Constrained Aggregations

Predictable, deterministic, results for common constraints ….

or ...

Avoiding dynamic keys

Mick Seaman
mick_seaman@3com.com

Further details

"Constrained Aggregations", Mick Seaman,
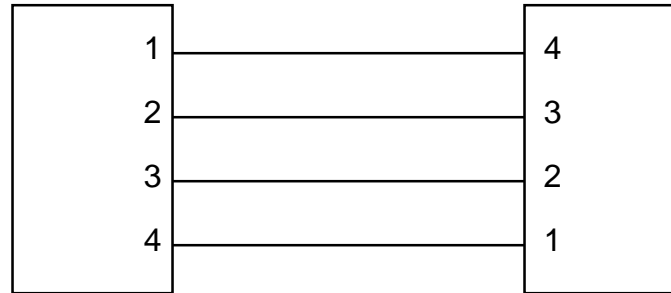 Rev1.0 1/9/99

P802.3ad D0.1

"Link Aggregation Control Protocol",Mick Seaman,
  Rev 1.5 11/7/99

all on the 802.3ad web site.

Agenda

- The Problem

- Opportunity for a solution

- How it works

- Alternative goals when constrained

- Recommendations required

- An opportunity

- Supporting changes and mechanisms

- Conclusions

The Problem

In this example both systems can aggregate up to 3 links

Dynamic key changes may have sub-optimal results

Dynamic key changes may have non-deterministic results

- Relative timing of key changes not specified

- Information progressively changed/lost

Current recommendations use at least one link

- Better than no recommendation at all!

Opportunity for a solution

Synchronization bit

- Already available to cope with delays and difficulties in attaching to/detaching from aggregator resources

- Extend use to cases where attaching/detaching may be permanently delayed

Why didn't we think of this before

One predecessor protocol added a Sync bit as a result of operational/test experience to accommodate hardware delays

Another predecessor protocol was constructed with more code points

How it works

Systems agree on a common prioritization of individual links to bring in sync if possible

- Use (port priorities +) port numbers of system with lowest System ID

Information persistent until one system changes a key

- System with lowest System ID may change the key

Alternative goals when constrained

Change keys to bring up separate aggregates?

• Good for equal cost load sharing by routers

Hold unusable links in hot standby?

• Good if alternative is Spanning Tree blocking

Recommendations required

Existing protocol supports behavior described

- So someone is going to use it

- For useful multi-vendor operation a common algorithm is required - otherwise aggregations of one may result

An Opportunity

LACP may be used to identify and activate "hot standby" links even if a system's hardware can only distribute and collect from one link at a time.

Supporting changes and mechanisms

Care when interpreting churn machine indications

• Good for equal cost load sharing by routers

Alternative goals when constrained

Change keys to bring up separate aggregates?

• Good for equal cost load sharing by routers

Hold unusable links in hot standby?

• Good if alternative is Spanning Tree blocking

Conclusions

No protocol changes

Explicitly allow and describe behavior

• include port priority (do we need system priority)

Dynamic key changes are needed less

• but should still be described, with different constraints on changes