

CS423: Lecture 17, Spanning Tree Protocol: Details

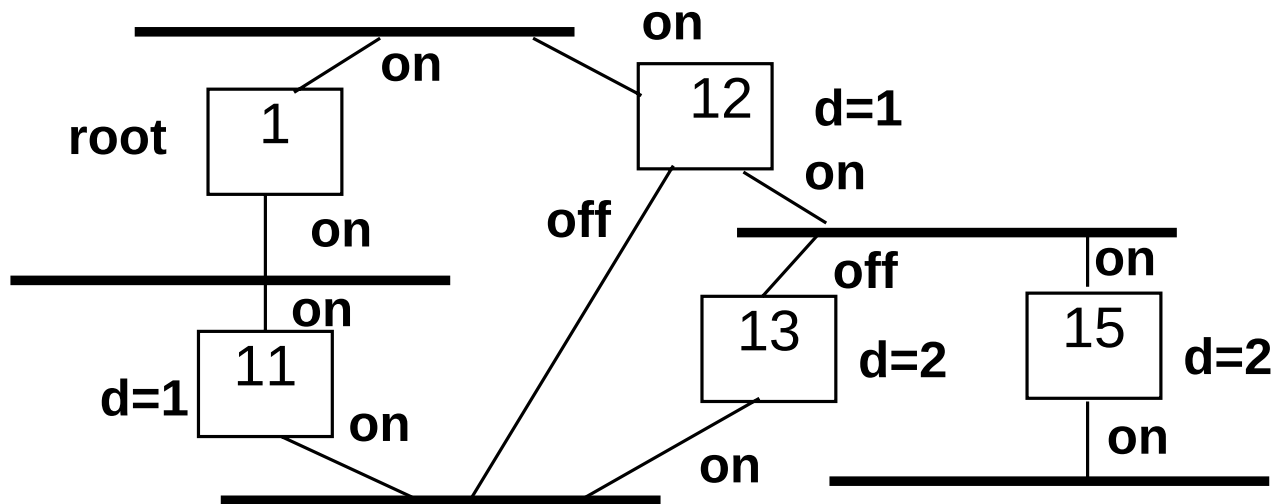
George Varghese

October 26, 1995

Spanning Tree Protocol

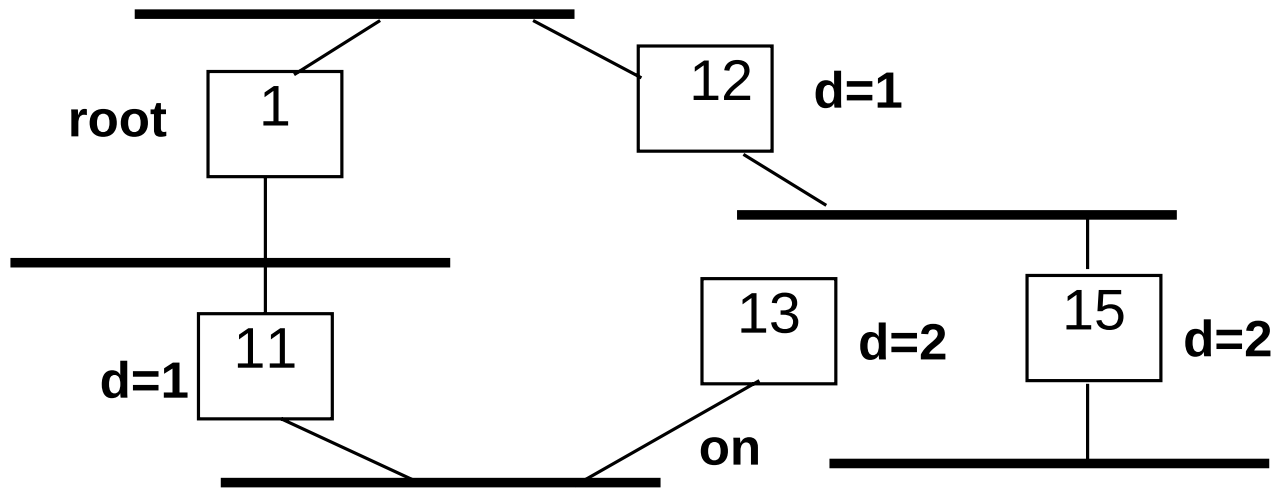
- Basic Centralized Algorithm
- Basic Distributed Algorithm
- Failures
- Cache Failures
- Other Details

