生成树协议技术白皮书

目 录

1 概述

1.1 产生背景

在二层交换网络中，一旦存在环路就会造成报文在环路内不断循环和增生，产生广播风暴，从而占用所有的有效带宽，使网络变得不可用。

在这种环境下生成树协议应运而生，生成树协议是一种二层管理协议，它通过选择性地阻塞网络中的冗余链路来消除二层环路，同时还具备链路备份的功能。最初的生成树协议为 STP（Spanning Tree Protocol，生成树协议），之后又发展出 RSTP（Rapid Spanning Tree Protocol，快速生成树协议）、PVST（Per-VLAN Spanning Tree，每 VLAN 生成树）和 MSTP（Multiple Spanning Tree Protocol，多生成树协议）。

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。

1.2 技术优点

MSTP 由 IEEE 制定的 802.1s 标准定义，相比于 STP、RSTP 和 PVST MSTP 的优点如下：

- MSTP 把一个交换网络划分成多个域，每个域内形成多棵生成树，生成树之间彼此独立。生成树间独立计算，实现快速收敛。
- MSTP 通过设置 VLAN 与生成树的对应关系表（即 VLAN 映射表），将 VLAN 与生成树联系起来。并通过“实例”的概念，将多个 VLAN 捆绑到一个实例中，从而达到了节省通信开销和降低资源占用率的目的。
- MSTP 将环路网络修剪成为一个无环的树型网络，避免报文在环路网络中的增生和无限循环，同时还提供了数据转发的多个冗余路径，不同 VLAN 的流量沿各自的路径转发，实现 VLAN 数据的负载分担。
- MSTP 兼容 STP 和 RSTP，部分兼容 PVST。

2 STP 技术实现

2.1 概念介绍

1. 根桥

树形的网络结构必须有树根，于是 STP 引入了根桥的概念。根桥在全网中有且只有一个，其他设备则称为叶子节点。根桥会根据网络拓扑的变化而改变，因此根桥并不是固定的。

在网络初始化过程中，所有设备都视自己为根桥，生成各自的配置 BPDU 并周期性地向外发送；但当网络拓扑稳定以后，只有根桥才会向外发送配置 BPDU，其他设备则对其进行转发。

2. 根端口

非根桥设备上离根桥最近的端口。根端口负责与根桥进行通信。非根桥设备上有且只有一个根端口，根桥上没有根端口。

3. 指定桥与指定端口

有关指定桥与指定端口的含义，请参见表 1 的说明。

表1 指定桥与指定端口的含义

分类	指定桥	指定端口
对于一台设备而言	与本机直接相连并且负责向本机转发BPDU的设备	指定桥向本机转发BPDU的端口
对于一个局域网而言	负责向本网段转发BPDU的设备	指定桥向本网段转发BPDU的端口

4. 端口状态

STP 的端口有 5 种工作状态。如表 2 所示。

表2 STP 的端口状态

状态	描述
Disabled	该状态下的端口没有激活，不参与STP的任何动作，不转发用户流量
Listening	该状态下的端口可以接收和发送BPDU，但不转发用户流量
Learning	该状态下建立无环的转发表，不转发用户流量
Forwarding	该状态下的端口可以接收和发送BPDU，也转发用户流量
Blocking	该状态下的端口可以接收BPDU，但不转发用户流量

5. 路径开销

路径开销是 STP 协议用于选择链路的参考值。STP 协议通过计算路径开销，选择较为“强壮”的链路，阻塞多余的链路，将网络修剪成无环路的树型网络结构。

2.2 STP的协议报文

STP 采用的协议报文是 BPDU（Bridge Protocol Data Unit，网桥协议数据单元），也称为配置消息。本文中将把生成树的协议报文简称为 BPDU。

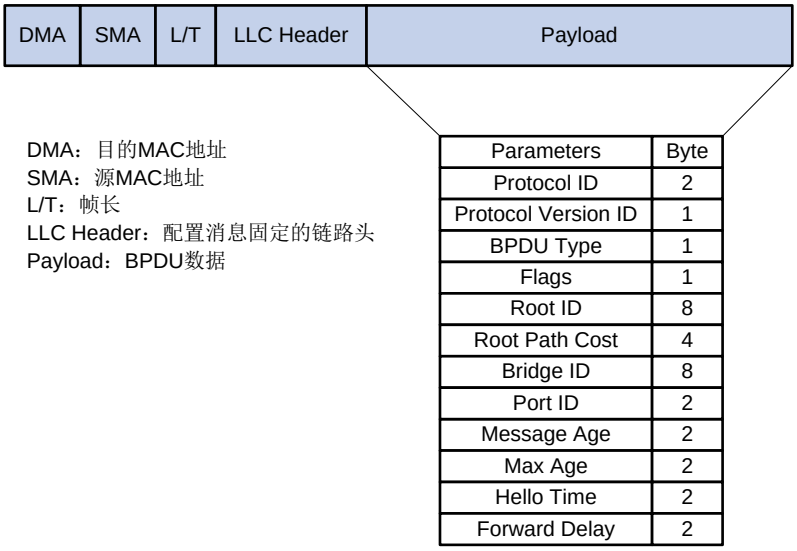
STP 通过在设备之间传递 BPDU 来确定网络的拓扑结构。BPDU 中包含了足够的信息来保证设备完成生成树的计算过程。STP 协议的 BPDU 分为以下两类：

- 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，拓扑变化通知 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。

1. 配置 BPDU

网桥之间通过交互配置 BPDU 来进行根桥的选举以及端口角色的确定。配置 BPDU 的格式如图 1 所示。

图1 配置 BPDU 格式



Parameters	Byte
Protocol ID	2
Protocol Version ID	1
BPDU Type	1
Flags	1
Root ID	8
Root Path Cost	4
Bridge ID	8
Port ID	2
Message Age	2
Max Age	2
Hello Time	2
Forward Delay	2

配置 BPDU 中 BPDU 数据的信息包括：

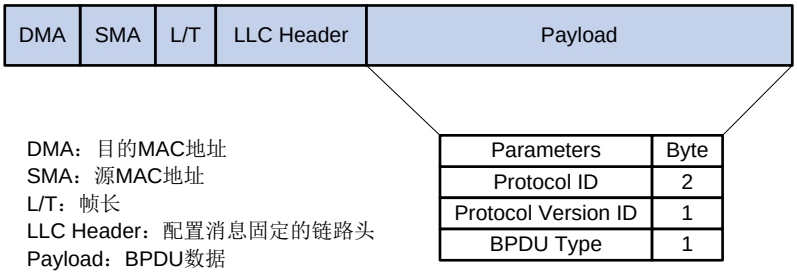
- 协议类型（Protocol ID）：固定为 0x0000，表示生成树协议。
- 协议版本号（Protocol Version ID）：目前生成树有三个版本，STP 的协议版本号为 0x00。
- BPDU 类型：配置 BPDU 类型为 0x00。
- BPDU Flags 位：BPDU 标志位，表示是哪种 BPDU。由 8 位组成，最低位（0 位）为 TC（Topology Change，拓扑改变）标志位；最高位（7 位）为 TCA（Topology Change Acknowledge，拓扑改变确认）标志位；其他 6 位保留。
- 根桥（Root Bridge）ID：由根桥的优先级和 MAC 地址组成。
- 根路径开销：到根桥的路径开销。
- 指定桥 ID：由指定桥的优先级和 MAC 地址组成。
- 指定端口 ID：由指定端口的优先级和该端口的全局编号组成。
- Message Age：BPDU 在网络中传播的生存期。
- Max Age：BPDU 在设备中的最大生存期。
- Hello Time：BPDU 的发送周期。
- Forward Delay：端口状态迁移的延迟时间。

其中通过根桥 ID、路径开销、指定桥 ID、指定端口 ID、Message Age、Max Age、Hello Time 和 Forward Delay 信息来保证设备完成生成树的计算过程。

2. TCN BPDU

如图 2 所示，TCN BPDU 和配置 BPDU 在结构上基本相同，也是由源/目的 MAC 地址、L/T 位、逻辑链路头和 BPDU 数据组成。但是 TCN BPDU 的 BPDU 数据组成非常简单，只包含三部分信息：协议类型、协议版本号和 BPDU 类型。协议类型和协议版本号字段和配置 BPDU 相同，BPDU 类型字段的值为 0x80，表示该 BPDU 为 TCN BPDU。

图2 TCN BPDU 格式



TCN BPDU 有两个产生条件：

- 网桥上有端口转变为 Forwarding 状态，且该网桥至少包含一个指定端口。
- 网桥上有端口从 Forwarding 状态或 Learning 状态转变为 Blocking 状态。

当上述两个条件之一满足时，说明网络拓扑发生了变化，网桥需要使用 TCN BPDU 通知根桥。根桥可以通过将配置 BPDU 中对应标志位置位来通知所有网桥网络拓扑发生了变化，需要使用较短的 MAC 地址老化时间，保证拓扑的快速收敛。

2.3 STP的拓扑计算过程

STP 的拓扑计算过程如下：设备通过比较不同端口收到的 BPDU 报文的优先级高低，选举出根桥、根端口、指定端口，完成生成树的计算，建立对应的树形拓扑。

1. 初始状态

各设备的各端口在初始时会生成以本设备为根桥的 BPDU，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。

2. 选择根桥

网络初始化时，需要在网络中所有的 STP 设备中选择一个根桥，根桥的选择方式有以下两种：

- 自动选举：网络初始化时，网络中所有的 STP 设备都认为自己是“根桥”，根桥 ID 为自身的设备 ID。通过交换 BPDU，设备之间比较根桥 ID，网络中根桥 ID 最小的设备被选为根桥。
- 手工指定：用户手工将设备配置为指定生成树的根桥或备份根桥。
  - 在一棵生成树中，生效的根桥只有一个，当两台或两台以上的设备被指定为同一棵生成树的根桥时，系统将选择 MAC 地址最小的设备作为根桥。
  - 用户可以在每棵生成树中指定一个或多个备份根桥。当根桥出现故障或被关机时，如果配置了一个备份根桥，则该备份根桥可以取代根桥成为指定生成树的根桥；如果配置了多个备份根桥，则 MAC 地址最小的备份根桥将成为指定生成树的根桥。但此时若配置了新的根桥，则备份根桥将不会成为根桥。

