Convergence of 802.1Q, PFC, AVB, and ETS

Joe Pelissier

bb-pelissier-convergence-proposal-1108

Agenda

- **Where we are today**
- **Some issues with ETS**
- **Goals**
- **Simplifications**
- **A proposed enhancement**
- **Examples**
- **Observations**

Where we are today…

- **802.1Q Specifies a Priority to Traffic Class Table**
  - **All .1Q switches support 8 priorities**
  - **Table maps the priorities into supported traffic classes**
    - **Between 1 and 8**
- **This is highly desirable**
  - **Ensures defined interoperability between bridges of different abilities**
  - **Bridges do not need to know the number of traffic classes supported by their neighbors**
    - Each bridge exercises its best effort to support the priority scheme

ETS provides a similar abstraction…

- **Priority to Priority Group Table**
  - **Each Priority or a set of Priorities mapped to a Priority Group**
  - **Neighbor switches are unaware of the mapping**
- **However:**
  - **This is not exactly the abstraction we want**
  - **Priority Groups define the bandwidth allocation, therefore this is generally an externally defined behavior**
    - We would like each switch to exercise its best effort to allocate bandwidth as defined by the Priority Groups
      - While providing opaqueness regarding the number of supported traffic classes

The Priority Group table is problematic…

- The Priority Group table has dependencies on the programming of Traffic Class table
  - Scheduling based on Traffic Class
  - Assume Priorities 1 & 2 are assigned to TC1
  - Assume Priority 3 assigned to TC2
  - Assume Priority 1 assigned to PG1 and 2&3 assigned to PG2
  - ->Behavior is undefined
- It would be much cleaner if it were not possible to program the tables in such a way that undefined behavior results

PFC also provides an external abstraction…

- **PFC is enabled or disabled per priority**
  - **All PFC capable bridges will support 8 priorities and the ability to enable / disable per priority**
- **Works for switches with less than 8 traffic classes**
  - **On Transmit: if one priority is flow controlled, then in general all traffic from the corresponding traffic class halts**
  - **On Receive: if a traffic class is becoming full, then all priorities assigned to that traffic class for which PFC is enabled may assert flow control**
    - Or flow control may be asserted based on some local policy
- **Again, switch exercises best effort and no drop due to congestion is assured**

Goals:

- **Provide a consistent external abstraction for Priority, ETS, PFC, and AVB**

  Bridges exercise best effort based on number of traffic classes supported, in a defined manner

- **Define a deterministic manner to map the external abstraction to the available Traffic Classes**
- **Eliminate the ability to program a bridge in a conflicting manner**

A new term: Traffic Type

- **There are five different Traffic Types that must be considered:**

  AVB: see 802.1Qav

  EP: ETS with PFC enabled

  En: ETS without PFC enabled

  nP: No ETS with PFC enabled

  nn: everything else (i.e. non-ETS and non-PFC)

A few proposed simplifications:

- Require all Priorities within a Priority Group to all be either PFC or not PFC (no mixing of PFC and non-PFC within a Priority Group)
- On an ETS enabled switch, the Priority to Traffic Class mapping is implied via the Priority Group mapping table
  - The Priority to TC table becomes read only
  - The mapping of Priority to TC is performed as described in the following slides
- A DCB switch must support at least six Traffic Classes if it supports AVB, otherwise it must support at least four Traffic Classes

Step 1: Define the Priority to PG table

- **Contains eight entries, one for each Priority**

  All DCB switches must support 8 Priorities and 8 PGs; however, these may be merged into fewer traffic classes as specified later

- **Each entry contains one value: the PG to which the Priority is assigned**

  Each Priority must be assigned to exactly one PG

  A PG may have multiple Priorities assigned to it

- **Default values: Priority Group = Priority**

Step 2: Define the PG Table

- **Contains eight entries, one for each Priority Group**
  - **All DCB switches must support 8 Priorities and 8 Priority Groups; however, these may be merged into fewer traffic classes as specified later**
- **Each entry contains three values:**
  - **Traffic Type (AVB, EP, En, nP, nn, or unused)**
  - **Bandwidth Allocation (percentage) (reserved for AVB, nP, nn, and unused)**
  - **Traffic Class (read only, traffic class to which this PG is assigned based on the algorithm defined in this slide set)**
    - A PG is assigned to exactly one TC
    - A TC may have multiple PGs assigned to it, all containing the same Traffic Type

Step 3: Determine the number of TCs for each TT

- **It is desirable for each Priority Group to be assigned to a separate Traffic Class, if possible**
  - **If there are not enough Traffic Classes, then multiple Priority Groups are merged into Traffic Classes**
    - Similar to the merging of Priorities into Traffic Classes today
    - All merged Priority Groups within a given Traffic Class must have the same Traffic Type
- **Based on the following rules:**
  - **A separate Traffic Class is assigned for each AVB Priority Group (up to two)**
  - **A Traffic Class is allocated for each Priority Group that contains a Traffic Type that does not appear in any other Priority Group**
  - **Considering only the remaining Priority Groups, Traffic Types, and Traffic Classes, each set of Priority Groups containing a given Traffic Type is allocated at least one Traffic Class**
    - If additional Traffic Classes are available, they may be allocated as indicated in the Priority Group to Traffic Class Allocation tables provided (recommended) or by other implementation specific means.
- **These seem confusing, but they become fairly self-evident when one observes the actual possible combinations**

Priority Group to Traffic Class Allocation

- **See TBD for a table of these allocations.**

  There are 495 possible combinations of Priority Group to Traffic Class allocations

