A Problem With Percents

Joe Pelissier

az-pelissier-percentproblem-0209

Background

- **We seem to have consensus that a percent of link bandwidth is a convenient method for the user to specify ETS bandwidth**
    - **Consequently, we should ensure that our underlying MIB (and SNMP access of it) as well as DCBX supports this**
- **The CEE Author's DCBX proposal takes this one step further**
    - **It requires that the percentage allocation *always* equal 100**
        - Otherwise the operation of the bridge is *undefined*.
    - **This is problematic from a an SNMP perspective**

A Possible MIB structure:

- Below is an example of what the MIB *might* look like:

```
lldpXdcbxFeatPgBwAllocBwOper    OBJECT-TYPE
   SYNTAX      LldpXdcbxPgBw
   UNITS       "percent"
   MAX-ACCESS  read-only
   STATUS      current
   DESCRIPTION
      "The operating bandwidth allocated to this priority group.”

::= { lldpXdcbxFeatPgBwAllocEntry 3 }
```

LldpXdcbxPgBw is defined as an Integer (0..100)
The above MIB object is part of a row in a table that has one entry for each Priority Group

MIB / SNMP Observation

- Structuring the MIB to contain a table with a row for each Priority Group clearly is the most straight forward approach.

  Representing the bandwidth as an integer with a valid range of 0-100 is exactly what we want to enable setting the bandwidths as a percentage

- However, consider the fact that SNMP can only set on object in a MIB at a time…

  Note: yes I know one can transport multiple sets in a single SNMP PDU. However, these sets are not atomic, the operate as independent sets.

An example

Current State

PG	BW
0	0
1	0
2	20
3	80
4	0
5	0
6	0
7	0

Desired State

PG	BW
0	0
1	0
2	20
3	50
4	30
5	0
6	0
7	0

We are asking SNMP to change the current state to the desired state by:

- **Setting one entry at a time**
- **After each set, the total BW must equal 100**

Obviously, this cannot be done…

Three Possible Solutions

- Change the MIB somehow so that all of the entries can be programmed at once
- Remove the restriction that the total must be 100

  Leave the switch behavior as undefined when the total does not equal 100

  Define the switch behavior when the total does not equal 100.

Change the MIB

- **One possible method is to not use a table structure for the Priority Groups**
  - **Instead, use a single object to contain all eight bandwidths**
  - **An octet string of length eight could do this**
- **However, this results in a much less obvious MIB structure**
  - **I know of no similar use of an octet string in any standard IETF or IEEE MIB (although I have not done an exhaustive search)**
  - **I suspect that we would need a good reason for doing this to get such a structure past the MIB police**
    - *In other words, why is it so important that this always total to 100?*

Remove the Restriction…

- Lets be clear about something:
- From a user interface perspective, it makes perfect sense to enforce the total to be 100%
  - We are not defining a user interface here
  - Removing the restriction in the implementation in no way hinders a management application (or CLI, or any other form of user interface) from enforcing this restriction
  - It does, however, make the management application simpler to implement

Remove the Restriction…

- **…and leave the behavior undefined**
- **After all, this is just a transitory condition while the table is being programmed**
- **However:**
  - **There is no way to bound how long "transitory" is**
    - We are not writing an SNMP specification
  - **Could be several seconds depending on the SNMP implementation and workload**
  - **Could remain "transitory" indefinitely**
- **Leaving a behavior undefined for a condition we know with 100% certainty will occur under normal operations seems like bad specmanship**
  - ***Besides, why is it so important that this always total to 100?***

Remove the Restriction…

- **…and define the behavior**
- **The ETS proposal currently states:**

  “Configured PG% (PG Percentage in Table 2) refers to the max percentage of available link bandwidth after priorities within PGID 15 are serviced, and assuming that all PGs are fully subscribed.”

- **We can add the sentence:**

  “If the percentages in Table 2 do not total to 100, the maximum percentage of available link bandwidth shall be the Configured PG % multiplied by 100 and divided by the total.”

- **This is trivial to implement**

  Its likely to be what most implementers would do as the “undefined behavior”.

  Keeps the MIB and management operations simple

For consideration

- There seems little down side to removing the restriction and defining the behavior
  - The defined behavior is a trivial software that works regardless of whether the total is 100 or not
- Leaving the behavior undefined seems to have no implementation value
- Changing the MIB to a more awkward form seems like a compromise solution to an artificial problem
  - We have no real need to enforce this restriction
  - Creates additional uncertainty during the ballot process

Postel's Law: Be conservative in what you do;
be liberal in what you accept from others

Thank You!