# Cisco − Understanding Spanning−Tree Protocol Topology Cha

# Table of Contents

# Understanding Spanning−Tree Protocol Topology Changes

## Contents



## Introduction

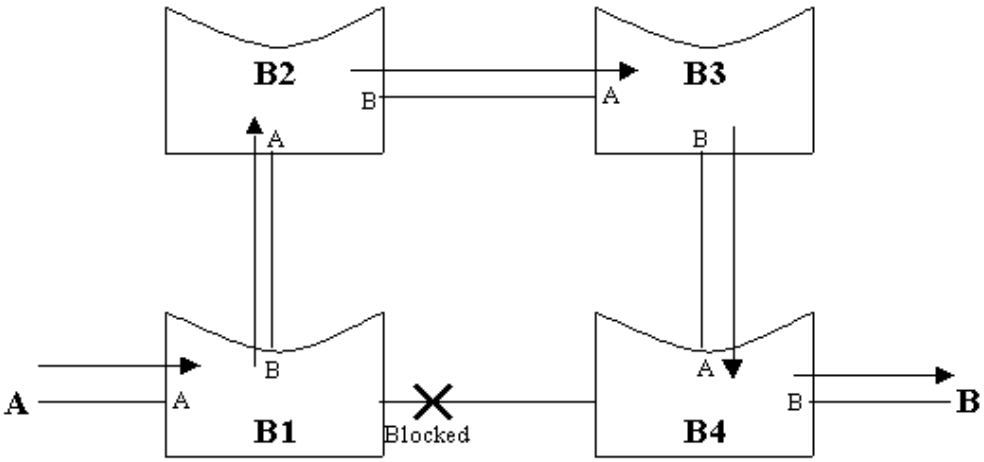
When monitoring Spanning−Tree Protocol (STP) operations, you may be concerned when seeing topology change counters incrementing in the statistics log. Topology changes are normal in STP, but too many of them can have an impact on network performances. This document explains the purpose of this topology change mechanism, what triggers a topology change event, and describes some issues related to the topology change mechanism.

## Purpose of the Topology Change Mechanism

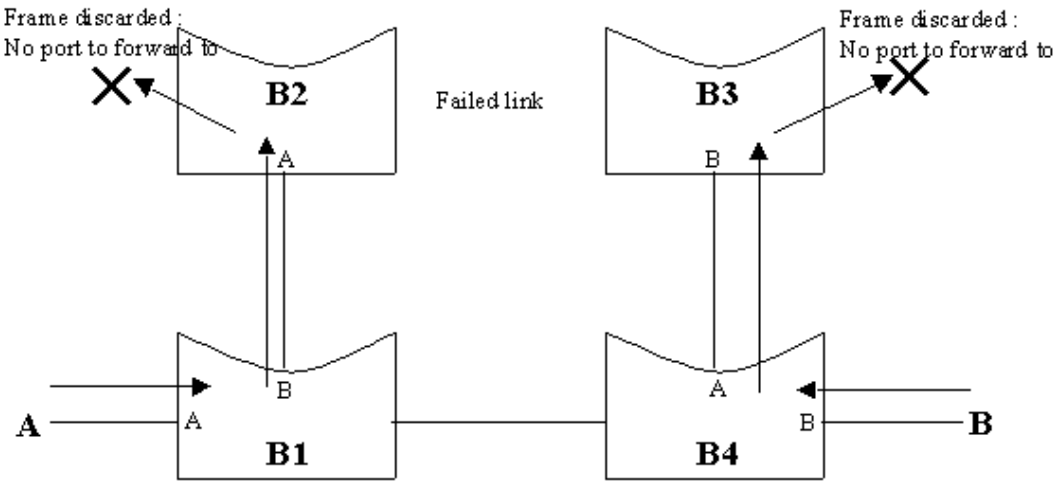
Learning from the frames it receives, a bridge creates a table that associates to a port the Media Access Control (MAC) addresses of the hosts that can be reached via this port. This table is used to forward frames directly to their destination port, therefore avoiding flooding.

