Truncating tree timing

Mick Seaman

This note proposes changes to the spanning tree protocol that significantly reduce the duration of service outages caused by equipment failure in switched LANs. These improvements are effective in networks of arbitrary topology, complementing a prior proposal targeting networks designed from the outset to provide high availability. They remove the protocol's reliance on estimated, worst case, network wide delays. If the final active topology cuts a potential loop one bridge away from a currently listening designated port, that port can transition to forwarding after a single message exchange with its neighbour. This might typically complete within a second - in sharp contrast to the standard default of thirty seconds. If the potential loop is to be blocked several bridges distant, the cut is moved there one bridge hop at a time.

The proposed improvement can be introduced into some or all bridges in a network without degrading the level of service provided by existing bridges. The benefit of faster reconfiguration will be experienced whenever a designated port attached to a point to point link between two upgraded bridges is blocked during reconfiguration.

Introduction

The IEEE Std. 802.1D Spanning Tree Protocol selects a fully connected ("spanning") loop-free ("tree") active topology from the potentially redundantly connected LANs and bridges of a bridged local area network. The active topology comprises bridges connected to LANs through bridge ports that are in a Forwarding state. Potential loops are defeated by holding other bridge ports in a Blocking state.

The spanning tree algorithm is distributed amongst the bridges. All communication between them is through the frame transmission services provided by the LANs, with variable delays and some probability of loss. When the required active topology changes, the algorithm cannot assume that changes in the states of bridge ports are synchronized. To prevent temporary loops, transitions to the Forwarding state are delayed.

The default delay allows for spanning tree information propagation in a maximum sized network. It can be actively managed to reduce reconfiguration time. However such management requires an accurate and complex assessment of the worst case timing in a particular network[1]. Requiring explicit network management subverts the simplicity and plug and play objectives of the spanning tree. Moreover the best reconfiguration time that can be achieved is still based on a worst case estimate. Thus the spanning tree protocol suffers from the defects common to protocols that are fundamentally timer based[2].

A prior proposal [1][3] shows how a bridge's Root Port[4] can be transitioned to a Forwarding state without delay. This note proposes new mechanisms, applicable to Designated Ports connected to point-to-point links, that allow such ports to transition to forwarding on the basis of the actual communication delay between neighbouring bridges.

References

[1] High Availability Spanning Tree. Mick Seaman. Rev 1.1 10/26/98

[1] See 802.1D Annex B to start.

[2] The best protocols only use timers to recover from conditions so rare that there is little reason to tweak timer values.

[3] Implementation of High Availability Spanning Tree is required for the additional improvements proposed in this note to be effective.

[4] The port that provides the shortest path from a given bridge to the uniquely selected root of the spanning tree.

The Problem

The prior proposal [1] reconfigures extremely rapidly if reselection of the root port on one or more bridges can provide failure recovery. Figure 1 provides an example of a network designed so that this is true for any single failure.

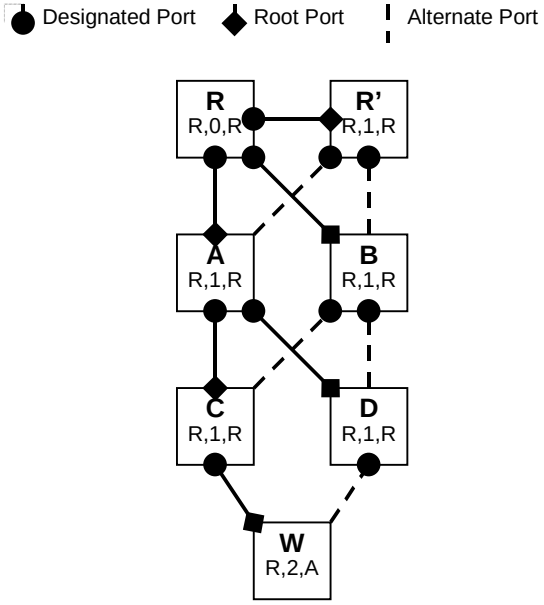
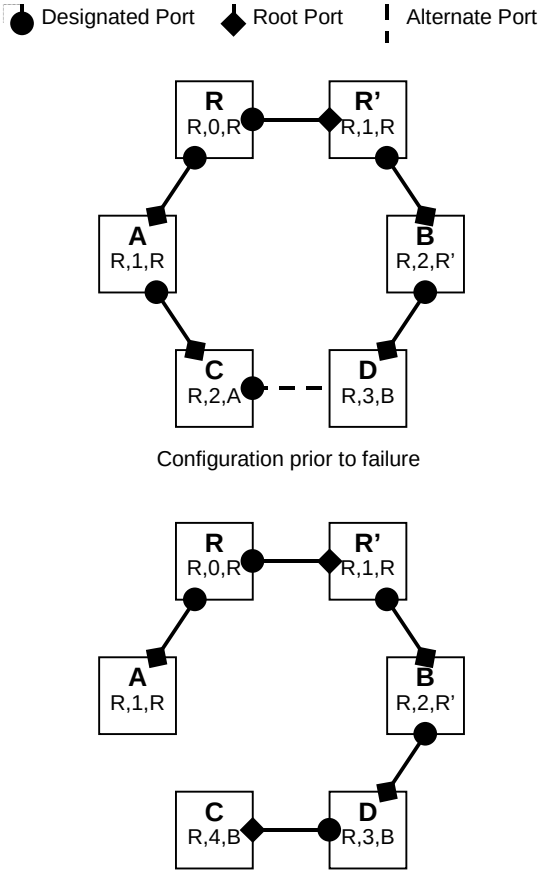


Figure 1

However, in an arbitrary topology, recovery may require that a newly designated port transition to forwarding. Figure 2 provides an example of a backbone ring design[5], and such a failure.