3. 选择根端口和指定端口

根端口和指定端口的选择过程如表 3 所示。

表3 根端口和指定端口的选择过程

步骤	内容
1	非根桥设备将接收最优BPDU（最优BPDU的选择过程如表4所示）的那个端口定为根端口

步骤	内容
2	设备根据根端口的BPDU和根端口的路径开销，为每个端口计算一个指定端口BPDU： • 根桥 ID 替换为根端口的 BPDU 的根桥 ID； • 根路径开销替换为根端口 BPDU 的根路径开销加上根端口对应的路径开销； • 指定桥 ID 替换为自身设备的 ID； • 指定端口 ID 替换为自身端口 ID。
3	设备将计算出的BPDU与角色待定端口自己的BPDU进行比较： • 如果计算出的 BPDU 更优，则该端口被确定为指定端口，其 BPDU 也被计算出的 BPDU 替换，并周期性地向外发送； • 如果该端口自己的 BPDU 更优，则不更新该端口的 BPDU 并将该端口阻塞。该端口将不再转发数据，且只接收不发送 BPDU。

说明

当拓扑处于稳定状态时，只有根端口和指定端口在转发用户流量。其他端口都处于阻塞状态，只接收 STP 协议报文而不转发用户流量。

表4 最优 BPDU 的选择过程

步骤	内容
1	每个端口将收到的BPDU与自己的BPDU进行比较： • 如果收到的 BPDU 优先级较低，则将其直接丢弃，对自己的 BPDU 不进行任何处理； • 如果收到的 BPDU 优先级较高，则用该 BPDU 的内容将自己 BPDU 的内容替换掉。
2	设备将所有端口的BPDU进行比较，选出最优的BPDU

说明

BPDU 优先级的比较规则如下：

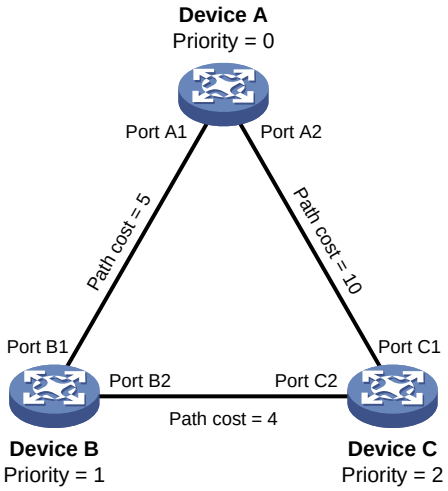
- 根桥 ID 较小的 BPDU 优先级较高；
- 若根桥 ID 相同，则比较根路径开销：将 BPDU 中的根路径开销与本端口对应的路径开销相加，二者之和较小的 BPDU 优先级较高；
- 若根路径开销也相同，则依次比较指定桥 ID、指定端口 ID、接收该 BPDU 的端口 ID 等，上述值较小的 BPDU 优先级较高。

一旦根桥、根端口和指定端口选举成功，整个树形拓扑就建立完毕了。

2.4 STP算法实现举例

下面结合例子说明 STP 算法实现的具体过程。

图3 STP 算法实现过程组网图



如图 3 所示，Device A、Device B 和 Device C 的优先级分别为 0、1 和 2，Device A 与 Device B 之间、Device A 与 Device C 之间以及 Device B 与 Device C 之间链路的路径开销分别为 5、10 和 4。

1. 各设备的初始状态

各设备的初始状态如表 5 所示。

表5 各设备的初始状态

设备	端口名称	端口的 BPDU
Device A	Port A1	{0，0，0，Port A1}
	Port A2	{0，0，0，Port A2}
Device B	Port B1	{1，0，1，Port B1}
	Port B2	{1，0，1，Port B2}
Device C	Port C1	{2，0，2，Port C1}
	Port C2	{2，0，2，Port C2}

说明

表 5 中 BPDU 各项的具体含义为：{根桥 ID，根路径开销，指定桥 ID，指定端口 ID}。

2. 各设备的比较过程及结果

各设备的比较过程及结果如表 6 所示。

表6 各设备的比较过程及结果

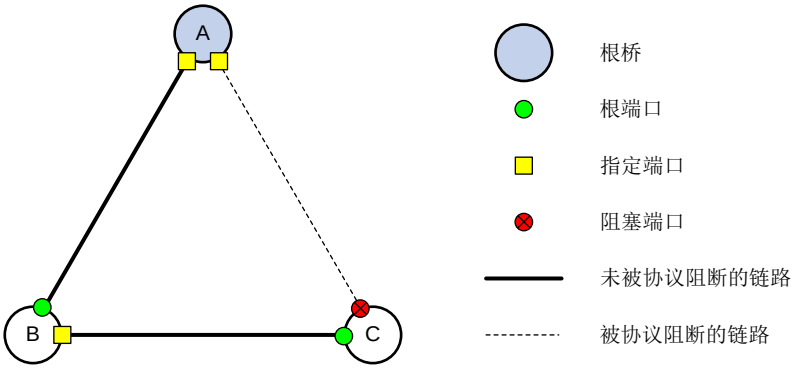
设备	比较过程	比较后端口的 BPDU
Device A	• Port A1 收到 Port B1 的 BPDU {1，0，1，Port B1}，发现自己的 BPDU {0，0，0，Port A1}更优，于是将其丢弃。	• Port A1：{0，0，0，Port A1}

设备	比较过程	比较后端口的 BPDU
	• Port A2 收到 Port C1 的 BPDU {2，0，2，Port C1}，发现自己的 BPDU {0，0，0，Port A2}更优，于是将其丢弃。 • Device A 发现自己各端口的 BPDU 中的根桥和指定桥都是自己，于是认为自己就是根桥，各端口的 BPDU 都不作任何修改，此后便周期性地向外发送 BPDU。	• Port A2： {0，0，0，Port A2}
Device B	• Port B1 收到 Port A1 的 BPDU {0，0，0，Port A1}，发现其比自己的 BPDU {1，0，1，Port B1}更优，于是更新自己的 BPDU。 • Port B2 收到 Port C2 的 BPDU {2，0，2，Port C2}，发现自己的 BPDU {1，0，1，Port B2}更优，于是将其丢弃。	• Port B1： {0，0，0，Port A1} • Port B2： {1，0，1，Port B2}
	• Device B 比较自己各端口的 BPDU，发现 Port B1 的 BPDU 最优，于是该端口被确定为根端口，其 BPDU 不变。 • Device B 根据根端口的 BPDU 和路径开销，为 Port B2 计算出指定端口的 BPDU {0，5，1，Port B2}，然后与 Port B2 本身的 BPDU {1，0，1，Port B2}进行比较，发现计算出的 BPDU 更优，于是 Port B2 被确定为指定端口，其 BPDU 也被替换为计算出的 BPDU，并周期性地向外发送。	• 根端口 Port B1： {0，0，0，Port A1} • 指定端口 Port B2： {0，5，1，Port B2}
Device C	• Port C1 收到 Port A2 的 BPDU {0，0，0，Port A2}，发现其比自己的 BPDU {2，0，2，Port C1}更优，于是更新自己的 BPDU。 • Port C2 收到 Port B2 更新前的 BPDU {1，0，1，Port B2}，发现其比自己的 BPDU {2，0，2，Port C2}更优，于是更新自己的 BPDU。	• Port C1： {0，0，0，Port A2} • Port C2： {1，0，1，Port B2}
	• Device C 比较自己各端口的 BPDU，发现 Port C1 的 BPDU 最优，于是该端口被确定为根端口，其 BPDU 不变。 • Device C 根据根端口的 BPDU 和路径开销，为 Port C2 计算出指定端口的 BPDU {0，10，2，Port C2}，然后与 Port C2 本身的 BPDU {1，0，1，Port B2}进行比较，发现计算出的 BPDU 更优，于是 Port C2 被确定为指定端口，其 BPDU 也被替换为计算出的 BPDU。	• 根端口 Port C1： {0，0，0，Port A2} • 指定端口 Port C2： {0，10，2，Port C2}
	• Port C2 收到 Port B2 更新后的 BPDU {0，5，1，Port B2}，发现其比自己的 BPDU {0，10，2，Port C2}更优，于是更新自己的 BPDU。 • Port C1 收到 Port A2 周期性发来的 BPDU {0，0，0，Port A2}，发现其与自己的 BPDU 一样，于是将其丢弃。	• Port C1： {0，0，0，Port A2} • Port C2： {0，5，1，Port B2}
	• Device C 比较 Port C1 的根路径开销 10（收到的 BPDU 中的根路径开销 0＋本端口所在链路的路径开销 10）与 Port C2 的根路径开销 9（收到的 BPDU 中的根路径开销 5＋本端口所在链路的路径开销 4），发现后者更小，因此 Port C2 的 BPDU 更优，于是 Port C2 被确定为根端口，其 BPDU 不变。 • Device C 根据根端口的 BPDU 和路径开销，为 Port C1 计算出指定端口的 BPDU {0，9，2，Port C1}，然后与 Port C1 本身的 BPDU {0，0，0，Port A2}进行比较，发现本身的 BPDU 更优，于是 Port C1 被阻塞，其 BPDU 不变。从此，Port C1 不再转发数据，直至有触发生成树计算的新情况出现，譬如 Device B 与 Device C 之间的链路 down 掉。	• 阻塞端口 Port C1： {0，0，0，Port A2} • 根端口 Port C2： {0，5，1，Port B2}

3. 计算出的生成树

经过上述比较过程之后，以 Device A 为根桥的生成树就确定下来了，其拓扑如图 4 所示。

图4 计算后得到的拓扑



说明

为了便于描述，本例简化了生成树的计算过程，实际的过程要更加复杂。

2.5 STP的BPDU传递机制

STP 的 BPDU 传递机制如下：

- 当网络初始化时，所有的设备都将自己作为根桥，生成以自己为根的 BPDU，并以 Hello Time 为周期定时向外发送。
- 接收到 BPDU 的端口如果是根端口，且接收的 BPDU 比该端口的 BPDU 优，则设备将 BPDU 中携带的 Message Age 按照一定的原则递增，并启动定时器为这条 BPDU 计时，同时将此 BPDU 从设备的指定端口转发出去。
- 如果指定端口收到的 BPDU 比本端口的 BPDU 优先级低时，会立刻发出自己的更好的 BPDU 进行回应。
- 如果某条路径发生故障，则这条路径上的根端口不会再收到新的 BPDU，旧的 BPDU 将会因为超时而被丢弃，设备重新生成以自己为根的 BPDU 并向外发送，从而引发生成树的重新计算，得到一条新的通路替代发生故障的链路，恢复网络连通性。

