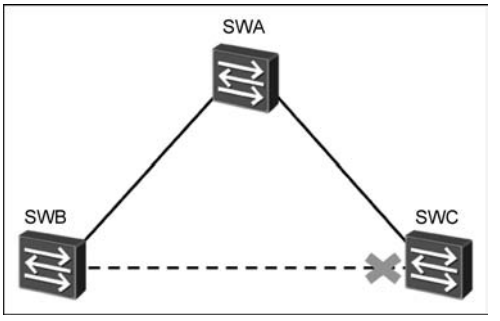


图 3.49　阻塞端口避免环路

注：交换机阻塞的是业务报文，STP 报文并没有被阻塞，BPDU 还是可以在 SWB 和 SWC 之间的链路上转发。

3.3.3　STP 计算过程

阻塞端口是怎么计算出来的呢？主要通过 4 个步骤，如图 3.50 所示：

步骤 1：根据各个交换机的优先级选取一个根桥(Root Bridge)；

步骤 2：每个非根交换机选取一个根端口(R：Root port)，即距离根桥最近的端口；

步骤 3：为每条链路选取一个指定端口(D：Designated port)，即距离根桥最近的端口；

步骤 4：非 R 非 D 端口就是阻塞端口(A：Alternative Port)。

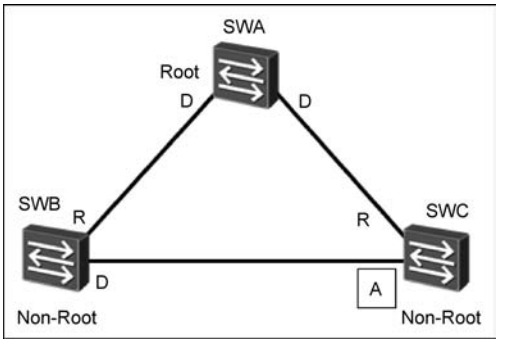


图 3.50　阻塞端口计算过程

注：步骤 2 是从交换机的角度来看，例如 SWB 有 2 个端口，比较哪个端口离根桥最近，步骤 3 是从链路的角度来看，例如图中下方那条链路，比较左右两个端口哪个距离根桥近。

下面介绍各个步骤的具体实现过程。

步骤 1：选取根桥。如图 3.51 所示，根据交换机优先级选取根桥，首先比较交换机优先级，优先级取值范围是 0～65 535，默认优先级为 32 768，如果优先级一样，就比较 MAC 地址，值越小优先级越高。

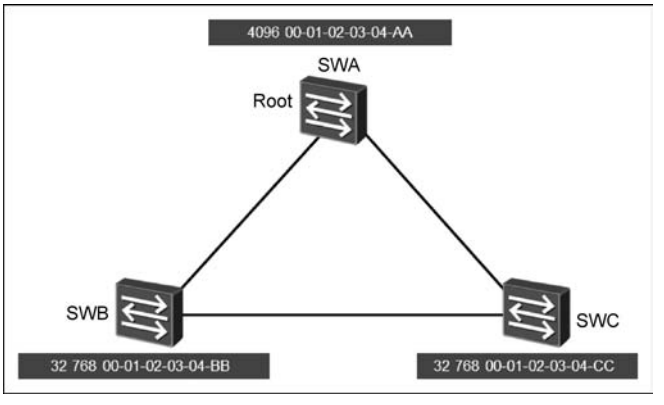


图 3.51　根桥选取

图中 SWA 的优先级是 4096，SWB 和 SWC 的优先级都是 32 768，因此 SWA 优先级最高，是根桥(Root)。此外，SWB 的 MAC 地址比 SWC 的值小，因此 SWB 的优先级又比 SWC 的高。

步骤 2：非根交换机选取根端口(R 端口)。如图 3.52 所示，共有 5 台交换机，分别是 SWA、SWB、SWC、SWD、SWE，其中 SWA 是根桥，SWB、SWC、SWD、SWE 是非根交换机，这一步就是为非根交换机选取根端口。

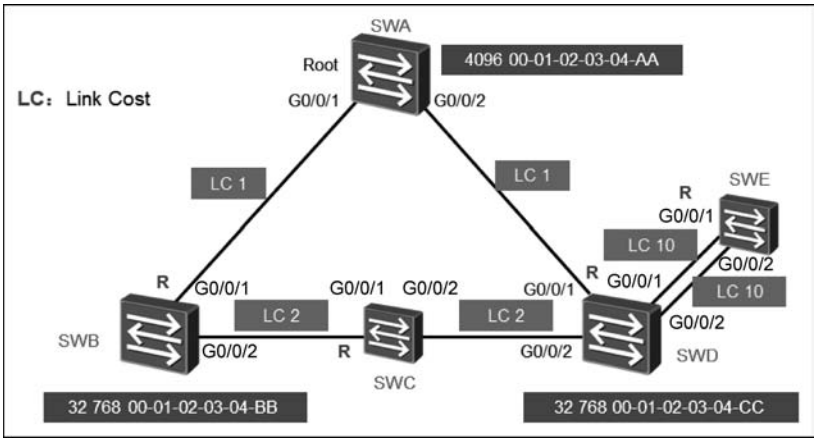


图 3.52　非根交换机选取根端口

根端口指离根桥最近的端口，与根桥距离的远近通过链路开销来计算，每条链路都有开销值，跟链路带宽有关，带宽越大开销越小，例如 100Mb/s 的接口开销是 1，50Mb/s 的接口开销是 2，10Mb/s 的接口开销是 10，简单的计算公式：100Mb/s/接口带宽＝开销。

图中 LC 指的就是链路开销，LC 1 指链路开销值为 1。

SWB 有 2 个接口，到达根桥 SWA 的路径分别是：

G0/0/1：SWB→SWA，途经 1 条链路。

G0/0/2：SWB→SWC→SWD→SWA，途经 3 条链路。

对应的开销分别是：

G0/0/1：1。

G0/0/2：2＋2＋1＝5。

G0/0/1 到达根桥的开销最小，因此 G0/0/1 就是 SWB 的根端口。SWD 根端口的选取与 SWB 相似。

SWC 有 2 个接口，到达根桥 SWA 的路径分别是：

G0/0/1：SWC→SWB→SWA，途经 2 条链路。

G0/0/2：SWC→ SWD→SWA，途经 2 条链路。

对应的开销分别是：

G0/0/1：2＋1＝3。

G0/0/2：2+1=3。

2个端口开销值一样，此时要判断接口对端交换机的优先级，接口G0/0/1对端的交换机是SWB，G0/0/2对方的交换机是SWD。根据前面介绍的规则，SWB的优先级高于SWD，因此G0/0/1是SWC的根端口。