Centralized Algorithm



- Each bridge finds Min port — port through which it has shortest path to root (a.k.a. parent)
- Each bridge finds the ports for which this bridge is on the shortest path between root and corresponding LAN: Designated Ports.
- Each bridge turns ON Min port and all Designated Ports. ON, OFF are software states: always receive hello and management messages on all ports. Drop data packets to/from OFF port.

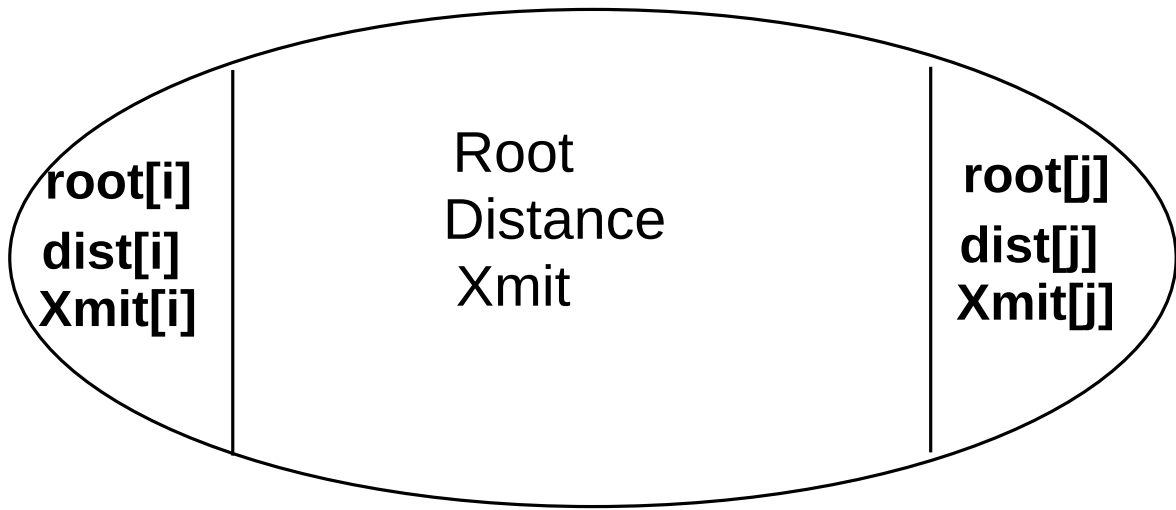
Final Tree



- To be a tree, each LAN must have a unique path to every other LAN.
- Algorithm guarantees that each LAN can get to root only through designated bridge for LAN and designated bridges have unique path through their parent LAN.

Distributed Algorithm

- Easy to calculate root and distance using a centralized algorithm. How to calculate if each bridge powers up knowing only its own ID and can only get info from received messages.
- Want to calculate Min ID node (root) and distance to Min ID node. Can keep an estimate of Root and Dist and send updates to neighbors and update own estimates based on neighbors distances.
- First try to find an update rule for root estimates based on received estimates from neighbors. Then find an update rule for distance, assuming your distance is right.



root	dist	Xmit	
98	12	13	Worse Hellos ↓
99	12	13	
99	14	13	
99	14	12	

Consider as a tuple with Root as MS

SPANNING TREE CODE

1) Initially:

Central and Port Values = me, 0, me

2) Periodically: send a HELLO message on all ports for which you are designated:

Multicast to All Local Bridges	Me	LLC	root	dist	me
-----------------------------------	----	-----	------	------	----

3) On receipt of HELLO on Port X:

If received hello is better than Port [X] value
Store received HELLO in Port [X]

SPANNING TREE CODE (continued)

- 3) Find Min of Port databases and me,0, me
If Min = me then central value = me, 0, me
else if Min = Port [X] then
 Root = root [X]
 Dist = dist [X] + 1
 Xmit = me
- 4) For all Ports Y compare central to Port Y
 If central is better write central to Port Y
- 5) Turn ON the "Min" port and all ports for
 which you are best (i.e., central = port).
 Turn OFF all other ports.