不过，重新计算得到的新 BPDU 不会立刻就传遍整个网络，因此旧的根端口和指定端口由于没有发现网络拓扑变化，将仍按原来的路径继续转发数据。如果新选出的根端口和指定端口立刻就开始数据转发的话，可能会造成暂时性的环路。

2.6 STP的时间参数

在 STP 的计算过程中，用到了以下三个重要的时间参数：

- Forward Delay：用于确定状态迁移的延迟时间。缺省情况下 Forward Delay 时间为 15 秒。链路故障会引发网络重新进行生成树的计算，生成树的结构将发生相应的变化。不过重新计算得到的新 BPDU 无法立刻传遍整个网络，如果新选出的根端口和指定端口立刻就开始数据转发的话，可能会造成暂时性的环路。为此，生成树协议在端口由 Blocking 状态向 Forwarding 状态迁移的过程中设置了 Listening 和 Learning 状态作为过渡（Listening 和 Learning 状态都

会持续 Forward Delay 时间），并规定状态迁移需要等待 Forward Delay 时间，以保持与远端的设备状态切换同步。新选出的根端口和指定端口要经过 2 倍的 Forward Delay 延时后才能进入转发状态，这个延时保证了新的 BPDU 已经传遍整个网络。

- Hello Time：用于设备检测链路是否存在故障。缺省情况下 Hello Time 为 2 秒。生成树协议每隔 Hello Time 时间会发送 BPDU，以确认链路是否存在故障。如果设备在超时时间（超时时间＝超时时间因子×3×Hello Time）内没有收到 BPDU，则会由于消息超时而重新计算生成树。
- Max Age：用于判断 BPDU 在设备内的保存时间是否“过时”，设备会将过时的 BPDU 丢弃。缺省情况下 Max Age 时间为 20 秒。在 MSTP 的 CIST 上，设备根据 Max Age 时间来确定端口收到的 BPDU 是否超时。如果端口收到的 BPDU 超时，则需要对该 MSTI 重新计算。Max Age 时间对 MSTP 的 MSTI 无效。

STP 每隔一个 Hello Time 发送一个 BPDU，并且引入 Keepalive 机制。Hello 包的发送可以避免最大失效定时器溢出。如果最大失效定时器溢出，通常表明有连接错误发生。此时，STP 会进入 Listening 状态。STP 要从连接错误中恢复过来，一般需要 50 秒的时间。其中 BPDU 最长的失效时间 20 秒；Listening 状态持续 15 秒；Learning 状态持续 15 秒。

为保证网络拓扑的快速收敛，需要配置合适的时间参数。上述三个时间参数之间应满足以下关系，否则会引起网络的频繁震荡：

- 2×（Forward Delay－1 秒）≥Max Age
- Max Age≥2×（Hello Time＋1 秒）

3 RSTP 技术实现

3.1 概念介绍

1. 端口角色

RSTP 中根端口和指定端口角色的定义和 STP 相同。与 STP 相比，RSTP 增加了三种端口角色替换端口（Alternate Port）、备份端口（Backup Port）和边缘端口（Edge Port）。

- 替换端口为网桥提供一条到达根桥的备用路径，当根端口或主端口被阻塞后，替换端口将成为新的根端口或主端口。
- 备份端口为网桥提供了到达同一个物理网段的冗余路径，当指定端口失效后，备份端口将转换为新的指定端口。当开启了生成树协议的同一台设备上的两个端口互相连接而形成环路时，设备会将其中一个端口阻塞，该端口就是备份端口。
- 边缘端口是不与其他设备或网段连接的端口，边缘端口一般与用户终端设备直接相连。

2. 端口状态

RSTP 将端口状态缩减为三个，分别为 Discarding、Learning 和 Forwarding 状态。STP 中的 Disabled、Blocking 和 Listening 状态在 RSTP 中都对应为 Discarding 状态，如表 7 所示。

表7 RSTP 的端口状态

STP 端口状态	RSTP 端口状态	是否发送 BPDU	是否进行 MAC 地址学习	是否收发用户流量
Disabled	Discarding	否	否	否

STP 端口状态	RSTP 端口状态	是否发送 BPDU	是否进行 MAC 地址学习	是否收发用户流量
Blocking	Discarding	否	否	否
Listening	Discarding	是	否	否
Learning	Learning	是	是	否
Forwarding	Forwarding	是	是	是

3.2 RSTP的协议报文

RSTP 也是通过在设备之间传递 BPDU 来确定网络的拓扑结构。RSTP 的 BPDU 格式和 STP 的配置 BPDU 格式非常相似，仅在以下几个信息有所不同：

- BPDU 类型变为 0x02，表示为 RSTP 的 BPDU。
- BPDU 协议版本号为 0x02，表示为 RSTP 协议。
- Flags 位字段使用了全 8 位。
- RSTP 在 BPDU 报文的最后增加了 Version1 Length 字段。该字段的值为 0x00，表示本 BPDU 中不包含 Version 1 内容。

在拓扑改变时，RSTP 的拓扑改变处理过程不再使用 TCN BPDU，而使用 Flags 位中 TC 置位的 RST BPDU 取代 TCN BPDU，并通过泛洪方式快速的通知到整个网络。

3.3 RSTP的工作原理

进行 RSTP 计算时，端口会在 Discarding 状态完成角色的确定，当端口确定为根端口和指定端口后，经过 Forward Delay 端口会进入 Learning 状态；当端口确定为替换端口，端口会维持在 Discarding 状态。

处于 Learning 状态的端口其处理方式和 STP 相同，开始学习 MAC 地址并在 Forward Delay 后进入 Forwarding 状态开始收发用户流量。

在 RSTP 中，根端口 的端口状态快速迁移的条件是：本设备上旧的根端口已经停止转发数据，而且上游指定端口已经开始转发数据。

在 RSTP 中，指定端口的端口状态快速迁移的条件是：指定端口是边缘端口（即该端口直接与用户终端相连，而没有连接到其他设备或共享网段上）或者指定端口与点对点链路（即两台设备直接相连的链路）相连。如果指定端口是边缘端口，则指定端口可以直接进入转发状态；如果指定端口连接着点对点链路，则设备可以通过与下游设备握手，得到响应后即刻进入转发状态。

3.4 RSTP中的BPDU处理

相比于 STP，RSTP 对 BPDU 的发送方式做了改进，RSTP 中网桥可以自行从指定端口发送 RST BPDU，不需要等待来自根桥的 RST BPDU，BPDU 的发送周期为 Hello Time。

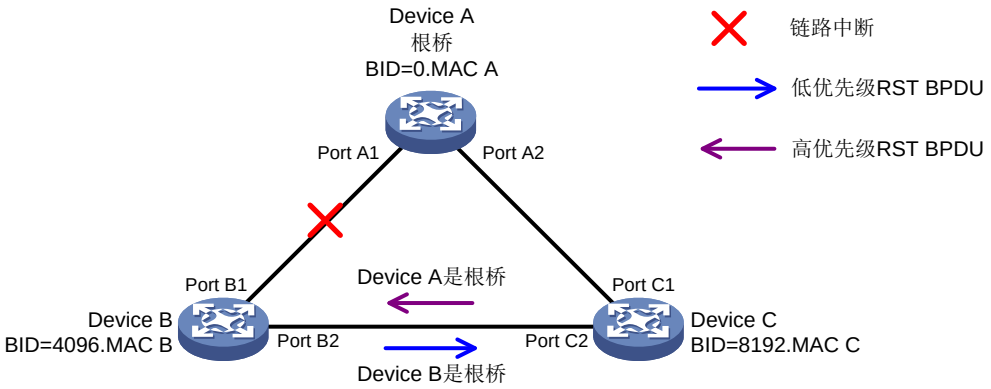
由于 RSTP 中网桥可以自行从指定端口发送 RST BPDU，所以在网桥之间可以提供一种保活机制，即在一定时间内网桥没有收到对端网桥发送的 RST BPDU，即可认为和对端网桥的连接中断。

RSTP 规定，若在三个连续的 Hello Time 时间内网桥没有收到对端指定桥发送的 RST BPDU，则网桥端口保存的 RST BPDU 老化，认为与对端网桥连接中断。新的老化机制大大加快了拓扑变化的感知，从而可以实现快速收敛。

在 RSTP 中，如果阻塞状态的端口收到低优先级的 RST BPDU，也可以立即对其做出回应。

如图 5，网络中 Device A 为根桥，Device C 阻塞和 Device B 相连的端口。当 Device B 和根桥之间的链路中断时，Device B 会发送以自己为根桥的 RST BPDU。Device C 收到 Device B 发送的 RST BPDU 后，经过比较，Device B 的值 RST BPDU 为低优先级的 RST BPDU，所以 Device C 的端口会立即对该 RST BPDU 做出回应，发送优先级更高的 RST BPDU。Device B 收到 Device C 发送的 RST BPDU 后，将会停止发送 RST BPDU，并将和 Device C 连接的端口确定为根端口。

图5 RSTP 对低优先级 RST BPDU 的处理



4 PVST 技术实现

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 RSTP 协议，不同 VLAN 之间的生成树完全独立。

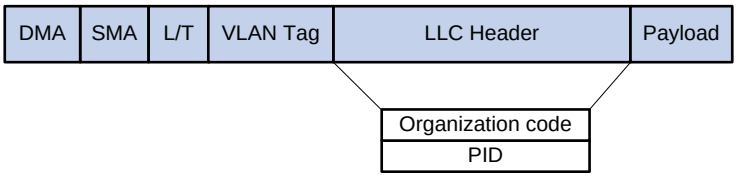
运行 PVST 的 H3C 设备可以与运行 Rapid PVST 或 PVST 的第三方设备互通。当运行 PVST 的 H3C 设备之间互联，或运行 PVST 的 H3C 设备与运行 Rapid PVST 的第三方设备互通时，H3C 设备支持像 RSTP 一样的快速收敛。

4.1 PVST的协议报文

如图 6，从报文结构对上看，PVST 的 BPDU 和 RSTP 的 BPDU 不同在于以下几点：

- 报文的目的 MAC 地址改变，变为私有 MAC 地址 01-00-0c-cc-cc-cd。
- 报文携带 VLAN 标签，确定该协议报文归属的 VLAN。
- 报文配置消息固定链路头字段添加 Organization code 和 PID 字段。

