DCBX Goals and Services

Eric Multanen – Intel

DCBX Goals

- Discover DCB capabilities of peer
- Identify configuration problems
  - Exchange relevant configuration of DCB features with peer
  - Defend as appropriate (e.g. disable the DCB feature)
  - Notification of issues
- Provide for “negotiation” of DCB features
  - One peer can take its DCB configuration from the other

intel

DCBX Services (End Station perspective)

Applications need to know whether and when required features are operational on the Link

- don't start up if specific features are not configured correctly
  - peer-to-peer configuration (e.g. peer is going to honor PFC)
  - cross feature configuration (e.g. App setting is consistent with PFC setting)
- avoidance of reconfiguration thrash
  - e.g. Baseline DCBX – only changes operational state when 'synced'
- detection of operational transitions
  - not just locally but on the peer as well
  - when should a notification be generated
- "plug-and-play" ability
  - start up an End Station and it will learn and adapt to the DCB network by default
- manual override option

intel

Result: Baseline DCBX Proposal

The required goals and services of DCBX, plus the general principle that it would be “bad” to have a misconfigured DCB feature on a link:

- Feature is “on” only if both sides have a consistent configuration
- Otherwise feature should be off on both sides

Resulted in the Baseline DCBX Proposal

- Control State Machine - SeqNo/AckNo for syncing with peer
- Management bits (Advertise, Willing, Enable)
- Feature state machines (managing Operational State, synced state)

Reference:

- az-wadekar-dcbxcapability-exchange-discovery-protocol-1108-v1.01.pdf

intel

The End Result

- Retain the DCBX concept and name
  - "Everyone" is talking about it/expecting it
- Continue to achieve the goals and services provided by the Baseline DCBX proposal
  - Plus more
- Simplfied specification
  - Top down (rework the Baseline proposal) vs. Bottom up (Joe's work)

Thanks!