Failures

- Basic idea: if any path from root to a bridge dies then we must get rid of old, incorrect info.
- Thus we want to timeout information received on every port.
- Idea is that when the root is alive it must periodically send HELLOs on every port; the children then send the info down the tree.
- If no hellos are received within a certain time on a port, we remove that information from the port database.

Enhancements

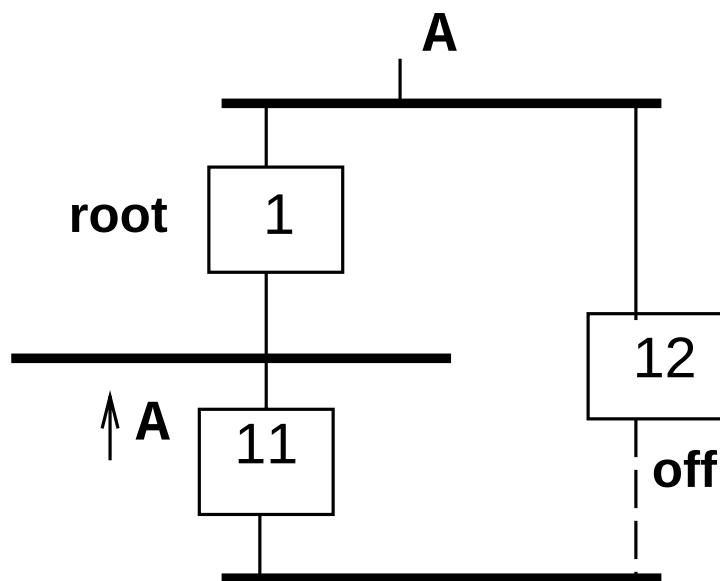
- **Avoiding temporary loops:** Wait a time period T before moving from OFF (Blocked) to ON (Forwarding). Have a PRE-ON state; if you stay in PRE-ON state for a T without changing state, then transition to ON. In ON state, you forward data packets received on port and learn from such packets. See handout for details.
- **Handling station movement due to topology changes:** A bridge that detects a port state change informs its parent, and so on till root knows. Root then includes a Topology Change Flag (TCF) for some time in its hellos until all bridges hear of the change. While TCF is set in hellos, bridges use a smaller cache timer (2 minutes) instead of the usual longer cache timer (15 minutes).

Final Details

- **Minimize Flooding:** Learning in last half of PRE-ON state.
- **Allowing managers to control topology:** Managers can specify a cost for each link instead of using a default hop cost of 1. (High speed links can be given low costs). Instead of having root being picked by lowest ID (which cannot be set by manager) we have a root priority field which is considered the MSB of a 7 byte root field.
- **Subtle Tiebreakers:** For cases where a bridge gets same info from parent on two ports, the Xmit field is insufficient and we need to add a (Port-ID) Xmit-Port field.

FAILURE CODE

- a) Add a hello-age field to HELLO messages
- b) Add an age field to all port databases.
- c) Root always sends hellos with age 0
- d) Modify send (if non-root) to send hellos to children when you get hello from root, your info changes, or you get a worse hello
- e) Store received hello if equal and better age. (Modify Receive-Hello code).
- f) Modify Step 3 to set Age = Age [Min]
- g) On Timer Tick:
 - Increment all port ages.
 - Age = Age [Min]
 - If Age [X] > Max-Age then
 - Port [X] = me, 0, me
 - Redo Steps 3, 4 and 5



**What happens
to direction of
A at Bridge 11
when root dies?**