图6 PVST 报文格式



根据端口类型的不同，PVST 所发送的 BPDU 格式也有所差别：

- 对于 Access 端口，PVST 将根据该 VLAN 的状态发送 RSTP 格式的 BPDU。
- 对于 Trunk 端口和 Hybrid 端口，PVST 将在缺省 VLAN 内根据该 VLAN 的状态发送 RSTP 格式的 BPDU，而对于其他本端口允许通过的 VLAN，则发送 PVST 格式的 BPDU。

4.2 PVST的工作原理

PVST 借助 MSTP 的实例和 VLAN 映射关系模型，将 MSTP 每个实例映射一个 VLAN。PVST 中每个 VLAN 独立运行 RSTP，独立运算，并允许以每个 VLAN 为基础开启或关闭生成树。每个 VLAN 内的生成树实例都有单独的网络拓扑结构，相互之间没有影响。这样既可以消除了 VLAN 内的冗余环路，还可以实现不同 VLAN 间负载分担。

PVST 在缺省 VLAN 上通过 RSTP 报文进行拓扑运算；在其他 VLAN 上通过带 VLAN Tag 的 PVST 报文进行拓扑运算。

PVST 的端口角色和端口状态和 RSTP 相同，能够实现快速收敛。

5 MSTP 技术实现

5.1 概念介绍

图7 MSTP 的基本概念示意图

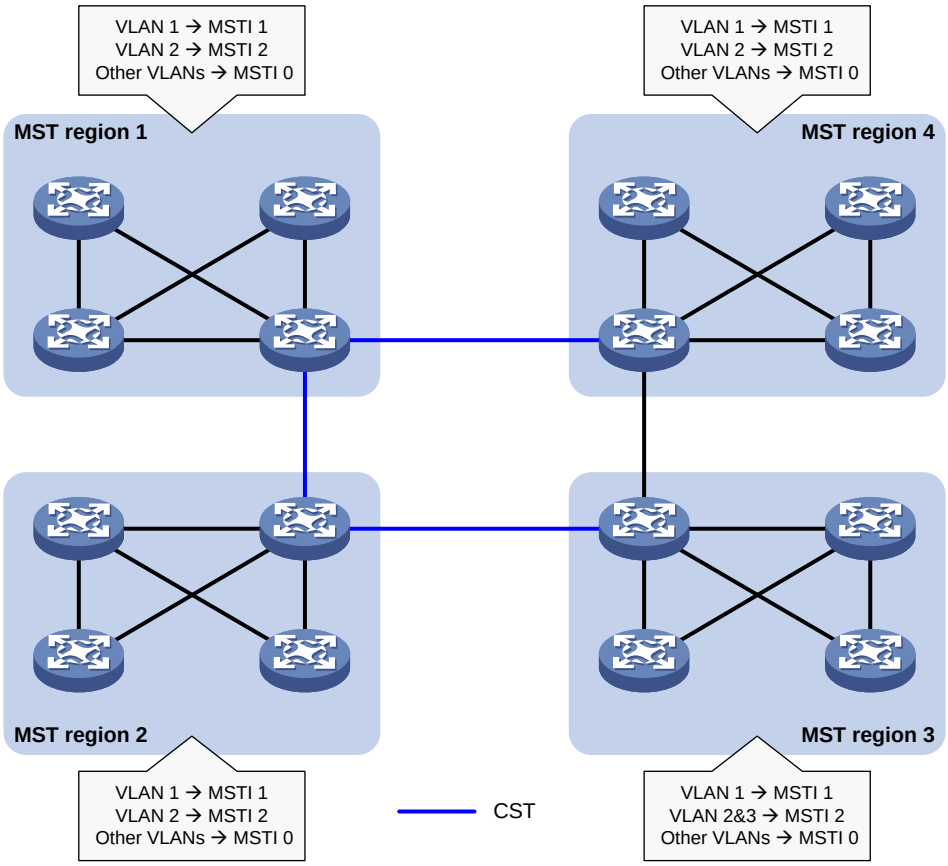
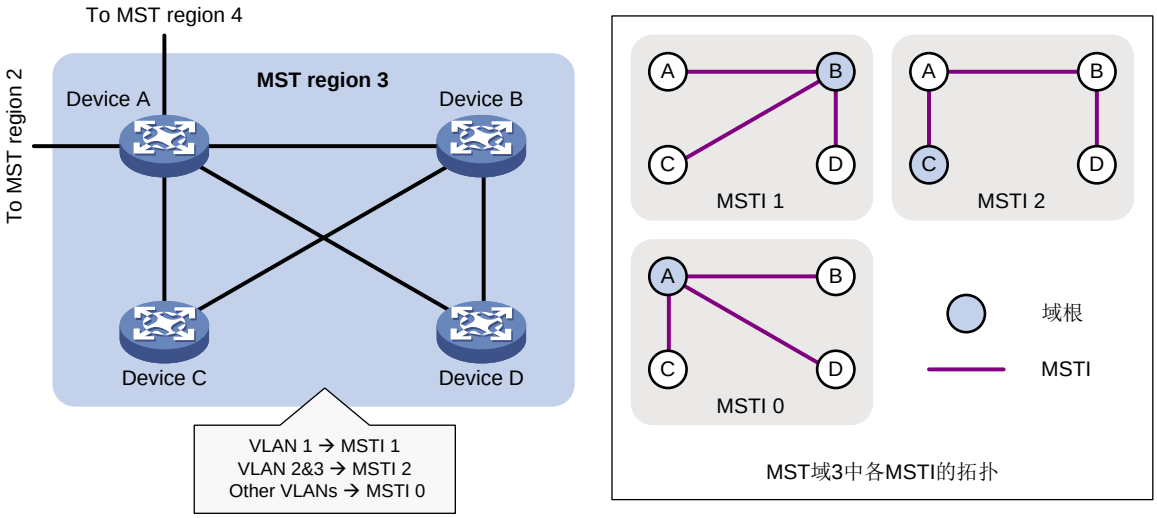


图8 MST 域 3 详图



1. MST 域

MST 域（Multiple Spanning Tree Regions，多生成树域）是由交换网络中的多台设备以及它们之间的网段所构成。这些设备具有下列特点：

- 都开启了生成树协议。
- 域名相同。
- VLAN 与 MSTI 间映射关系相同。
- MSTP 修订级别相同。
- 这些设备之间有物理链路连通。

一个交换网络中可以存在多个 MST 域，用户可以将多台设备划分在一个 MST 域内。如在图 7 所示的网络中就有 MST 域 1～MST 域 4 这四个 MST 域，每个域内的所有设备都具有相同的 MST 域配置。

2. MSTI

一个 MST 域内可以通过 MSTP 生成多棵生成树，各生成树之间彼此独立并分别与相应的 VLAN 对应，每棵生成树都称为一个 MSTI（Multiple Spanning Tree Instance，多生成树实例）。如在图 8 所示的 MST 域 3 中，包含有三个 MSTI：MSTI 1、MSTI 2 和 MSTI 0。

3. VLAN 映射表

VLAN 映射表是 MST 域的一个属性，用来描述 VLAN 与 MSTI 间的映射关系。如图 8 中 MST 域 3 的 VLAN 映射表就是：VLAN 1 映射到 MSTI 1，VLAN 2 和 VLAN 3 映射到 MSTI 2，其余 VLAN 映射到 MSTI 0。MSTP 就是根据 VLAN 映射表来实现负载分担的。

4. CST

CST（Common Spanning Tree，公共生成树）是一棵连接交换网络中所有 MST 域的单生成树。如果把每个 MST 域都看作一台“设备”，CST 就是这些“设备”通过 STP 协议、RSTP 协议计算生成的一棵生成树。如图 7 中的蓝色线条描绘的就是 CST。

5. IST

IST（Internal Spanning Tree，内部生成树）是 MST 域内的一棵生成树，它是一个特殊的 MSTI，通常也称为 MSTI 0，所有 VLAN 缺省都映射到 MSTI 0 上。如图 8 中的 MSTI 0 就是 MST 域 3 内的 IST。

6. CIST

CIST（Common and Internal Spanning Tree，公共和内部生成树）是一棵连接交换网络内所有设备的单生成树，所有 MST 域的 IST 再加上 CST 就共同构成了整个交换网络的一棵完整的单生成树，即 CIST。如图 7 中各 MST 域内的 IST（即 MSTI 0）再加上 MST 域间的 CST 就构成了整个网络的 CIST。

7. 域根

域根（Regional Root）就是 MST 域内 IST 或 MSTI 的根桥。MST 域内各生成树的拓扑不同，域根也可能不同。如在图 8 所示的 MST 域 3 中，MSTI 1 的域根为 Device B，MSTI 2 的域根为 Device C，而 MSTI 0（即 IST）的域根则为 Device A。

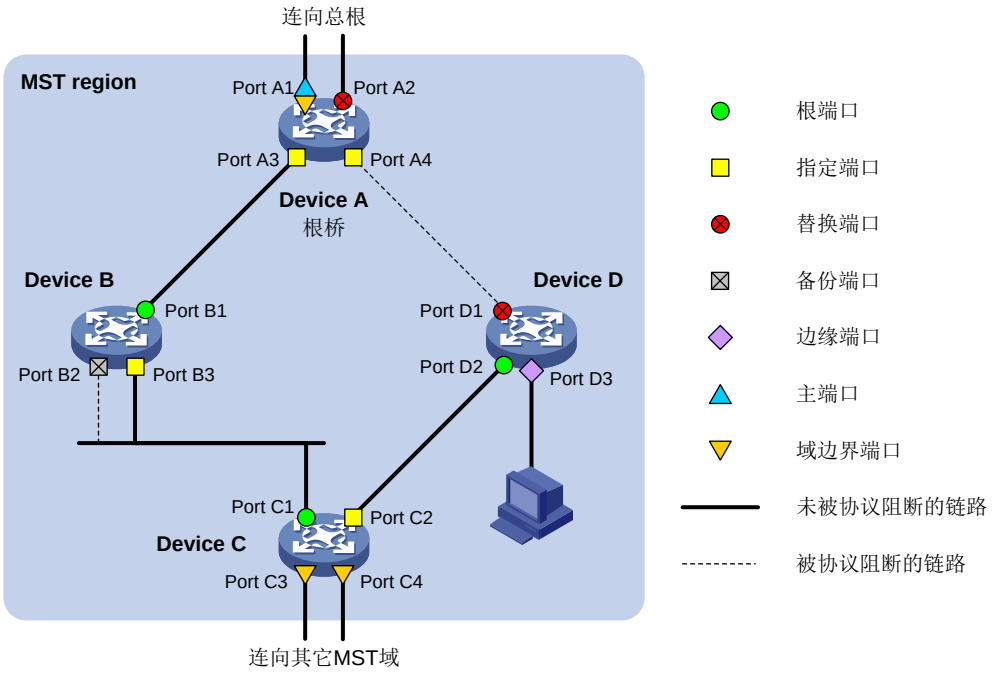
8. 总根

总根（Common Root Bridge）就是 CIST 的根桥。如图 7 中 CIST 的总根就是 MST 域 1 中的某台设备。

9. 端口角色

端口在不同的 MSTI 中可以担任不同的角色。如图 9 所示，在由 Device A、Device B、Device C 和 Device D 共同构成的 MST 域中，Device A 的端口 Port A1 和 Port A2 连向总根方向，Device B 的端口 Port B2 和 Port B3 相连而构成环路，Device C 的端口 Port C3 和 Port C4 连向其他 MST 域，Device D 的端口 Port D3 直接连接用户主机。