Default aging time for this table is 300 seconds (5 minutes). Only after a host has been silent for five minutes, its entry disappears from the table of the bridge. Here is an example showing why we could want this aging to be faster:

In this network, let's suppose that bridge B1 is blocking its link to B4. A and B are two stations that have an established connection. Traffic from A to B is going to B1, B2, B3, and then B4. The scheme shows the MAC addresses table learned by the four bridges in this situation:




Now, let's suppose that the link between B2 and B3 fails. Communication between A and B is interrupted at least until B1 puts its port to B4 in forwarding mode (a maximum of 50 seconds with default parameters). But when A wants to send a frame to B, B1 still has an entry leading to B2 and the packet is sent to a black hole. The same applies when B wants to reach A: communication is lost for five minutes, until the entries for A and B MAC addresses age out.




The forwarding databases implemented by bridges are very efficient in a **stable** network, but there are many situations where the five minute aging time is a problem after the topology of the network has changed. The topology change mechanism is a workaround for that kind of problem. As soon as a bridge detects a change in the topology of the network (basically a link going down or going to forwarding), it advertises the event to the whole bridged network. We'll see in the next section how this is practically implemented. Every bridge is then notified and reduces the aging time to forward_delay (15 seconds by default) for a certain period of time (max_age + forward_delay). It is more clever to reduce the aging time instead of simply clearing the table because currently active hosts, that effectively transmit traffic, are not cleared from the table.

So, in our example, as soon as bridge B2 or B3 detects the link going down, they send topology change notifications. Very soon, all bridges are aware of the event and reduce their aging time to 15 seconds. As B1 doesn't receive any packet from B on its port leading to B2 in 15 seconds, it ages out the entry for B on this port. The same happens to the entry for A on port leading to B3 on B4. Then, when the link between B1 and B4 goes to forwarding, traffic is immediately flooded and re−learned on this link.

# Principle of Operation

This section explains at the Bridge Protocol Data Unit (BPDU) level how a bridge advertises a topology change.

We already briefly explained when a bridge considers it detected a topology change. The exact definition is:

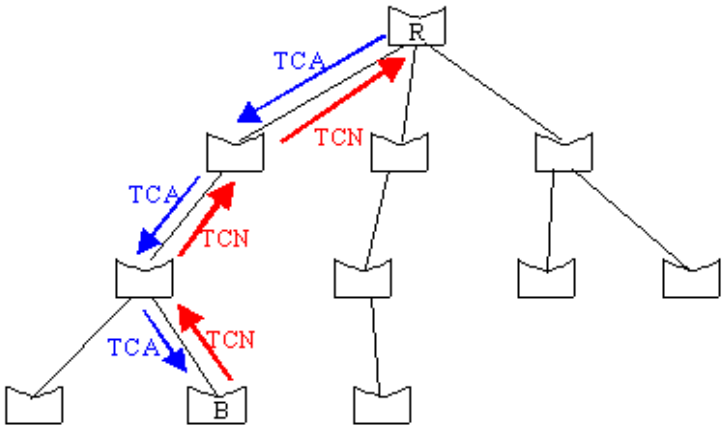
- When a port that was forwarding is going down (blocking for instance).
- When a port transitions to forwarding and the bridge has a designated port. (This basically means that the bridge is not standalone.)

The real challenge here is to send a notification to all bridges in the network. This is achieved in two steps:

- The bridge notifies the root bridge of the spanning tree.
- The root bridge "broadcasts" the information into the whole network.

## Notifying the Root Bridge

In normal STP operation, a bridge keeps receiving configuration BPDUs from the root bridge on its root port, but it never sends out a BPDU toward the root bridge. So, in order to achieve that, a special BPDU called the topology change notification (TCN) BPDU has been introduced. Thus, when a bridge needs to signal a topology change, it starts sending TCNs on its root port. The designated bridge receives the TCN, acknowledges it, and generates another one for its own root port. And so on until the TCN hits the root bridge.



