Clause 9 Editorial Proposals

Bob Sultan

Proposals

- Description of fairness algorithm
- Description and terminology for MC-FCM
- Clarification of aggressive and conservative

Fairness algorithm description

- Follow example of 802.1D STP
  - informative overview
  - informative 'elements of procedure' (suggest omit)
  - <u>normative</u> code

- Why not normative state tables
  - fairness algorithm not highly 'state driven'
  - 25 character column makes code difficult to read and no room for comments
  - maintain informative state tables where useful

Normative ANSI C (excerpt)

```
record_config_information(port_no, config) /* (8.6.2) */
Int port_no;
Config_bpdu *config;
{
port_info[port_no].designated_root = config->root_id; /*
   (8.6.2.3.1) */
port_info[port_no].designated_cost = config->root_path_cost;
port_info[port_no].designated_bridge = config->bridge_id;
port_info[port_no].designated_port = config->port_id;
start_message_age_timer(port_no, config->message_age); /*
   (8.6.2.3.2) */
}
```

Informative overview (excerpt)

8.3.4 Changing Port State

Since there are propagation delays in passing protocol information throughout a Bridged LAN, there cannot be a sharp transition from one active topology to another. Topology changes may take place at different times in different parts of the Bridged LAN and to move a Bridge Port directly from nonparticipation in the active topology to the Forwarding State would be to risk having temporary data loops and the duplication and misordering of frames. It is also desirable to allow other Bridges time to reply to inferior protocol information before starting to forward frames.

Explanation of code as normative

8.9 Procedural model

This subclause constitutes the definitive description of the operation of the Spanning Tree Algorithm and Protocol. The natural language text in 8.6, 8.7, and 8.8 of this standard is intended to informally present the semantics of operation specified here. Should differences of interpretation exist between that text and this procedural model, the latter shall take precedence.

Explanation of code as normative

8.9.1 Overview

The parameters, timers, elements of procedure, and operation of the protocol are presented below as a compilable program in the computer language C (ANSI X3.159). The objective of presenting this program is to precisely and unambiguously specify the operation of the algorithm and protocol. The description of the operation of the protocol in a computer language is in no way intended to constrain the implementation of the protocol; a real implementation may employ any appropriate technology.

Fairness control messages

- SC-FCM conveys rate/congestion info one hop
  - <u>essential</u> element of fairness algorithm
- MC-FCM bcasts rate/congestion info to clients
  - <u>no</u> role in fairness algorithm
  - reports (previously computed) fair rates to clients
- <u>Function</u> of MC-FCM is essential
  - change name to rate reporting message (RRM)
    - or rate distribution message (RDM)
  - move out of clause 9 as unrelated to fairness algorithm

Conservative vs. Aggressive

- no more than 10% code difference
  - do <u>not</u> describe as two entirely distinct procedures
- explain the effects of trading utilization for 'stability'
  - provide guidelines for selecting mode
- provide explanation of how stations of different modes coexist on the same ring
  - provide simulation or analytic results to indicate that rates remain fair

Proposals

- C code normative for fairness algorithm
- Grant editor license to construct suitable overview
- Change MC-FCM to rate reporting message (RRM)
- Change SC-FCM to FCM
- Remove RRM description to separate clause
- Clarification of aggressive vs. conservative