图9 端口角色示意图



如图 9 所示，MSTP 计算过程中涉及到的主要端口角色有以下几种：

- 根端口（Root Port）：在非根桥上负责向根桥方向转发数据的端口就称为根端口，根桥上没有根端口。
- 指定端口（Designated Port）：负责向下游网段或设备转发数据的端口就称为指定端口。
- 替换端口（Alternate Port）：是根端口和主端口的备份端口。当根端口或主端口被阻塞后，替换端口将成为新的根端口或主端口。
- 备份端口（Backup Port）：是指定端口的备份端口。当指定端口失效后，备份端口将转换为新的指定端口。当开启了生成树协议的同一台设备上的两个端口互相连接而形成环路时，设备会将其中一个端口阻塞，该端口就是备份端口。
- 边缘端口（Edge Port）：不与其他设备或网段连接的端口就称为边缘端口，边缘端口一般与用户终端设备直接相连。

- 主端口（Master Port）：是将 MST 域连接到总根的端口（主端口不一定在域根上），位于整个域到总根的最短路径上。主端口是 MST 域中的报文去往总根的必经之路。主端口在 IST/CIST 上的角色是根端口，而在其他 MSTI 上的角色则是主端口。
- 域边界端口（Boundary Port）：是位于 MST 域的边缘、并连接其他 MST 域或 MST 域与运行 STP/RSTP 的区域的端口。主端口同时也是域边界端口。在进行 MSTP 计算时，域边界端口在 MSTI 上的角色与 CIST 的角色一致，但主端口除外——主端口在 CIST 上的角色为根端口，在其他 MSTI 上的角色才是主端口。

10. 端口状态

MSTP 中的端口状态可分为三种，如表 8 所示。同一端口在不同的 MSTI 中的端口状态可以不同。

表8 MSTP 的端口状态

状态	描述
Forwarding	该状态下的端口可以接收和发送BPDU，也转发用户流量
Learning	是一种过渡状态，该状态下的端口可以接收和发送BPDU，但不转发用户流量
Discarding	该状态下的端口可以接收和发送BPDU，但不转发用户流量

端口状态和端口角色是没有必然联系的，表 9 给出了各种端口角色能够具有的端口状态（“√”表示此端口角色能够具有此端口状态；“-”表示此端口角色不能具有此端口状态）。

表9 各种端口角色具有的端口状态

端口角色（右） 端口状态（下）	根端口/主端口	指定端口	替换端口	备份端口
Forwarding	√	√	-	-
Learning	√	√	-	-
Discarding	√	√	√	√

5.2 MSTP的协议报文

如图 10，MST BPDU 和 RST BPDU 的前 36 个字节格式是相同的，其中 BPDU 协议版本号为 0x03，表示 MSTP 协议，BPDU 类型为 0x02，表示为 RST/MST BPDU。

图10 MSTP 的 BPDU 格式

Parameters	Byte
Protocol ID	2
Protocol Version ID	1
BPDU Type	1
Flags	1
Root ID	8
Root Path Cost	4
Bridge ID	8
Port ID	2
Message Age	2
Max Age	2
Hello Time	2
Forward Delay	2
Version1 Length=0	1
Version3 Length	2
MST Configuration ID	51
CIST IRPC	4
CIST Bridge ID	8
CIST Remaining ID	1
MSTI Configuration Messages	LEN

MSTP专有字段

RST BPDU 中的 Root ID 字段在 MSTP 中表示 CIST（Common and Internal Spanning Tree，公共和内部生成树）总根 ID，Root Path Cost 字段在 MSTP 中表示 CIST 外部路径开销（External Path Cost，EPC），Bridge ID 字段在 MSTP 中表示 CIST 域根 ID，Port ID 字段在 MSTP 中表示 CIST 指定端口 ID。

从第 37 字节开始是 MSTP 的专有字段：

- Version3 Length：表示 MSTP 专有字段长度，该字段用于接收到 BPDU 后进行校验。
- MST 配置标识（Configuration ID）：包含格式选择符（Format Selector）、域名（Configuration Name）、修订级别（Revision Level）和配置摘要（Configuration Digest）四个字段。其中格式选择符字段固定为 0x00，其余三个字段用来判断网桥是否属于某 MST 域。
- CIST 内部路径开销（Internal Root Path Cost，IRPC）：表示发送此 BPDU 的网桥到达 CIST 域根的路径开销。
- CIST Bridge ID：表示发送此 BPDU 的网桥 ID。
- CIST 剩余跳数：用来限制 MST 域的规模。从 CIST 域根开始，BPDU 每经过一个网桥的转发，跳数就被减 1；网桥将丢弃收到的跳数为 0 的 BPDU，使出于最大跳数外的网桥无法参与生成树的计算，从而限制了 MST 域的规模。CIST 剩余跳数默认值为 20。
- MSTI Configuration Messages：包含 0 个或最多 64 个 MSTI（Multiple Spanning Tree Instance，多生成树实例）配置信息，MSTI 配置信息数量由域内 MST 实例数决定，每一个 MSTI 配置信息长度为 16 字节。

5.3 MSTP的工作原理

MSTP 将整个二层网络划分为多个 MST 域，各域之间通过计算生成 CST；域内则通过计算生成多棵生成树，每棵生成树都被称为是一个 MSTI，其中的 MSTI 0 也称为 IST。MSTP 同 STP 一样，使用 BPDU 进行生成树的计算，只是 BPDU 中携带的是设备上 MSTP 的配置信息。

1. CIST 生成树的计算

通过比较 BPDU 后，在整个网络中选择一个优先级最高的设备作为 CIST 的根桥。在每个 MST 域内 MSTP 通过计算生成 IST；同时 MSTP 将每个 MST 域作为单台设备对待，通过计算在域间生成 CST。CST 和 IST 构成了整个网络的 CIST。

2. MSTI 的计算

在 MST 域内，MSTP 根据 VLAN 与 MSTI 的映射关系，针对不同的 VLAN 生成不同的 MSTI。每棵生成树独立进行计算，计算过程与 STP 计算生成树的过程类似，请参见“2.3 STP 的拓扑计算过程”。

MSTP 中，一个 VLAN 报文将沿着如下路径进行转发：

- 在 MST 域内，沿着其对应的 MSTI 转发；
- 在 MST 域间，沿着 CST 转发。

5.4 快速收敛机制

在 STP 中，为避免临时环路，端口从开启到进入转发状态需要等待默认 30 秒的时间，如果想要缩短这个时间，只能手工方式将 Forward Delay 设置为较小值。但是 Forward Delay 是由 Hello Time 和网络直径共同决定的一个参数，如果将 Forward Delay 设置太小，可能会导致临时环路的产生，影响网络的稳定性。

目前，RSTP/PVST/MSTP 都支持快速收敛机制。快速收敛机制包括边缘端口机制、根端口快速切换机制、指定端口快速切换机制。其中指定端口快速切换机制也称为 P/A（Proposal/Agreement，请求/回应）机制。

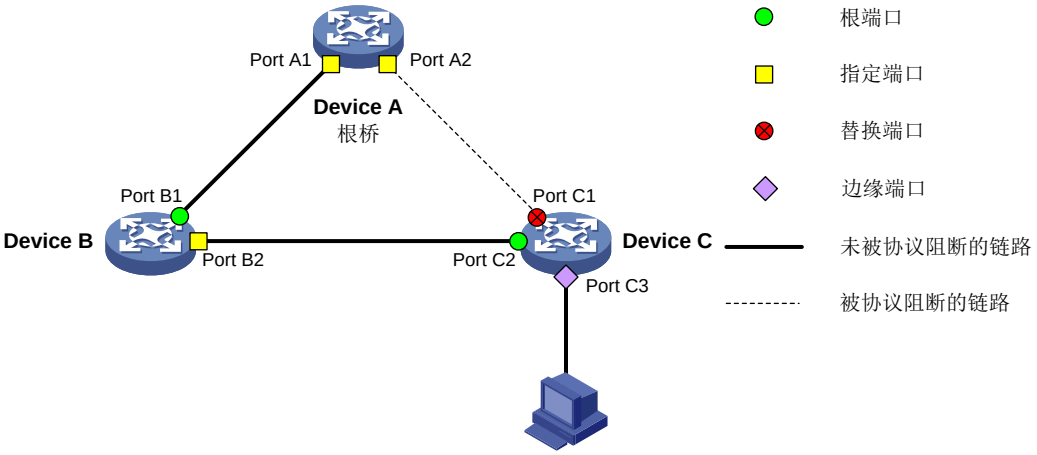
5.4.1 边缘端口机制

当端口直接与用户终端相连，而没有连接到其他网桥或局域网网段上时，该端口即为边缘端口。

边缘端口连接的是终端，当网络拓扑变化时，边缘端口不会产生临时环路，所以边缘端口可以略过两个 Forward Delay 的时间，直接进入 Forwarding 状态，无需任何延时。