Step 4: Allocate Priority Groups to Traffic Classes

- Priority Groups with Traffic Type EP are allocated the lowest numbered Traffic Classes, followed by En, nn, AVB, and unused
- Merge Priority Groups into Traffic Classes from lowest to highest numbered
- If possible, the same number of Priority Groups are assigned to each Traffic Class for a give Traffic Type

  If not, the lowered numbered Traffic Classes are assigned one more Priority Group than the higher numbered Traffic Classes

- If multiple Priority Groups with EP or En Traffic Types are merged into a single Traffic Class, then the bandwidth allocated to that Traffic Class is the sum of the merged Priority Groups

Step 5

- **Populate the read only Priority to Traffic Class table**

  Simply extract from the Priority to PG table and the PG table

An non-AVB Example

This is what the administrator would set up:

Priority	Traffic Contents	Priority Group
7	High Priority Management Traffic	7
6	Standards Group LAN	6
5	Engineering LAN	5
4	IPC	4
3	Storage	3
2	iSCSI	2
1	All other LAN	1
0	Backup	0

Priority to Priority Group Table

Priority Group	Traffic Type	BW Allocation	Traffic Class (RO)
7	nn	na (reserved)	
6	nn	na (reserved)	
5	nn	na (reserved)	
4	EP	50	
3	EP	30	
2	En	20	
1	nn	na (reserved)	
0	nn	na (reserved)	

Priority Group Table

Determine Number of Traffic Classes

Priority Group	Traffic Type	BW Allocation	Traffic Class (RO)
7	nn	na (reserved)	
6	nn	na (reserved)	
5	nn	na (reserved)	
4	EP	50	
3	EP	30	
2	En	20	
1	nn	na (reserved)	
0	nn	na (reserved)	

- **Assume bridge supports 4 Traffic Classes**

  Only 1 Priority Group is carrying En Traffic so it is allocated one Traffic Class

  This leaves two Priority Groups carrying EP traffic and 5 carrying nn traffic, with only 3 Traffic Classes remaining.

  Using the table, refer to row {EP,NP,En,nn}={2 0 1 5}

  From the table, we see that the EP traffic is allocated 1 Traffic Class, En is allocated 1 Traffic Class and the nn traffic gets 2 Traffic Classes (kind of self-evident)

Assigning the Traffic Classes

- EP is assigned first; one Traffic Class for all EP Priority Groups, which would be Traffic Class 0
- Since there is no nP traffic, En is assigned next, with one Traffic Class (Traffic Class 1)
- nn is next with two Traffic Classes (Traffic Classes 2 and 3)

    Note the lower numbered Traffic Classes & Priority Groups get greater merging

- Note that since Priority Groups 3 and 4 are merged into one Traffic Class, the BW allocation for that class is the sum of the merged Priority Groups, or 80%

Priority Group	Traffic Type	BW Allocation	Traffic Class (RO)
7	nn	na (reserved)	3
6	nn	na (reserved)	3
5	nn	na (reserved)	2
4	EP	50	0
3	EP	30	0
2	En	20	1
1	nn	na (reserved)	2
0	nn	na (reserved)	2

An AVB Example

This is what the administrator would set up:

Priority	Traffic Contents	Priority Group
7	High Priority Management Traffic	7
6	iSCSI	2
5	AVB Type 1	5
4	AVB Type 2	4
3	IPC	3
2	Storage	1
1	Backup	0
0	General IP Traffic	0

Priority to Priority Group Table

Priority Group	Traffic Type	BW Allocation	Traffic Class (RO)
7	nn	na (reserved)	
6	unused	na (reserved)	
5	AVB	na (reserved)	
4	AVB	na (reserved)	
3	EP	50	
2	En	20	
1	EP	30	
0	nn	na (reserved)	

Priority Group Table

Determine Number of Traffic Classes

Priority Group	Traffic Type	BW Allocation	Traffic Class (RO)
7	nn	na (reserved)	
6	unused	na (reserved)	
5	AVB	na (reserved)	
4	AVB	na (reserved)	
3	EP	50	
2	En	20	
1	EP	30	
0	nn	na (reserved)	

- **Assume bridge supports 5 Traffic Classes**

  AVB always gets separate Traffic Classes, so one Traffic Class each is allocated to Priority Groups 5 and 4

  Priority Group 2 is the only one with En traffic, so it is allocated one Traffic Class

  There are two Priority Groups with nn traffic and two with EP traffic, but only two Traffic Classes remain, therefore, Priority Group 3&1 share a Traffic Class and Priority Group 7&0 share a Traffic Class

Assigning Traffic Classes

- **EP is assigned first; one Traffic Class has been allocated for EP, which would be Traffic Class 0 (start from 0 and work up)**
- **En is assigned next, with one Traffic Class (Traffic Class 1)**
- **nn is next with one Traffic Class (Traffic Class 2)**
- **AVB is last, with a Traffic Class each (Traffic Class 3 and Traffic Class 4)**
- **Note that since we merged Priority Group 1 and 3 into one Traffic Cass, the bandwidth allocation for that class is the sum of the merged Priority Groups, or 80%**

Priority Group	Traffic Type	BW Allocation	Traffic Class (RO)
7	nn	na (reserved)	2
6	unused	na (reserved)	
5	AVB	na (reserved)	4
4	AVB	na (reserved)	3
3	EP	50	0
2	En	20	1
1	EP	30	0
0	nn	na (reserved)	2

Observations

- **The Priority to Priority Group table and Priority Group table provide a consistent external abstraction**
- **The algorithm maps these tables into the available traffic classes in a deterministic manner**
  - **Available traffic classes remains opaque externally**
  - **Bridges exercise best effort to exhibit behavior defined by the two tables**
  - **Reasonable approximation is obtained if there are fewer Traffic Classes than Priority Groups**
- **The tables cannot be programmed in such a way that results in undefined behavior**

Thank You!