SWE有两个端口，开销值一样都是11，对端是同一台交换机，优先级一样，此时要判断对端接口ID，SWE的G0/0/1对端的接口ID是SWD的G0/0/1，SWE的G0/0/2对端的接口ID是SWD的G0/0/2，值越小越优，因此SWE的根端口是G0/0/1。

步骤3：为每条链路选取一个指定端口（D端口）。如图3.53所示，3台交换机之间共有4条链路，分别是左上链路L1，右上链路L2，底部链路L3，右下角链路L4。

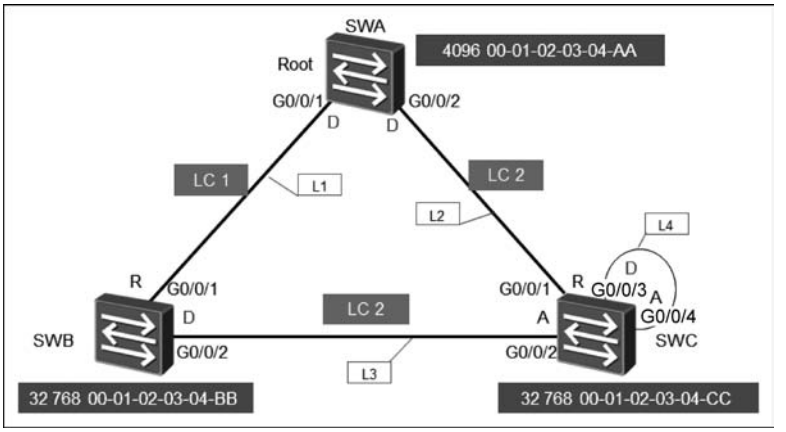


图3.53　选指定端口

根桥的所有端口都是指定端口，因此L1的D端口是SWA的G0/0/1口，L2的D端口是SWA的G0/0/2口。

L3左右两个端口怎么选D端口呢，和根端口类似，也是通过计算距离根桥的开销来确定的，左边端口的开销是2+1=3，右边端口的开销是2+2=4，左边的开销更小，因此L3的左边端口是D端口。

L4链路两端距离根桥的开销一样，交换机优先级一样，此时比较端口优先级，端口3的值小一点，优先级更高，因此SWC的G0/0/3端口是D端口。

步骤4：非R非D端口就是阻塞端口（A端口）。图3.53中，SWC的G0/0/2和G0/0/4不是R端口，也不是D端口，它就是A端口，因此会被阻塞，阻塞之后，网络中就不存在环路了。

总结一下STP的工作过程如下。

① 选取根桥（Root）：比较交换机的优先级，如果优先级一样，则比较交换机的MAC地址大小；

② 选取根端口（R端口）：比较交换机各端口到达根桥的cost，如果cost相等则比较对端交换机的优先级，如果对端交换机优先级相等则比较对端端口的优先级（端口也有优先级，默认值是128，一般不做配置，实际上比较的是端口的编号）；

③ 选取指定端口(D端口)：首先，根桥的每个端口都是D端口，接着，看其他不与根桥直连的链路，比较其两端到达根桥的cost，如果相等，则比较对端交换机优先级，如果优先级相等则比较对端端口号；

④ 非R非D端口就是A端口。

STP协议中，都是值越小越优，包括交换机的优先级、MAC地址、端口优先级、端口号。

所有这些参数都在STP的BPDU报文中交互，然后交换机通过计算，来确定本交换机各个端口的角色。

举例说明：如图3.54所示，共有SWA、SWB、SWC、SWD、SWE 5台交换机，各交换机的优先级、MAC地址和各个链路的cost见图中标注。其中SWB和SWD的优先级最高，都是4096，但是SWB的MAC地址比SWD小，因此SWB是根桥Root。

