during this time are kept in the bridge table. This condition lasts for the sum of the Forward Delay and the Max Age (default 15 + 20 seconds).

The theory behind topology changes is fairly straightforward, but it's often difficult to grasp how a working network behaves during a change. For example, suppose that you have a Layer 2 network (think of a single VLAN or a single instance of STP) that is stable and loop free. If a switch uplink suddenly failed or a new uplink was added, how would the various switches in the network react? Would users all over the network lose connectivity while the STP "recomputes" or reconverges?

Examples of different types of topology changes are presented in the following sections, along with the sequence of STP events. Each type has a different cause and a different effect. To provide continuity as the STP concepts are presented, the same network previously shown in Figures 7-3 through 7-5 is used in each of these examples.

Direct Topology Changes

A direct topology change is one that can be detected on a switch interface. For example, if a trunk link suddenly goes down, the switch on each end of the link can immediately detect a link failure. The absence of that link changes the bridging topology, so other switches should be notified.

Figure 7-6 shows a network that has converged into a stable STP topology. The VLAN is forwarding on all trunk links except port 1/2 on Catalyst C, where it is in the Blocking state.

This network has just suffered a link failure between Catalyst A and Catalyst C. The sequence of events unfolds as follows:

1. Catalyst C detects a link down on its port 1/1; Catalyst A detects a link down on its port 1/2.
2. Catalyst C removes the previous "best" BPDU it had received from the root over port 1/1. Port 1/1 is now down so that BPDU is no longer valid.

   Normally, Catalyst C would try to send a TCN message out its root port, to reach the root bridge. Here, the root port is broken, so that isn't possible. Without an advanced feature such as STP UplinkFast, Catalyst C isn't yet aware that another path exists to the root.

   Also, Catalyst A is aware of the link down condition on its own port 1/2. It normally would try to send a TCN message out its root port to reach the root bridge. Here, Catalyst A *is* the root, so that isn't really necessary.
3. The root bridge, Catalyst A, sends a Configuration BPDU with the TCN bit set out its port 1/1. This is received and relayed by each switch along the way, informing each one of the topology change.
4. Catalysts B and C receive the TCN message. The only reaction these switches take is to shorten their bridging table aging times to the Forward Delay time. At this point, they don't know how the topology has changed; they only know to force fairly recent bridging table entries to age out.

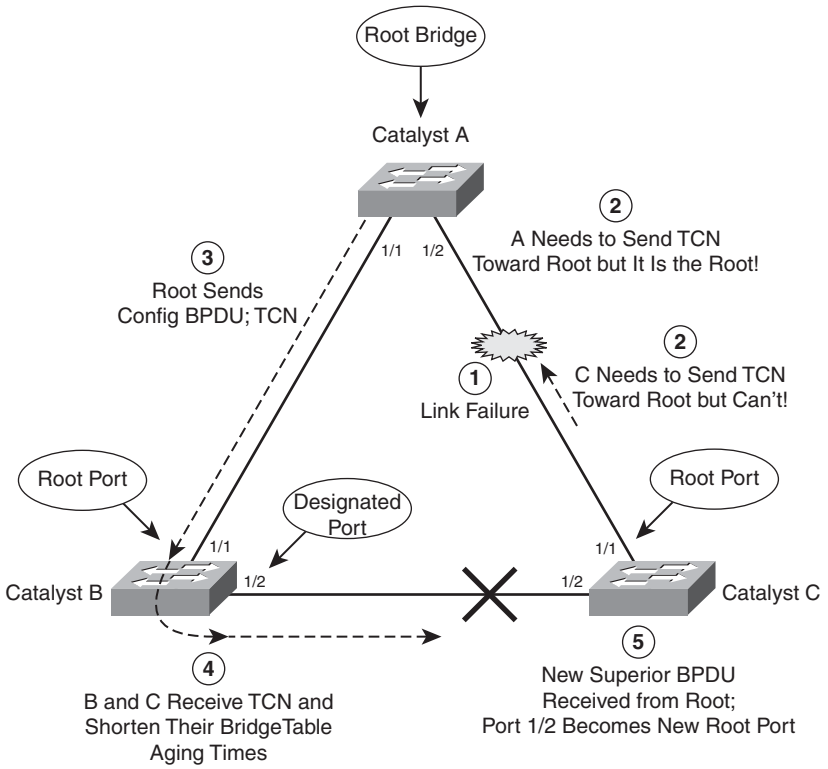

Figure 7-6 *Effects of a Direct Topology Change*

5. Catalyst C basically just sits and waits to hear from the root bridge again. The Config BPDU TCN message is received on port 1/2, which was previously in the Blocking state. This BPDU becomes the "best" one received from the root, so port 1/2 becomes the new root port.

   Catalyst C now can progress port 1/2 from Blocking through the Listening, Learning, and Forwarding states.

As a result of a direct link failure, the topology has changed and STP has converged again. Notice that only Catalyst C has undergone any real effects from the failure. Switches A and B heard the news of the topology change but did not have to move any links through the STP states. In other words, the whole network did not go through a massive STP reconvergence.

The total time that users on Catalyst C lost connectivity was roughly the time that port 1/2 spent in the Listening and Learning states. With the default STP timers, this amounts to about two times the Forward Delay period (15 seconds), or 30 seconds total.

Indirect Topology Changes

Figure 7-7 shows the same network as Figure 7-6, but this time the link failure indirectly involves Catalysts A and C. The link status at each switch stays up, but something between