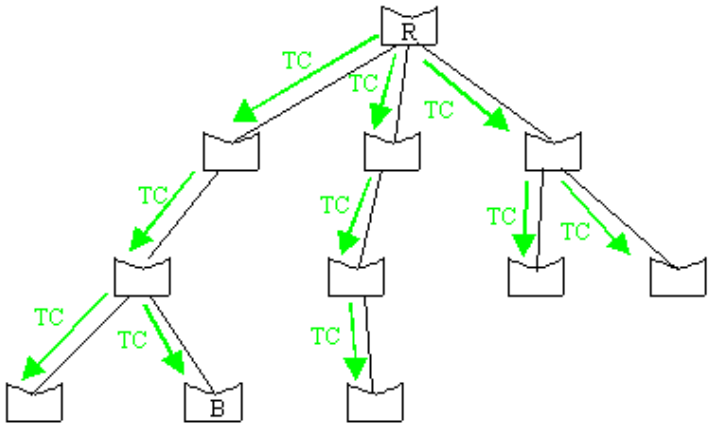

Bridge B notifies a topology change by sending a TCN on its root port. The TCN is acknowledged and forwarded up to the root bridge R.

The TCN is a very simple BPDU that contains absolutely no information that a bridge sends out every hello_time seconds (this is locally configured hello_time, not the hello_time specified in configuration BPDUs). The designated bridge acknowledges the TCN by immediately sending back a normal configuration BPDU with the topology change acknowledgement (TCA) bit set. The bridge notifying the topology change will not stop sending its TCN until the designated bridge has acknowledged it, so the designated bridge answers the TCN even though it does not receive configuration BPDU from its root.

## Broadcasting the Event to the Network

Once the root is aware there has been a topology change event in the network, it starts sending out its configuration BPDUs with the topology change (TC) bit set. These BPDUs are relayed by every bridge in the network with this bit set, so that every single bridge is aware of the topology change situation and can reduce its aging time to forward_delay.

The TC bit will be set by the root for a period of max_age + forward_delay seconds, which is 20+15=35 seconds by default.



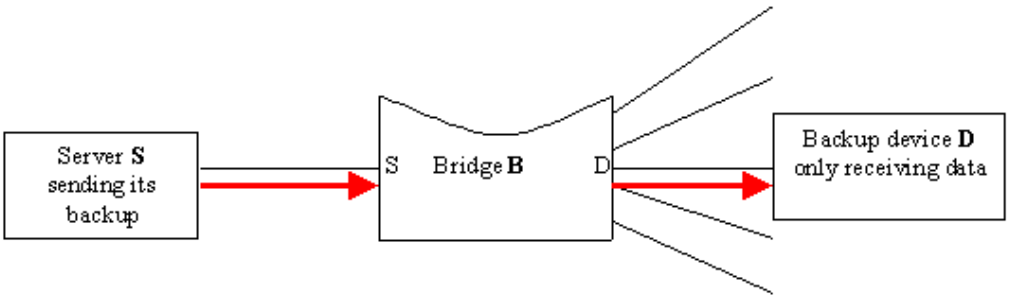

The root R sets the TC bits in its bpdus. This bpdu is relayed to the whole network.

# What to Do When There are Many Topology Changes in the Network

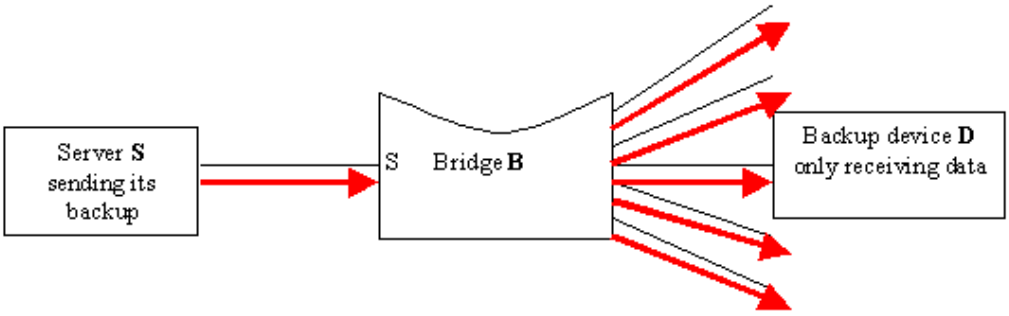
Here are the only few problems that can be generated by TCN, followed by some information on how to limit topology changes and find from where they are coming.