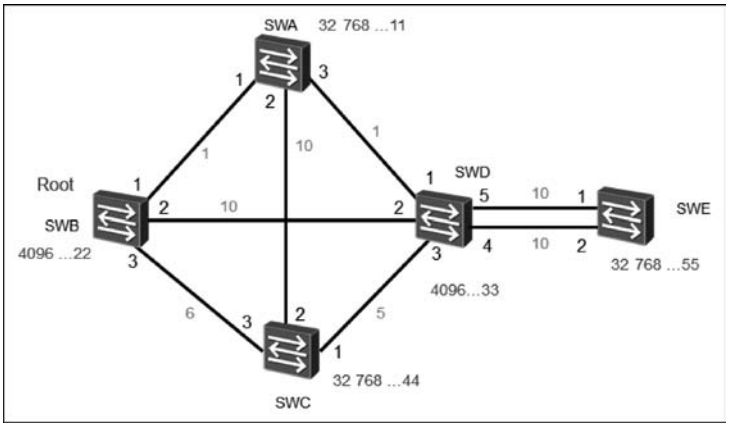


图3.54 STP拓扑示例

根端口选取：

SWA有3个端口，到达根桥SWB的路径和cost如下。

端口1：SWA→SWB，cost=1，根端口(R)；

端口2：SWA→SWC→SWB，cost=15；

端口3：SWA→SWD→SWB，cost=11。

SWC有3个端口，到达根桥SWB的路径和cost如下。

端口1：SWC→SWD→SWA→SWB，cost=7；

端口2：SWC→SWA→SWB，cost=11；

端口3：SWC →SWB，cost=6，根端口(R)。

SWD有3个端口，到达根桥SWB的路径和cost如下。

端口1：SWD→ SWA→SWB，cost=2，根端口(R)；

端口2：SWD→ SWB，cost=10；

端口3：SWD→SWC→SWB，cost=11。

SWE有2个端口，到达根桥SWB的cost相同，对端交换机优先级相同，取对端端口号

最小的，因此 SWE 的端口 2 是根端口(R)。各个交换机的 R 端口分布如图 3.55 所示。

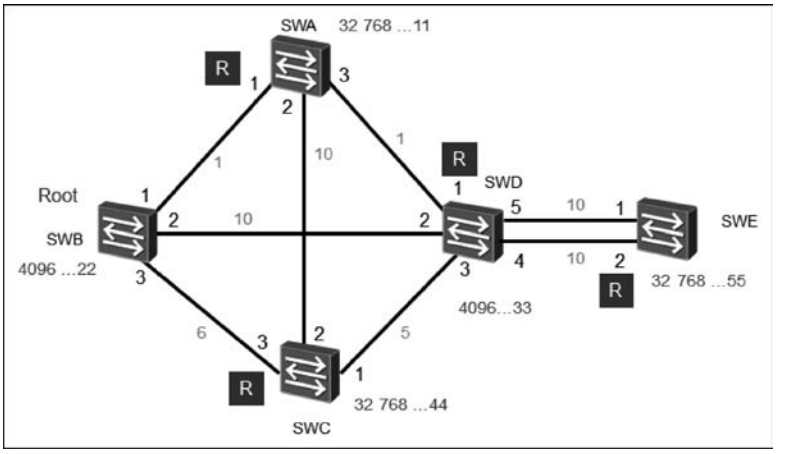


图 3.55　非根交换机的 R 端口选取

指定端口选取：

根交换机的所有端口都是 D 端口，因此 SWB 的 1、2、3 端口都是 D 端口；

SWA-SWC 之间的链路：上端 cost 为 10＋1＝11，下端 cost 为 10＋6＝16，因此 SWA 的端口 2 是 D 端口；

SWA-SWD 之间的链路：上端 cost 为 1＋1＝2，下端 cost 为 1＋10＝11，因此 SWA 的端口 3 是 D 端口；

SWC-SWD 之间的链路：上端 cost 为 5＋1＋1＝7，下端 cost 为 5＋6＝11，因此 SWD 的端口 3 是 D 端口；

SWD-SWE 之间的两条链路：左边接口距离根桥最近，因此 D 端口都在链路左边。

各链路的 D 端口分布如图 3.56 所示。

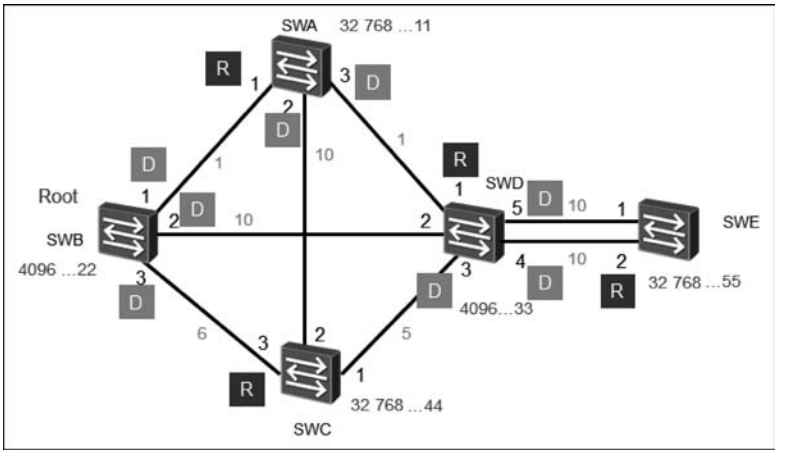


图 3.56　各链路指定端口选取

非 R 非 D 端口会被阻塞，因此最后的生成树如图 3.57 所示。

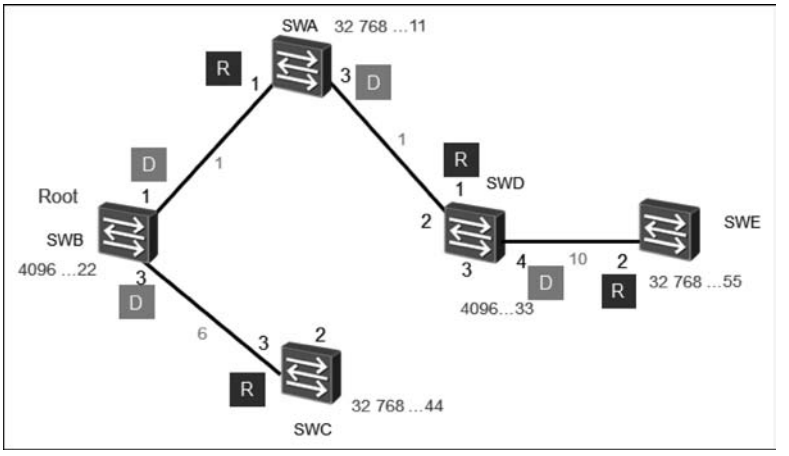


图 3.57 最终生成树

3.3.4 临时环路问题

如图 3.58 所示，最开始的时候 SWA 是 Root，SWC 的左边端口是 A 端口。后来修改了 SWC 的优先级，SWC 变成了 Root，根据规则，SWC 左边的端口肯定是 D 端口，如果 A 端口马上变为 D 端口，短时间内网络中还是会有环路存在。

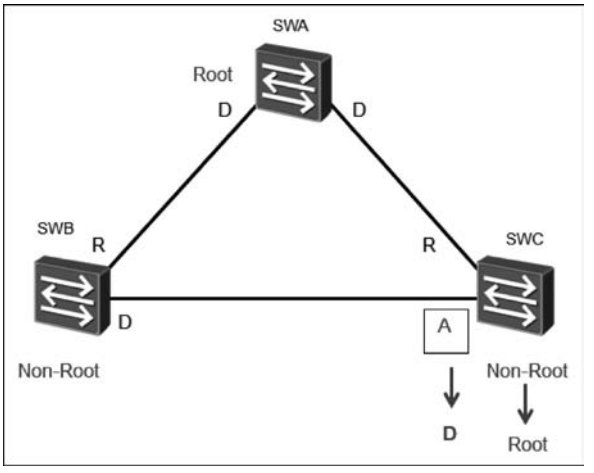


图 3.58 临时环路

为了避免临时环路存在，A 端口切换到 D 端口的时候，需要经过两个中间状态，每个状态持续 15s，共需要 30s。30s 内，新的 A 口被选出来，就可以避免临时环路的存在。

交换机端口共有 5 种状态，分别如下。

① Disabled：禁用状态。端口不能处理任何报文，使用命令 disable 后进入该状态；

② Blocking：阻塞状态。端口只能接收 BPDU，不能发送报文，A 端口处于这种状态；

③ Forwarding：转发状态。端口可以收发任何报文，R 端口和 D 端口处于这种状态；

④ Listening：侦听状态。端口可以收发 BPDU 报文，但不能收发业务报文；

⑤ Learning：学习状态。端口可接收业务报文(更新 MAC 表)，不能发送业务报文。

A 端口切换到 D 端口需要经过如下状态，如图 3.59 所示。

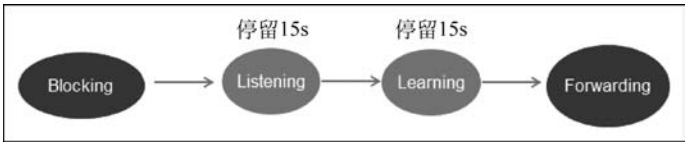


图 3.59　A 端口到 D 端口的切换过程

3.3.5　故障恢复过程

交换机在根桥选取前，都认为自己是根桥，主动发 BPDU 给旁边的交换机，各个交换机通过比较优先级选取出根桥之后，只有根桥才周期性发 BPDU，如图 3.60 所示。