After link A-C fails, C selects C-D as root port. D is now designated bridge for that link, and has to transition to forwarding to complete the recovery.

Figure 2

[5] Not the best way to build switched networks today, but such configurations do exist in real networks.

The Solution

If, in the above example, D is made aware that C had chosen C-D as its root port, and no longer had connectivity through C's previous root port, then D could immediately transition the newly designated port D-C to Forwarding.

In general, any designated port can be made forwarding immediately, provided that:

(a) it connects to a point to point link[6], and

(b) the bridge at the other end of the link has selected that port as its root port or as an alternate port, i.e. it is not also believed to be a designated port, and

(c) if the bridge at the other end of the link believes its port to be the root port, i.e. it has or proposes to make the port forwarding, and it has recently become the root port, then the immediately prior root port or root ports must be made blocking.

If [1] is used in its simplest form ([1] Table 3), in which prior root ports are always made blocking, this final condition is not required. Otherwise, where an attempt is made to maximize connectivity through transitions, ([1] Table 4 or Table 5), the action required can be obtained by sending a "retire root port" signal to all designated ports. Those that have recently been the root port will be forced to the Listening or Learning state.

These rules prevent temporary loops such as might arise in the following example (Figure 3).

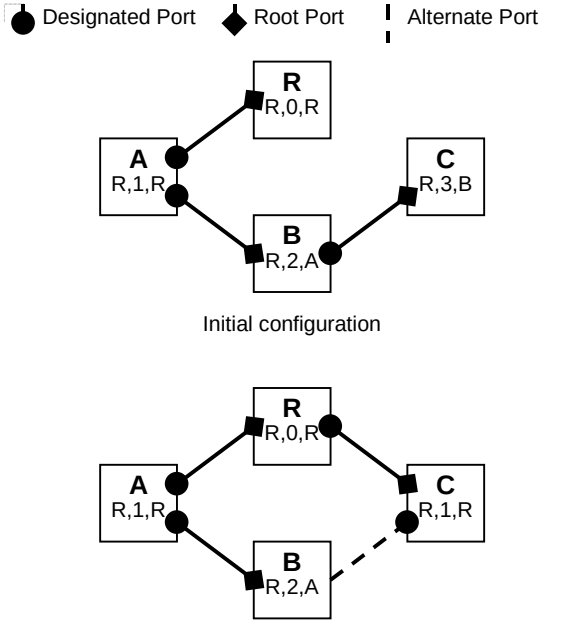


Figure 3

[6] An extension to shared media could be made, but would require all attached bridges to agree on the designated bridge.

In this case the reconfiguration would proceed as follows.

1. R sends a Configuration BPDU to C on the newly established link, claiming to be the designated bridge for that link, and requesting a fast transition to forwarding.
2. C receives that information and agrees that R, and not C is the designated bridge. C selects C-R as its new root port, and prior to notifying R of the agreement, transitions its prior root port, C-B, to blocking. It then sends a message to R telling R that it can safely transition the port to forwarding.
3. R receives the message and transitions its port to forwarding.
4. C now attempts to become the designated bridge for the link C-B, and to transition that port to forwarding rapidly. So far as C knows that could be an essential part of the connectivity. It sends an appropriately marked Configuration BPDU to B.
5. B replies, telling C that it can go ahead. Because B's root port remains B-A, there is nothing further to be done.

Since no further connectivity is added by the B-C link[7], the reconfiguration is complete and the final active topology established once a message has gone from R to C and back again. This is a great improvement on waiting thirty seconds!

Signaling Requirements

The proposed improvement requires the spanning tree protocol to carry a small amount of additional information.

(a) A designated bridge on a point to point link has to be able to request its partnering bridge[8] on the link to return information as to its own port's spanning tree role and to confirm that the loop cut has been moved to its own designated ports if necessary.

(b) The partner has to be able to return the confirmation.

At minimum this requires two extra bits for communication, one in each direction[9]. If the partner does not wish to return the confirmation he can simply allow the normal lengthy spanning tree timers to run their course[10]. This eventuality has to be acceptable to the originator of the transaction anyway, since the partner bridge may

[7] Apart from some possible connectivity to a higher layer protocol entity in B itself

[8] Or attached end system if the spanning tree protocol is being used to provide redundant links through diverse network paths.

[9] It is important to realize that the originator of the fast transition request only needs to know its partner's port role, not that that role was a result of the latest information sent. There is, therefore, no need for sets of sequence numbers or anything like that.

[10] This also protects against message loss.

not have implemented the improvement. It is probably desirable to include an unambiguous statement of port role in the message to aid network diagnosis.

On the assumption that a limited number of bits[11] are actually used, these can be sent in the flags field of a Configuration BPDU. The additional information in the Configuration BPDU returned by the port that is not designated will be ignored by its partner[12], as will the extra flags if received by a bridge not implementing this improvement.

Conclusion

This note extends the mechanisms described in “High Availability Spanning Tree” in order to hasten failure recovery and reconfiguration in arbitrary switch topologies. While these may not support service restoration within tens of milliseconds, sub-second recovery times are possible, even for bridges implementing spanning tree processing in normal software operating system processes.

The proposed enhancements are backwards compatible, bridges implementing them can be freely mixed with existing standard bridges in a network without degrading the performance of the latter. Where any two new bridges are connected by point to point links the benefit of lower reconfiguration time may be enjoyed.

[11] Somewhere between 2 and 6.

[12] Again this information may be useful for diagnostics.