## Flooded Traffic

The more hosts are in the network, the higher are the probabilities of getting a topology change. For instance, a directly attached host will trigger a topology change when it is power cycled. In very large (and flat) networks, we can reach a point where the network is perpetually in a topology change status. Practically, this is as if the aging time was configured to 15 seconds, which will lead to a high level of flooding. Here is a worst case scenario that happened at one of our customers who was doing some server backup.




Server S is sending heavy unicast traffic to device D. Due to the nature of the protocol used, D nearly never sends any traffic. If a topology change occurs and B reduces its ageing time, entry for D will be removed from B.




Traffic is then flooded on the whole network and will reduce the bandwidth of every single link until device D sends a frame again.

The aging out of the entry for the device receiving the backup was a disaster as it caused a very heavy traffic to hit all users. See later for how to avoid TCN generation.

## Problem in ATM LANE Bridged Environments

This case is more critical than the normal flooding of traffic implied by a quick aging. On the receipt of a topology change for a VLAN, a Catalyst switch will have its LAN emulation (LANE) blades reconfirming their LE−arp table for the corresponding emulated LAN (ELAN). As every LANE blade in the ELAN will issue at the same time the same request, it may put a high stress on the LAN Emulation Server (LES) if there are a lot of entries to reconfirm. We have seen connectivity issues in this scenario. If the network is sensitive to a topology change, the real problem is not the topology change itself but the design of the network. Though, it really makes sense to limit as much as possible the TCN generation to save the CPU of the LES (at least). See the following section limiting TCN generation.

## Avoiding TCN Generation with the Portfast Command

The portfast feature is a Cisco proprietary change in the STP implementation. The command is applied to specific ports and has two effects:

- Ports coming up in are put directly in the forwarding STP mode, instead of going through the learning and listening process. Note that STP is still running on ports with portfast.
- The switch never generates a TCN when a port configured for portfast is going up or down.

You can then enable portfast on ports where are connected hosts that a very likely to bring their link up and down (typically end stations that users frequently power cycle). The feature should not be necessary for server ports and should be definitely avoided on ports leading to hubs or other bridges: a port directly transitioning to forwarding state on a redundant link can cause a temporary bridging loop or can cause temporary bridging loops

But remember that topology changes can be useful, so don't enable portfast on port for which a link going up or down is a significant event for the network.

### Tracking the Source of a TCN

In itself, a topology change is not a bad thing, but as a good network administrator, it is better to know where they are coming from in order to be sure they are not related to a real problem. Identifying the bridge that issued the topology change is not an easy task, though not technically very complex.

Most bridges only count the number of TCNs they have issued or received. The Catalyst 4000, 5000, and 6000 are able to show the port and the ID of the bridge that sent the last topology change they received. Starting from the root, it is then possible to go downstream to the initiator bridge. See information on the **show spantree statistics** command for more information.

## Conclusion

Remember that a TCN does not start a STP recalculation. This fear comes from the fact that TCNs are often associated with unstable STP environments; TCNs are a consequence of this, not a cause. The TCN only has an impact on the aging time; it will not change the topology nor create a loop.

The number or the rate of topology changes is not an issue in itself. The problem is to know what the topology change means. A healthy network can experience a high rate of topology change. Nevertheless, ideally, a topology change would be related to a significant event in the network like a server going up or down or a link transitioning. This can be achieved by enabling portfast on ports that are going up and down as part of their normal operation.

## Related Information

- **LAN Technologies Technical Tips**