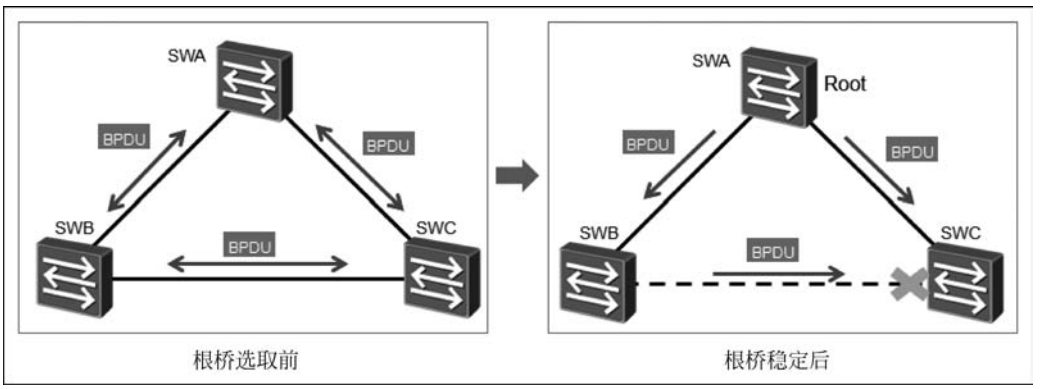


图 3.60　根桥选取前后 BPDU 发送情况

如果根桥出现故障，网络怎么恢复业务呢？如图 3.61 所示，SWA 是 Root，正常工作的时候会周期性发 BPDU 给各个交换机。如果 SWA 出现故障，网络中就没有 BPDU 更新，经过 20s 后，之前根桥发的 BPDU 老化删除，SWB 和 SWC 开始互发 BPDU，重新选取根桥。

SWC 左边的端口是 A 端口，BPDU 老化后，还要经过 Listening 和 Learning 这两个状态才能进入正常转发状态，需要时长 20＋15＋15＝50s，也就是说根桥失效后，恢复业务需要 50s。

如果链路出现故障，如何恢复呢？

如图 3.62 所示，SWB 和 SWA 之间的链路出现逻辑故障，而非物理故障。

SWB 认为 SWA 失效，因此尝试发送 BPDU 给 SWC，但是 SWC 能收到 SWA 发来的 BPDU，知道 Root 还正常工作，因此忽略来自 SWB 的 BPDU，但是 SWC 左边的 A 端口一直收不到 SWA 发的 BPDU，因此 20s 后会开始切换到 D 端口，A 端口切换到 D 端口还需要经过两个中间状态，因此恢复业务共需要 20＋15＋15＝50s。

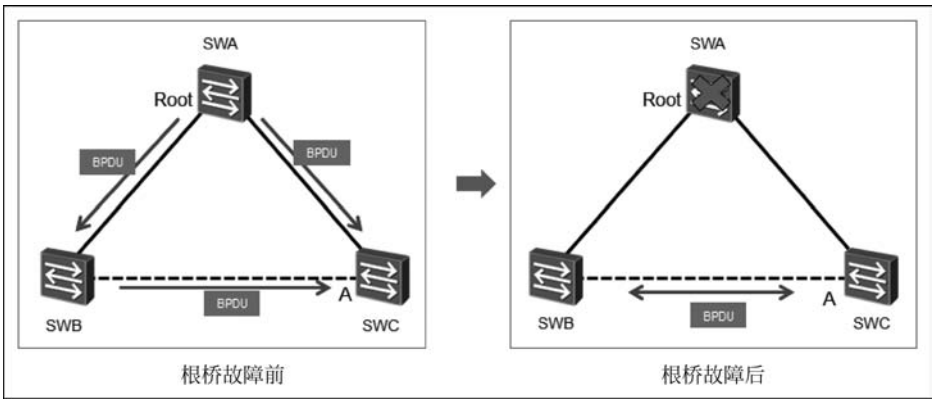


图 3.61　根桥故障场景

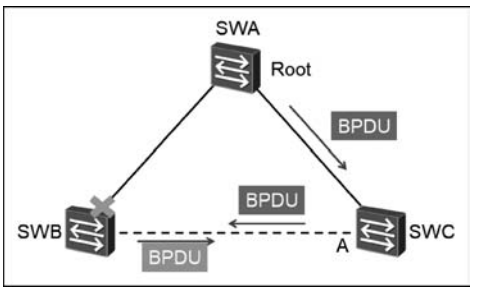


图 3.62　链路故障恢复

3.3.6　MAC 地址表错误问题

如图 3.63 所示，主机 A 和主机 B 的通信走上面的路径，SWB 的 MAC 地址表如图所示。

MAC地址表

MAC	端口
00-05-06-07-08-AA	G0/0/3
00-05-06-07-08-BB	G0/0/2

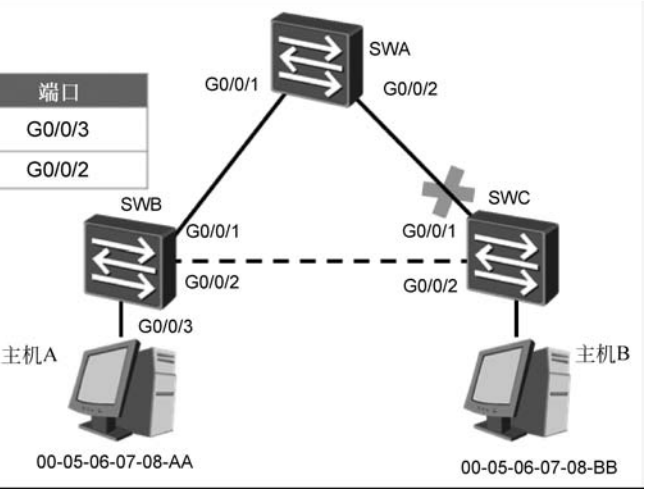


图 3.63　示例拓扑

后来 SWC 的 G0/0/1 口出现故障，路径改为下面那条链路，但是 SWB 的 MAC 表项没有更新，因为交换机的 MAC 表项是根据 ARP 报文来更新的，交换机链路故障，但是主机 A、主机 B 本身还有 ARP 缓存表，并不会再发 ARP 请求，所以不会更新。

交换机的 MAC 表老化时间是 300s，因此 300s 内，主机 A 发往主机 B 的报文，SWB 会从 G0/0/1 转发出去，最终无法到达主机 B。这种情况下，业务恢复时间需要 300s。