由于网桥无法自动判断端口是否直接与终端相连，所以用户需要手工将与终端连接的端口配置为边缘端口。

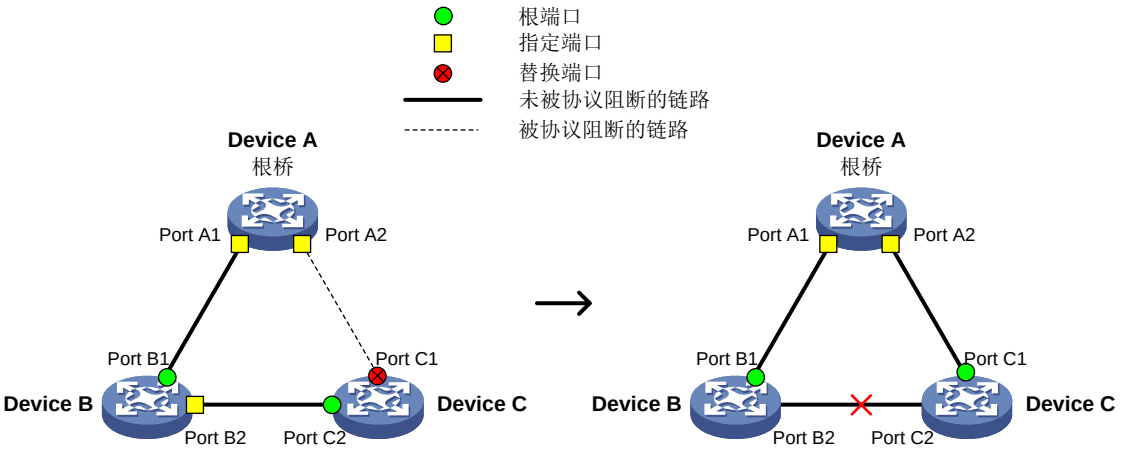
图11 边缘端口示意图



5.4.2 根端口快速切换机制

当旧的根端口进入阻塞状态，网桥会选择优先级最高的替换端口作为新的根端口，如果当前新根端口连接的对端网桥的指定端口处于 Forwarding 状态，则新根端口可以立刻进入 Forwarding 状态。

图12 根端口快速切换示意图



如图 12，Device C 有两个端口，一个为根端口另一个为替换端口，当根端口链路中断时，替换端口会立刻成为新的根端口并进入 Forwarding 状态，期间不需要延时。

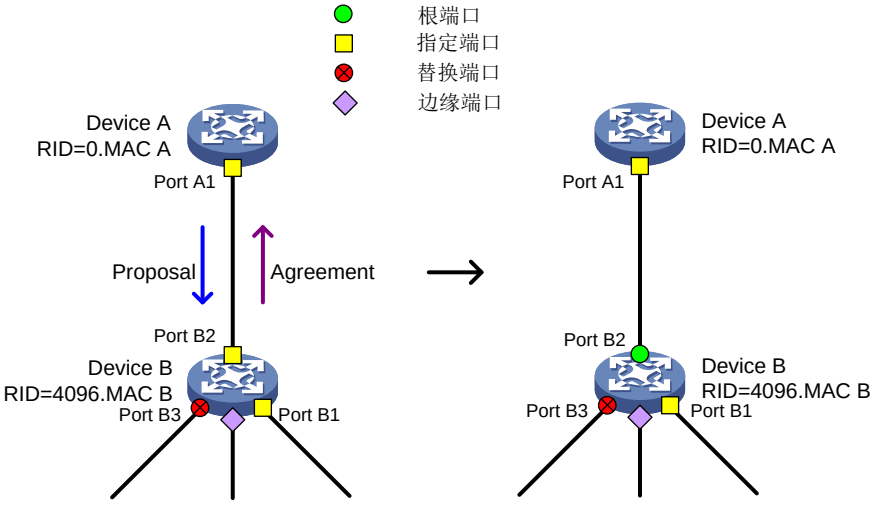
5.4.3 P/A 机制

P/A 机制是指指定端口可以通过与对端网桥进行一次握手，即可快速进入转发状态，期间不需要任何定时器。P/A 机制的前提条件是：握手必须在点到点链路上进行。有点到点链路作为前提，P/A 机制可以实现网络拓扑的逐链路收敛，而不必像 STP，需要被动等待 30 秒的时间以确保全网实现收敛。

1. RSTP/PVST 的 P/A 机制

当新链路连接或故障链路恢复时，链路两端的端口初始都为指定端口并处于阻塞状态。当指定端口处于 Discarding 状态和 Learning 状态，其所发送的 BPDU 中 Proposal 位将被置位，端口角色为指定端口。收到 Proposal 置位的 BPDU 后，网桥会判断接收端口是否为根端口，如果是，网桥会启动同步过程。同步过程指网桥阻塞除边缘端口之外的所有端口，在本网桥层面消除环路产生的可能。

图13 RSTP/PVST 的 P/A 机制实现快速收敛



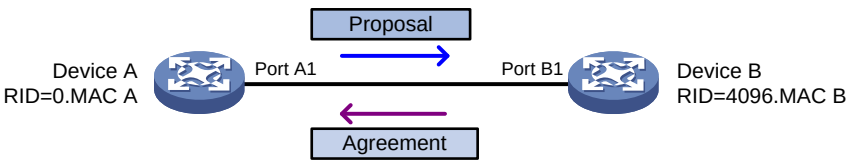
如图 13，当 Device A 和 Device B 之间的链路连接后，P/A 机制处理过程如下：

- Device A 从端口 Port A1 发送 Proposal 置位的 BPDU 给 Device B。
- Device B 收到 Proposal BPDU 后，判断端口 Port B2 为根端口，启动同步过程阻塞指定端口 Port B1 和替换端口 Port B3 避免环路产生，然后将根端口 Port B2 设置为转发状态，并向 Device A 回复 Agreement BPDU。
- Device A 收到 Agreement BPDU 后，指定端口 Port A1 立即进入转发状态。
- Device A 的端口 Port A1 和 Device B 的端口 Port B2 均进入转发状态，P/A 收敛过程结束。

2. MSTP 的 P/A 机制

在 MSTP 中，上游网桥发送的 Proposal BPDU 中的 Proposal 位和 Agreement 位均置位，下游网桥收到 Proposal 位和 Agreement 位均置位的 BPDU 后，执行同步操作然后回应 Agreement 置位的 BPDU，使得上游指定端口快速进入转发状态。

图14 MSTP 的 P/A 机制实现快速收敛



如图 14，Device A 和 Device B 之间的 P/A 机制处理过程如下：

- Device A 从端口 Port A1 发送 Proposal 位和 Agreement 位均置位的 BPDU 给 Device B。

- Device B 收到 Proposal 位和 Agreement 位均置位的 BPDU 后，判断端口 Port B1 为根端口，执行同步操作然后将根端口 Port B1 设置为转发状态，并向 Device A 回复 Agreement BPDU。
- Device A 收到 Agreement BPDU 后，指定端口 Port A1 立即进入转发状态。
- Device A 的端口 Port A1 和 Device B 的端口 Port B1 均进入转发状态，P/A 收敛过程结束。

从 RSTP/PVST 和 MSTP 的 P/A 机制处理过程可以看到，P/A 机制没有依赖任何定时器，可以实现快速的收敛。

需要注意的是，如果指定端口发出的 Proposal BPDU 后没有收到 Agreement BPDU，则该端口将切换到 STP 方式，需要等待 30 秒时间才能进入转发状态。

6 Comware 实现的技术特色

6.1 No Agreement Check功能

RSTP 和 MSTP 的指定端口快速迁移机制使用两种协议报文：

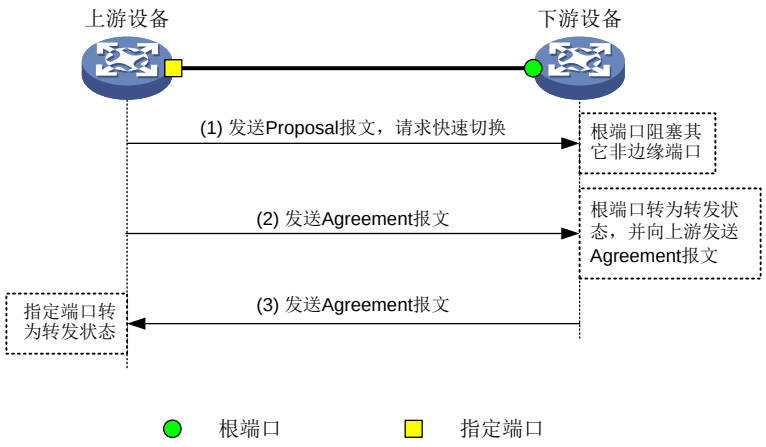
- Proposal 报文：指定端口请求快速迁移的报文。
- Agreement 报文：同意对端进行快速迁移的报文。

RSTP 和 MSTP 均要求上游设备的指定端口在接收到下游设备的 Agreement 报文后才能进行快速迁移。不同之处如下：

- 对于 MSTP，上游设备先向下游设备发送 Agreement 报文，而下游设备的根端口只有在收到了上游设备的 Agreement 报文后才会向上游设备回应 Agreement 报文。
- 对于 RSTP，下游设备无需等待上游设备发送 Agreement 报文就可向上游设备发送 Agreement 报文。

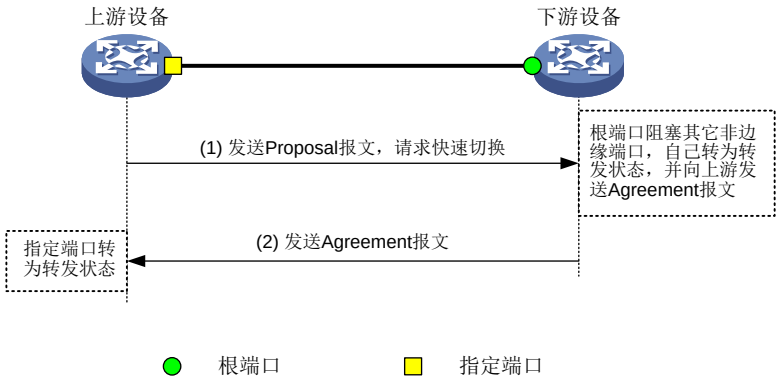
如图 15 所示，是 MSTP 的指定端口快速迁移机制。

图15 MSTP 指定端口快速迁移机制



如图 16 所示，是 RSTP 的指定端口快速迁移机制。

图16 RSTP 指定端口快速迁移机制



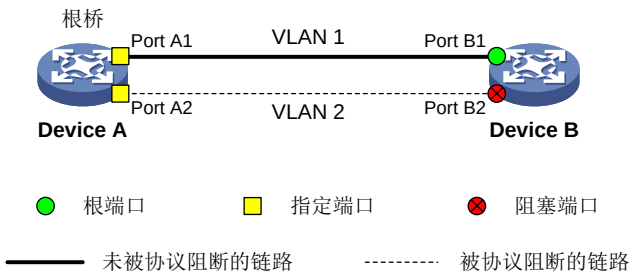
当我方设备与作为上游设备且与对生成树协议的实现存在差异的第三方厂商设备互联时，二者在快速迁移的配合上可能会存在一定的限制。例如：上游设备指定端口的状态迁移实现机制与 RSTP 类似；而下游设备运行 MSTP 并且不工作在 RSTP 模式时，由于下游设备的根端口接收不到上游设备的 Agreement 报文，它不会向上游设备发 Agreement 报文，所以上游设备的指定端口无法实现状态的快速迁移，只能在 2 倍的 Forward Delay 延时后变成转发状态。