为了解决这个长时间业务中断问题，STP 引入了拓扑更新机制，如图 3.64 所示，SWC 感知到 G0/0/1 出现故障后，通过一系列动作强制刷新 MAC 地址表，具体过程如下。

① SWC 往 G0/0/2 发送 TCN(Topology Change Notification，拓扑变更通知)；

② SWB 收到 TCN 后，给 SWC 回 TCA，让 SWC 停止发 TCN；

③ SWB 给根桥 SWA 转发 TCN；

④ SWA 发 TC 给 SWB，通知 SWB 刷新 MAC 地址表；

⑤ SWB 给 SWC 转发 TC，通知 SWC 刷新 MAC 地址表。

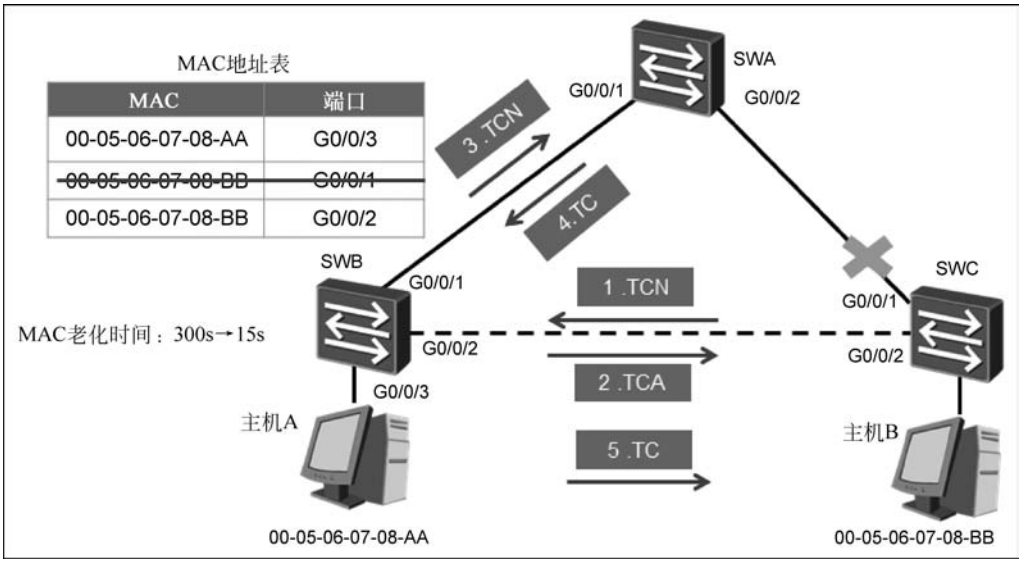


图 3.64　网络拓扑刷新过程

交换机收到从根桥发来的 TCN 后，将自己的 MAC 地址表老化时间改成 15s(可配置)，15s 后，MAC 地址表被清空，此时如果收到主机 A 发往主机 B 的报文时，对 SWB 来说就是未知单播，根据规则会泛洪到各个端口，SWC 也同样泛洪，业务恢复。后续再通过 ARP 报文更新 MAC 地址表。

这个 MAC 更新过程和 listening、learning 状态同时进行，因此网络最大业务中断时间是 50s。

3.3.7 STP 配置

如图 3.65 所示，STP 主要有以下几个常用配置命令：

stp disable：停止使用 STP 功能；

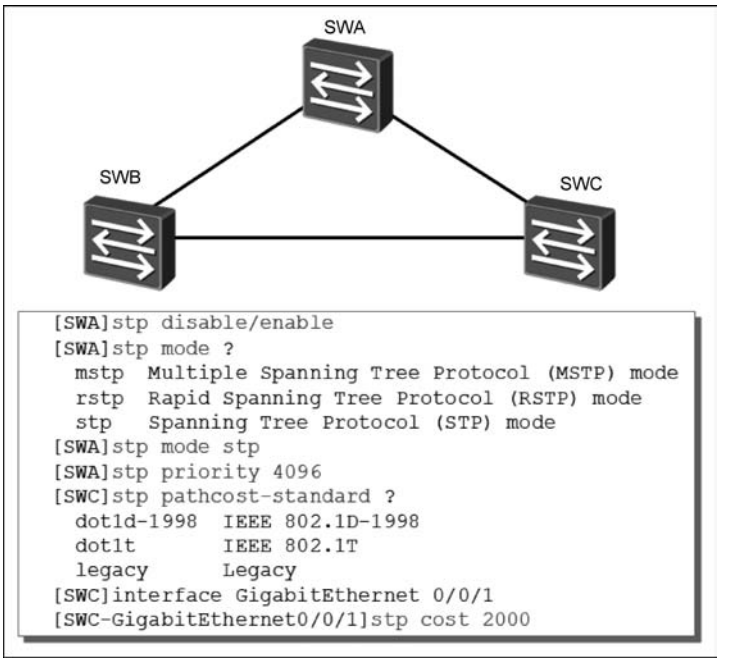
stp enable：开始使用 STP 功能；

stp mode：选择 STP 模式，共有 3 个，RSTP 和 MSTP 下一节再介绍，华为交换机默认开启 STP，模式是 MSTP；

stp priority：配置交换机的优先级；

stp pathcost-standard：指定链路 cost 的计算标准，同样的带宽，不同标准算出来的 cost 值不一样，默认采用 dot1t 标准；

stp cost 2000：对接口强制指定 cost 值，一般不建议这么配置，可能会产生次优路径。



Speed	Link type	802.1D cost	802.1t cost
10Mb/s	Half Duplex	100	2 000 000
	Full Duplex	95	1 999 999
	Aggregated link	90	1 000 000
100Mb/s	Half Duplex	19	200 000
	Full Duplex	18	199 999
	Aggregated link	15	100 000
1000Mb/s	Full Duplex	4	20 000
	Aggregated link	3	10 000

图 3.65 STP 配置命令和路径 cost 计算标准

查看实验结果，如图 3.66 所示，使用 display stp 查看相关信息，上面方框内是当前交换机信息，下面方框内是当前接口相关信息。

```
[Huawei]display stp
-------[CIST Global Info][Mode MSTP]-------
CIST Bridge          :32768.4c1f-cc04-3a3f  //当前交换机的优先级和MAC
Config Times         :Hello 2s MaxAge 20s FwDly 15s MaxHop 20 //每2s发一个BPDU，超时计时器是20s，每个
Active Times         :Hello 2s MaxAge 20s FwDly 15s MaxHop 20 状态延迟15s，网络最大条数20跳
CIST Root/ERPC       :32768.4c1f-cc04-3a3f / 0 //根桥的优先级和MAC，可以看出来当前交换机就是根桥
CIST RegRoot/IRPC    :32768.4c1f-cc04-3a3f / 0
CIST RootPortId      :0.0                      //CITS是MSTP的概念，先不管
BPDU-Protection      :Disabled
TC or TCN received   :5
TC count per hello   :0
STP Converge Mode    :Normal
Time since last TC   :0 days 0h:10m:44s
Number of TC         :4
Last TC occurred     :Ethernet0/0/1
----[Port1(Ethernet0/0/1)][FORWARDING]----
 Port Protocol       :Enabled
 Port Role           :Designated Port  //接口的角色，当前接口是D口
 Port Priority       :128  //端口优先级，端口优先级实际上是：128 + 端口ID
 Port Cost(Dot1T )   :Config=auto / Active=1 //接口开销，自动计算
 Designated Bridge/Port   :32768.4c1f-cc04-3a3f / 128.1
 Port Edged          :Config=default / Active=disabled
 Point-to-point      :Config=auto / Active=true
 Transit Limit       :147 packets/hello-time
 Protection Type     :None
 Port STP Mode       :MSTP  //STP模式，华为交换机默认是MSTP
 Port Protocol Type  :Config=auto / Active=dot1s
 BPDU Encapsulation  :Config=stp / Active=stp
 PortTimes           :Hello 2s MaxAge 20s FwDly 15s RemHop 20
 TC or TCN send      :2
 TC or TCN received  :4
 BPDU Sent           :319
         TCN: 0, Config: 0, RST: 0, MST: 319
 BPDU Received       :4
         TCN: 0, Config: 0, RST: 0, MST: 4
----[Port2(Ethernet0/0/2)][FORWARDING]----
```

图 3.66　查看 STP 相关参数

实验建议：自己设计各种拓扑，各链路都使用 Ethernet 100M 口，这样每一条链路的 cost 都一样，方便计算，如图 3.67 所示。

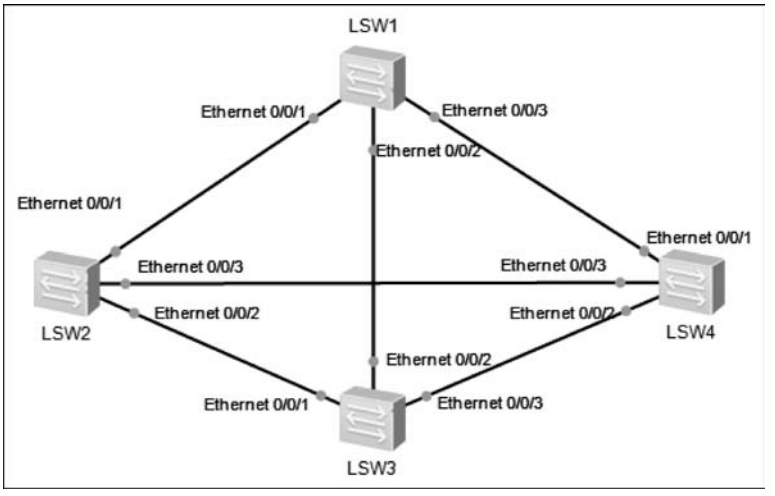


图 3.67　实验拓扑

拓扑搭建好之后，使用 display stp 查看各个交换机的优先级，然后按照前面介绍的方法计算根桥、根端口、指定端口、阻塞端口，再用命令查询各个接口的模式，验证 R 端口、D 端口、A 端口所处的位置是不是和计算的一致。

3.3.8 小结

本节介绍了 STP 的应用场景和工作原理，还介绍了网络发生故障时的恢复过程，以及相关配置。

本节内容是重点也是难点，初学的时候很抽象，特别是以前学过又没有学懂的读者，头脑里很多杂念，学起来更加费劲。建议清空头脑里的杂念，按照课程里面介绍的步骤一步步来学习，再多做实验加以练习。

3.4 RSTP 与 MSTP 原理与配置

STP 虽然能够解决环路问题，但是收敛速度慢，最长需要 50s 才能恢复业务，影响了用户通信质量。RSTP(Rapid Spanning-Tree Protocol，快速生成树协议)在 STP 基础上进行了改进，实现了网络拓扑快速收敛。

MSTP(Multi Spanning-Tree Protocol，多生成树协议)在 RSTP 的基础上进行了扩展，一个网络中同时存在多个生成树，可以优化网络的转发效率。

3.4.1 RSTP 原理

RSTP 可以实现快速收敛，收敛时间缩小到 3～5s，如图 3.68 所示。

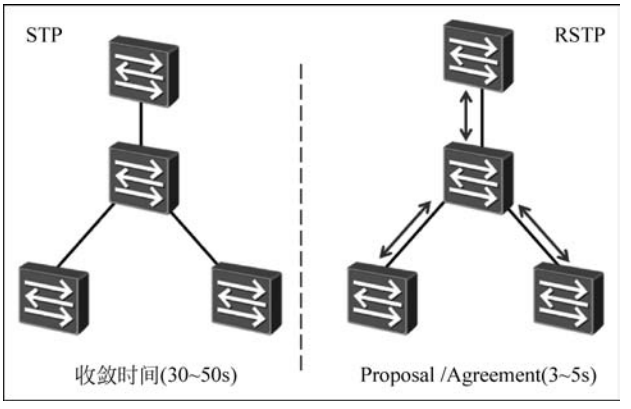


图 3.68 STP 与 RSTP 的差异

RSTP 主要是通过以下几个机制实现快速收敛：

① 备份端口；

② 边缘端口；

③ P/A 机制。

备份端口：如图 3.69 所示，SWC 的 G0/0/1 和 G0/0/2 同时连接到右边的 Hub，相当于一个物理环路。STP 中，SWC 的 G0/0/1 是 D 端口，G0/0/2 是 A 端口，如果 G0/0/1 发生故障，G0/0/2 恢复业务需要 50s，但实际上业务可以立刻恢复，因为右边网络不存在临时环路问题。