通过在我方设备与对生成树协议的实现存在私有性差异的上游第三方厂商设备相连的端口上开启 No Agreement Check 功能，可避免这种情况的出现，使得上游的第三方厂商设备的指定端口能够进行状态的快速迁移。

6.2 VLAN Ignore功能

在网络拓扑比较复杂的情况下，某些 VLAN 的拓扑有可能会被生成树阻塞，造成该 VLAN 的业务流量不通。

图17 MSTP 阻塞 VLAN 连通性示意图



如所示，Device A 的端口 Port A1 允许 VLAN 1 通过，Port A2 允许 VLAN 2 通过；Device B 的端口 Port B1 允许 VLAN 1 通过，Port B2 允许 VLAN 2 通过。Device A 和 Device B 都正常运行生成树协议，通过计算，Device A 为根桥，其端口 Port A1 和 Port A2 为指定端口，Device B 的端口 Port B1 为根端口，Port B2 为阻塞端口，则 VLAN 2 的业务流量无法实现正常连通。

通过在指定 VLAN 上开启 VLAN Ignore 功能，可使该 VLAN 中每个端口的实际转发状态不再遵从生成树的计算结果，而是一直保持转发状态。

6.3 摘要侦听功能

根据 IEEE 802.1s 规定，只有在 MST 域配置（包括域名、修订级别和 VLAN 映射关系）完全一致的情况下，相连的设备才被认为是在同一个域内。当设备开启了生成树协议以后，设备之间通过识别 BPDU 数据报文内的配置 ID 来判断相连的设备是否与自己处于相同的 MST 域内；配置 ID 包含域名、修订级别、配置摘要等内容，其中配置摘要长 16 字节，是由 HMAC-MD5 算法将 VLAN 与 MSTI 的映射关系加密计算而成。

在网络中，由于一些厂商的设备在对生成树协议的实现上存在差异，即用加密算法计算配置摘要时采用私有的密钥，从而导致即使 MST 域配置相同，不同厂商的设备之间也不能实现在 MST 域内的互通。

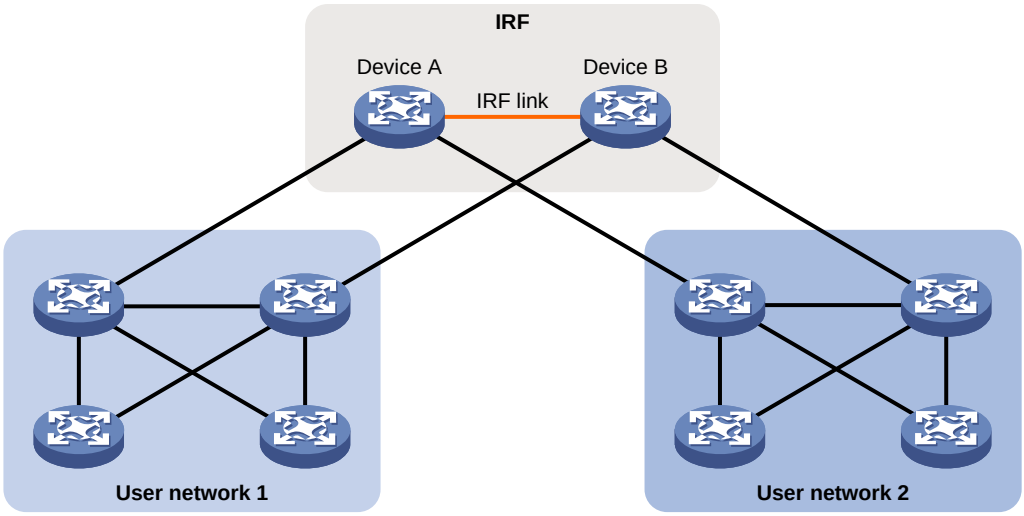
通过在我方设备与对生成树协议的实现存在差异的第三方厂商设备相连的端口上开启摘要侦听功能，可以实现我方设备与这些厂商设备在 MST 域内的完全互通。

6.4 TC Snooping功能

TC Snooping 功能的典型应用环境如图 18 所示。在该组网中，由 Device A 和 Device B 组成的 IRF 设备未开启生成树协议，而用户网络 1 和用户网络 2 中的所有设备均开启了生成树协议。用户网络 1 和用户网络 2 均通过双上行链路与 IRF 设备相连以提高链路可靠性，IRF 设备可以透明传输每个用户网络中的 BPDU。

在该组网中，当用户网络的拓扑结构发生改变时，由于 IRF 设备对 BPDU 进行了透明传输而不参与生成树计算，因而其本身可能需经过较长时间才能重新学到正确的 MAC 地址表项和 ARP 表项，在此期间可能导致网络中断。

图18 TC Snooping 功能典型应用组网图



为了避免这种情况，可以通过在 IRF 设备上开启 TC Snooping 功能，使其在收到 TC-BPDU（网络拓扑发生变化的通知报文）后，主动更新接收该报文的端口所属的 VLAN 所对应的 MAC 地址表和 ARP 表，从而保证业务流量的正常转发。

6.5 执行mCheck操作

生成树的工作模式有 STP 模式、RSTP 模式、PVST 模式和 MSTP 模式四种。在运行 RSTP、PVST 或 MSTP 的设备上，若某端口连接着运行 STP 协议的设备，该端口收到 STP 报文后会自动迁移到 STP 模式；但当对端运行 STP 协议的设备关机或撤走，而该端口又无法感知的情况下，该端口将无法自动迁移回原有模式，此时需要通过执行 mCheck 操作将其手工迁移回原有模式。

当运行 STP 的设备 A、未开启生成树协议的设备 B 和运行 RSTP/PVST/MSTP 的设备 C 三者顺次相连时，设备 B 将透传 STP 报文，设备 C 上连接设备 B 的端口将迁移到 STP 模式。在设备 B 上开启生成树协议后，若想使设备 B 与设备 C 之间运行 RSTP/PVST/MSTP 协议，除了要在设备 B 上配置生成树的工作模式为 RSTP/PVST/MSTP 外，还要在设备 B 与设备 C 相连的端口上都执行 mCheck 操作。

6.6 关闭PVST的PVID不一致保护功能

在当链路相连的两端 PVID 不一致时，PVST 的计算可能出现错误，为了防止这样的错误，系统默认会开启 PVID 不一致保护功能，即做 PVID 不一致的检查。若端口 PVID 不一致保护功能触发后，端口在 PVID 不一致的 VLAN 中，会变为阻塞状态。

在某些特定的组网场景中，比如网络中的接入层设备采用同样的配置，其接口 PVID 一致，而网络管理员在汇聚层设备的下行口（即连接接入层设备的接口）上做了不同的 PVID 配置，该配置与接入层设备的上行口（即连接汇聚层设备的接口）的 PVID 配置不一致时，有可能引起生成树的阻塞，为避免这种情况的发生，保持流量的转发，可以关闭 PVID 不一致保护功能。

6.7 生成树保护功能

6.7.1 BPDU 保护功能

对于接入层设备，接入端口一般直接与用户终端（如 PC）或文件服务器相连，此时接入端口被设置为边缘端口以实现这些端口的快速迁移；当这些端口接收到 BPDU 时系统会自动将这些端口设置为非边缘端口，重新计算生成树，引起网络拓扑结构的变化。这些端口正常情况下应该不会收到 STP 的 BPDU。如果有人伪造 BPDU 恶意攻击设备，就会引起网络震荡。

生成树协议提供了 BPDU 保护功能来防止这种攻击：设备上开启了 BPDU 保护功能后，如果边缘端口收到了 BPDU，系统就将这些端口关闭，同时通知网管这些端口已被生成树协议关闭。被关闭的端口在经过一定时间间隔之后将被重新激活。

6.7.2 根保护功能

本功能应用在设备的指定端口上。

生成树的根桥和备份根桥应该处于同一个域内，特别是对于 CIST 的根桥和备份根桥，网络设计时一般会把 CIST 的根桥和备份根桥放在一个高带宽的核心域内。但是，由于维护人员的错误配置或网络中的恶意攻击，网络中的合法根桥有可能会收到优先级更高的 BPDU，这样当前合法根桥会失去根桥的地位，引起网络拓扑结构的错误变动。这种不合法的变动，会导致原来应该通过高速链路的流量被牵引到低速链路上，导致网络拥塞。

为了防止这种情况发生，生成树协议提供了根保护功能：对于开启了根保护功能的端口，其在所有MSTI上的端口角色只能为指定端口。一旦该端口收到某MSTI优先级更高的BPDU，立即将该MSTI端口设置为侦听状态，不再转发报文（相当于将此端口相连的链路断开）。当在2倍的Forward Delay时间内没有收到更优的BPDU时，端口会恢复原来的正常状态。

6.7.3 环路保护功能

本功能应用在设备的根端口和替换端口上。

依靠不断接收上游设备发送的BPDU，设备可以维持根端口和其他阻塞端口的状态。但是由于链路拥塞或者单向链路故障，这些端口会收不到上游设备的BPDU，此时下游设备会重新选择端口角色，收不到BPDU的下游设备端口会转变为指定端口，而阻塞端口会迁移到转发状态，从而交换网络中会产生环路。环路保护功能会抑制这种环路的产生。

在开启了环路保护功能的端口上，其所有MSTI的初始状态均为Discarding状态：如果该端口收到了BPDU，这些MSTI可以进行正常的状态迁移；否则，这些MSTI将一直处于Discarding状态以避免环路的产生。