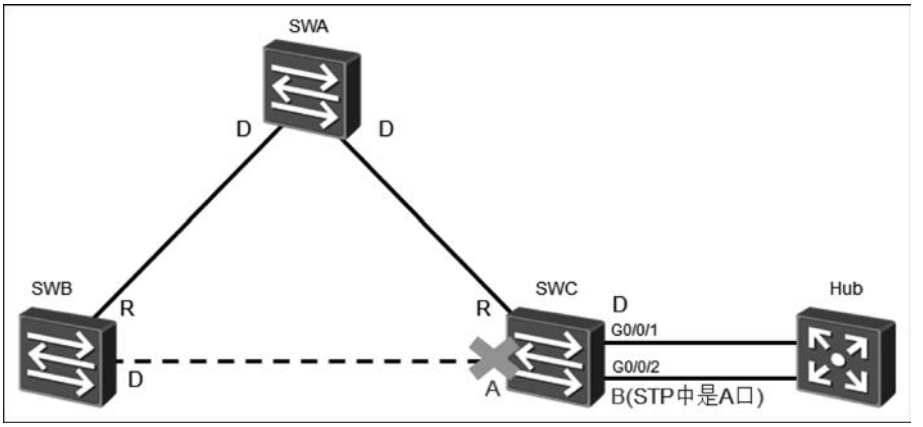


图 3.69　备份端口

在这个场景下，RSTP 会将 G0/0/2 置为 B 端口(Backup，备份端口)，如果 G0/0/1 发生故障，G0/0/2 可以马上恢复业务。

边缘端口：如图 3.70 所示，SWC 的 G0/0/1 口连接终端，没有必要参与 STP 计算，可以直接从 Disabled 变成 Forwarding 状态，无需再经过 Listening 和 Learning 状态。

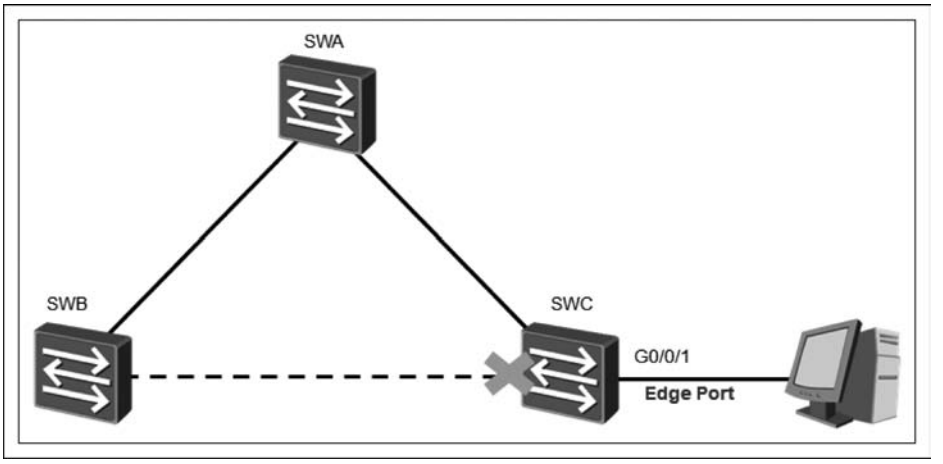


图 3.70　边缘端口

但是如果将计算机替换为交换机，在其向 SWC 的 G0/0/1 发送 RSTP 报文之后，该端口会重新变成普通端口，参与 RSTP 计算。

P/A 机制(Proposal/Agreement)：如图 3.71 所示，刚开始的时候 SWA、SWB、SWC 的优先级都是默认值 32 768，其中 SWA 的 MAC 地址最小，因此 SWA 是 Root。经过计算，SWC 的 G0/0/1 口是 A 端口，被阻塞。

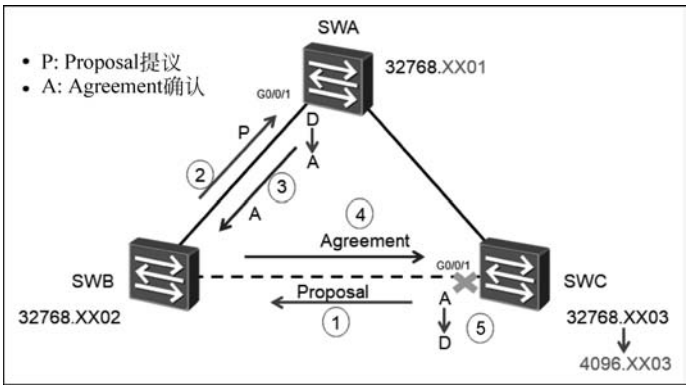


图 3.71 P/A 机制工作过程

后来手动修改了 SWC 的优先级，从 32 768 改成 4096，SWC 成为新的 Root，G0/0/1 口成为 D 端口。RSTP 中，SWC 通过 P/A 机制让 G0/0/1 快速进入转发状态，具体过程如下：

① SWC 给 SWB 发 Proposal，咨询 SWB，自己是否可以进入转发状态；

② SWB 不知道新的 A 端口在哪里，发送 Proposal 给 SWA，把球踢给 SWA；

③ SWA 的 G0/0/1 是新的 A 端口，可以确定网络不会存在环路，所以回 Agreement 给 SWB；

④ SWB 收到 SWA 的 Agreement 之后，也发 Agreement 给 SWC；

⑤ SWC 的 G0/0/1 收到 Agreement 之后，确定新的 A 端口已经存在，不会有环路，立刻进入转发状态，成为 D 端口。

这个过程看起来步骤很多，但实际上交换机很短时间内就可以完成。

除了上面 3 个优化点之外，RSTP 与 STP 还有 2 个差异点：

① Hello BPDU 发送差异；

② 拓扑变化处理差异。

Hello 报文发送差异：如图 3.72 所示，根桥稳定之后，每个交换机都会发 Hello BPDU，STP 只有根桥才能发 Hello BPDU。

拓扑变化差异：如图 3.73 所示，SWA 是 Root，SWC 的 G0/0/2 口最开始是 A 端口，后来 SWC 的 G0/0/1 口发生故障。

SWA 感知到 G0/0/1 口状态变化后，马上清除 G0/0/1 口相关的 MAC 地址表项。SWC 为了快速恢复业务，从 G0/0/2 口发出 Proposal，SWB 收到这个 Proposal 后也向 SWA 发 Proposal，同时将 Proposal 的出接口，也就是 SWB 的 G0/0/1 接口相关的 MAC 地址表项清除，防止业务报文发送到错误的端口。

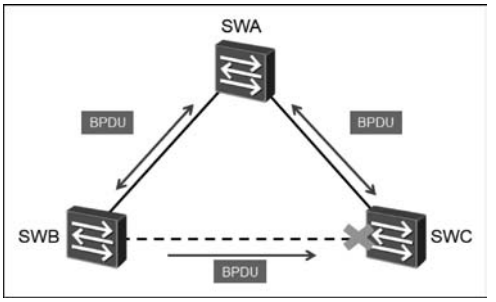


图 3.72　RSTP 的 Hello BPDU

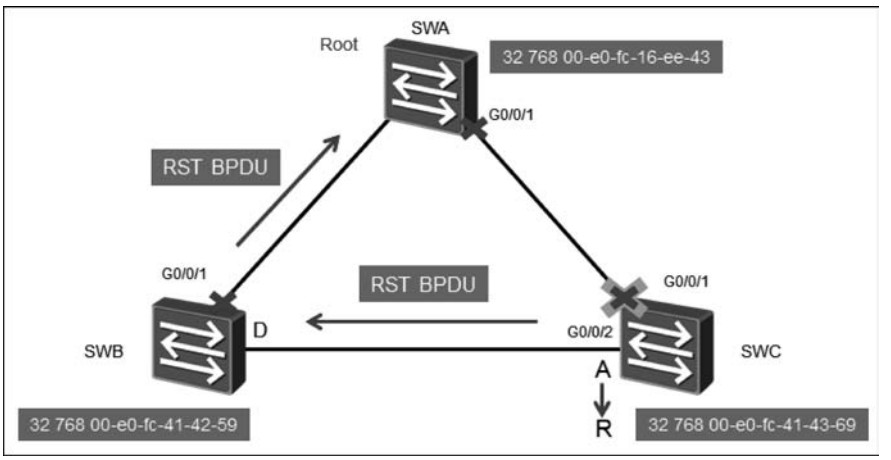


图 3.73　RSTP 拓扑变化处理机制

如果网络中有多台交换机，途经的每台交换机都会将发出 Proposal 的接口相关的 MAC 地址表项清除。大家可以自己比较一下 STP 与 RSTP 的差异。

关于 RSTP 与 STP 的兼容性，当同一个网段里既有工作在 STP 模式的交换机又有工作在 RSTP 模式的交换机时，STP 交换机会忽略接收到的 RSTP BPDU，而 RSTP 交换机在某端口上接收到 STP BPDU 时，会在等待两个 Hello Time 时间之后，把自己的端口切换到 STP 工作模式，此后便发送 STP BPDU，这样就实现了兼容性操作。全网同步成 STP 之后，收敛速度变慢，一般不这么用。

3.4.2　RSTP 配置

RSTP 配置与 STP 配置类似，不过多了几个特性，如图 3.74 所示。

SWC 的 G0/0/3 接口配置了边缘端口后，如果收到 BPDU 会变成普通端口。配置 BPDU 保护功能后，如果边缘端口收到 BPDU 报文，该端口将会被立即关闭，并通知网管系统。被关闭的边缘端口只能通过管理员手动恢复，以防止非法接入交换机，影响网络拓扑结构。

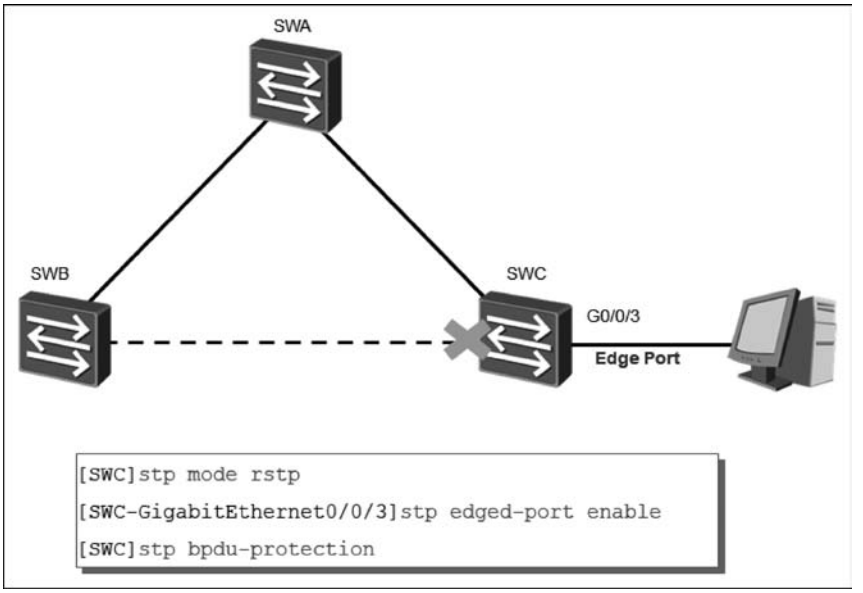


图 3.74　RSTP 配置

3.4.3　MSTP 工作原理

实际应用中，不同的业务可能会走不同的路径，如图 3.75 所示，VLAN 2 和 VLAN 3 的业务走不同路径，如果此时通过计算将 SWD 的 G0/0/1 口阻塞，那么 VLAN 3 的业务会受影响。为了解决这个问题，需要使用 MSTP(Multi Spanning Tree Protocol，多生成树协议)，在一个网络中同时存在多个生成树，可以以 VLAN 为单位，一个 VLAN 一个生成树，也可以多个 VLAN 共享一个生成树。

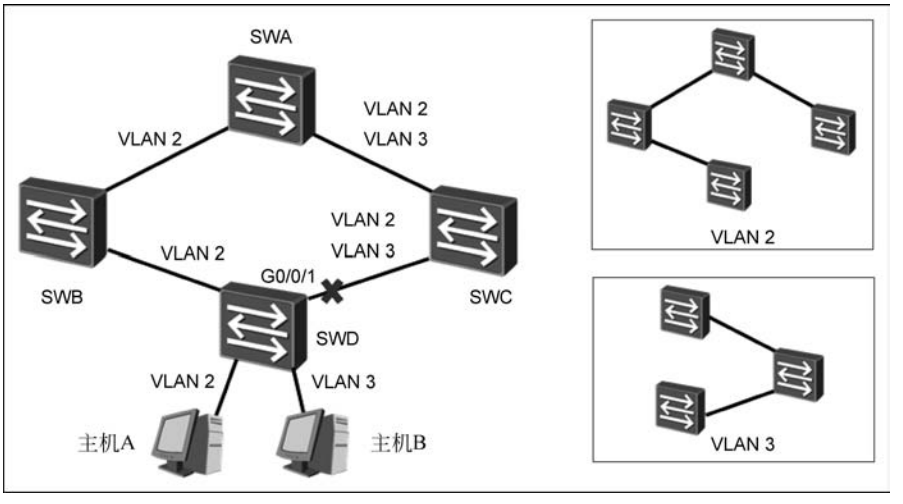


图 3.75　MSTP 工作原理

3.4.4　小结

本节主要介绍了 RSTP，另外简要介绍了 MSTP 的工作背景。RSTP 采用了多个快速收敛机制，其中最主要的是 P/A 机制，大大减少了业务恢复时间，从 STP 的 50s 减到 3～5s。MSTP 支持多个生成树，其收敛时间和 RSTP 一样，也是 3～5s。