6.7.4 端口角色限制功能

本功能应用在与用户接入网络相连的端口上。

用户接入网络中设备桥 ID 的变化会引起核心网络生成树拓扑的改变。为了避免这种情况，可以在端口上开启端口角色限制功能，此后当该端口收到最优根消息时将不再当选为根端口，而是成为替换端口。

6.7.5 TC-BPDU 传播限制功能

本功能应用在与用户接入网络相连的端口上。

用户接入网络的拓扑改变会引起核心网络的转发地址更新，当用户接入网络的拓扑因某种原因而不稳定时，就会对核心网络形成冲击。为了避免这种情况，可以在端口上开启TC-BPDU传播限制功能，此后当该端口收到TC-BPDU时，不会再向其他端口传播。

6.7.6 防 TC-BPDU 攻击保护功能

设备在收到TC-BPDU后，会执行转发地址表项的刷新操作。在有人伪造TC-BPDU恶意攻击设备时，设备短时间内会收到很多的 TC-BPDU，频繁的刷新操作给设备带来很大负担，给网络的稳定带来很大隐患。而通过在设备上开启防TC-BPDU攻击保护功能，就可以避免转发地址表项的频繁刷新。

当开启了防TC-BPDU攻击保护功能后，如果设备在单位时间（固定为十秒）内收到TC-BPDU的次数大于允许收到TC-BPDU后立即刷新转发地址表项的最高次数（假设为N次），那么该设备在这段时间之内将只进行N次刷新转发地址表项的操作，而对于超出N次的那些TC-BPDU，设备会在这段时间过后再统一进行一次地址表项刷新的操作，这样就可以避免频繁地刷新转发地址表项。

6.7.7 MSTP 的 PVST 报文保护功能

对于开启 MSTP 的设备，并不识别 PVST 报文，所以开启 MSTP 的设备会将 PVST 报文当做数据报文转发。在另一个并不相干的网络中，开启 PVST 的设备收到该报文，处理后可能导致该网络的拓扑计算出现错误。

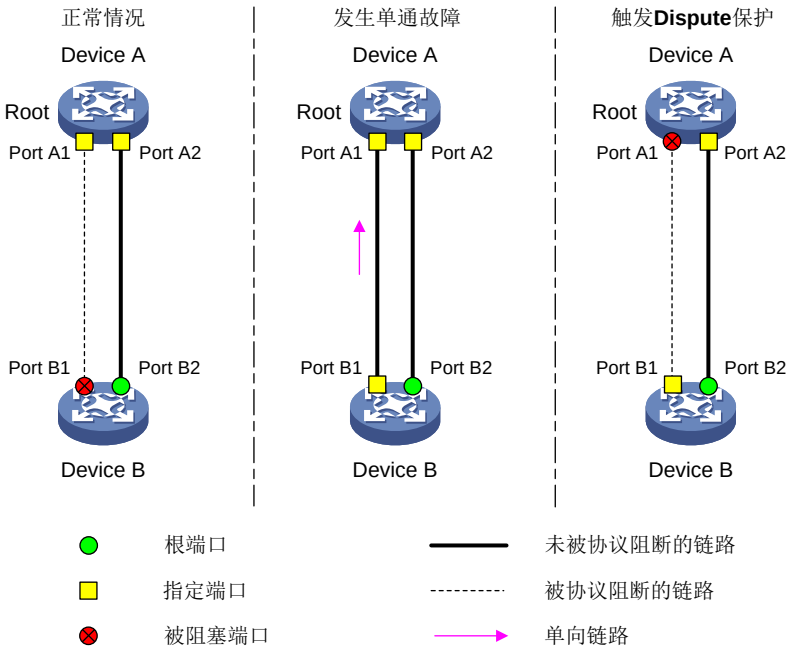
对于这个问题，可以通过 MSTP 的 PVST 报文保护功能来解决。在 MSTP 模式下，设备上开启了 PVST 报文保护功能后，如果端口收到了 PVST 报文，系统就将这些端口关闭。

6.7.8 关闭 Dispute 保护功能

当端口收到指定端口发出的低优先级消息，且发送端口处于 Forwarding 或 Learning 状态时，会触发 Dispute 保护，阻塞端口以防止环路。

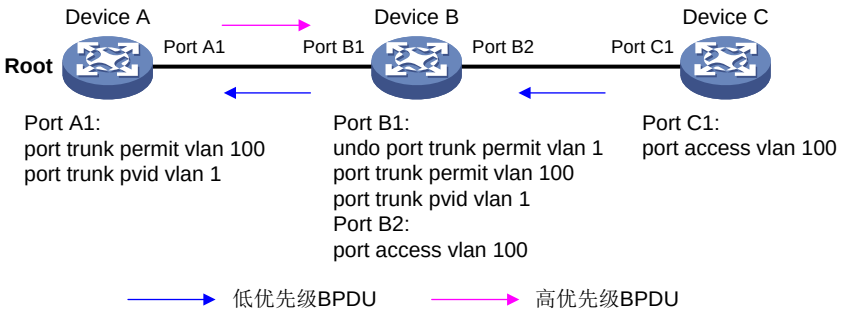
如图 19 所示，正常情况下，Device A 是根桥，经过生成树计算后，Port B1 被阻塞。如果 Port A1 发生单通故障，即 Port A1 不能发送报文，只能接收报文。Port B1 在一定时间内未收到 Port A1 发送的 BPDU，则 Device B 认为自己是根桥，由 Port B1 发送低优先级 BPDU 到 Port A1。此时，Port A2 和 Port B2 之间链路正常，Device B 会接收到自己发送 BPDU，导致产生环路。因此当链路出现单通故障后，会触发 Dispute 保护功能，阻塞端口，防止环路。

图19 Dispute 保护触发场景



在如图 20 所示的 VLAN 组网的场景中，需要关闭 Dispute 保护功能，防止链路被阻塞。Device A 和 Device C 开启生成树功能，Device B 关闭生成树功能，此时 Devcie B 会透传 BPDU。由于 Device B 上 Port B1 的配置，导致 Device C 不能收到根桥 Device A 发送的 VLAN 1 的高优先级 BPDU。Device C 在一定时间内未收到根桥发送的 BPDU，则 Device C 认为自己是根桥，由 Port C1 发送 VLAN 100 的低优先级 BPDU 到 Device A。Device A 收到低优先级 BPDU 后，会触发 Dispute 保护阻塞端口，导致用户业务流量中断。为了保证业务流量正常处理，用户可以关闭 Dispute 保护功能，避免链路被生成树阻塞而影响用户业务。

图20 关闭 Dispute 保护功能使用场景

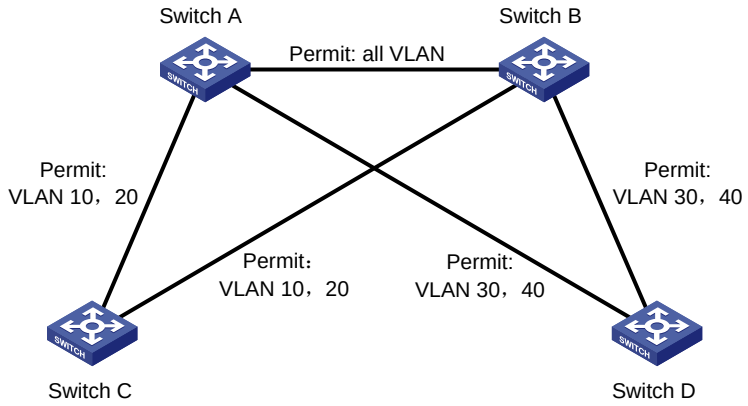


7 典型组网应用

7.1 MSTP典型组网应用

MSTP 可以使得同一组网中的不同 VLAN 的报文按照不同的生成树进行转发，从而实现不同 VLAN 数据的负载分担和冗余备份。

图21 MSTP 典型组网图

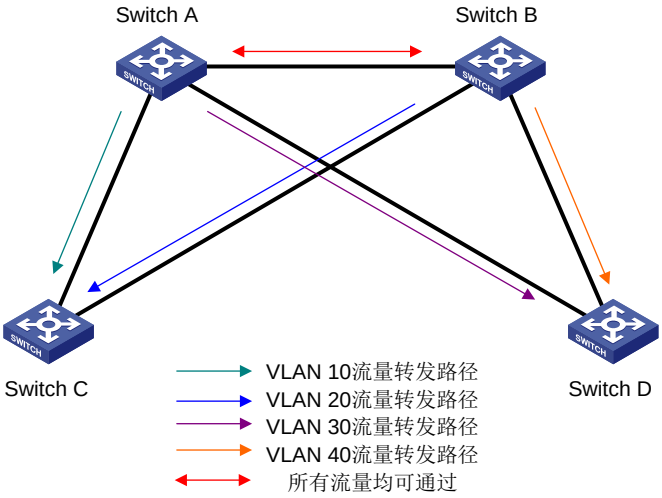


如图 21 所示，Switch A 和 Switch B 为汇聚层设备，Switch C 和 Switch D 为接入层设备。为了合理均衡各条链路上的流量，可以在设备上按照下列思路进行配置：

- 所有设备属于同一个 MST 域；
- VLAN 10 的报文沿着实例 1 转发，实例 1 的根桥为 Switch A；
- VLAN 20 的报文沿着实例 2 转发，实例 2 的根桥为 Switch B；
- VLAN 30 的报文沿着实例 3 转发，实例 3 的根桥为 Switch A；
- VLAN 40 的报文沿着实例 4 转发，实例 4 的根桥为 Switch B。

MSTP 计算完成后，不同 VLAN 流量的转发路径如图 22 所示，这样可以大大减少各链路的负载。同时，每个 VLAN 都有一条冗余备份链路，当前工作链路失效后，冗余备份链路会马上生效，大大减小由于链路故障而导致的流量丢失。

图22 流量转发路径图



对于上述组网，也可以部署 PVST 协议来达到同样负载分担和链路备份的目的。配置如下：

- VLAN 10 的根桥为 Switch A；
- VLAN 20 的根桥为 Switch B；
- VLAN 30 的根桥为 Switch A；
- VLAN 40 的根桥为 Switch B。

8 参考文献

- IEEE 802.1D：Media Access Control (MAC) Bridges
- IEEE 802.1w：Part 3: Media Access Control (MAC) Bridges—Amendment 2: Rapid Reconfiguration
- IEEE 802.1s：Virtual Bridged Local Area Networks—Amendment 3: Multiple Spanning Trees
- IEEE 802.1Q-REV/D1.3：Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks—Clause 13: Spanning tree Protocol