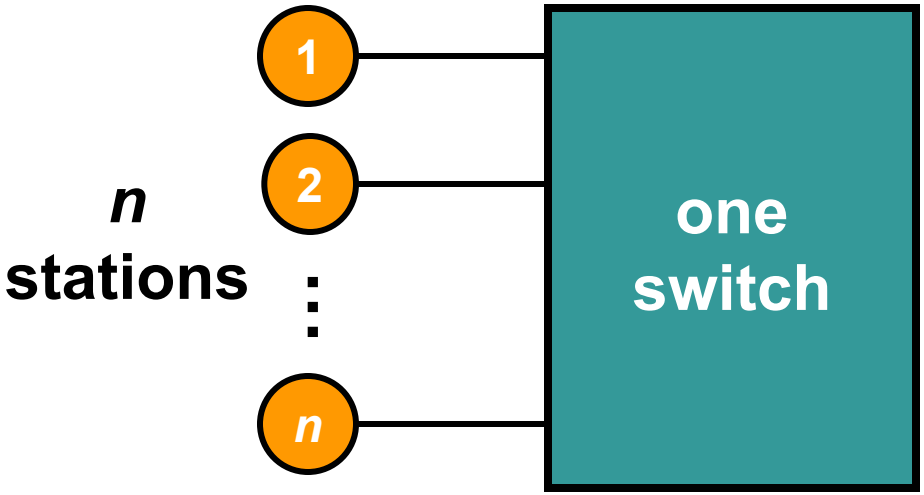
VLANs, Classes of Service, and Flows

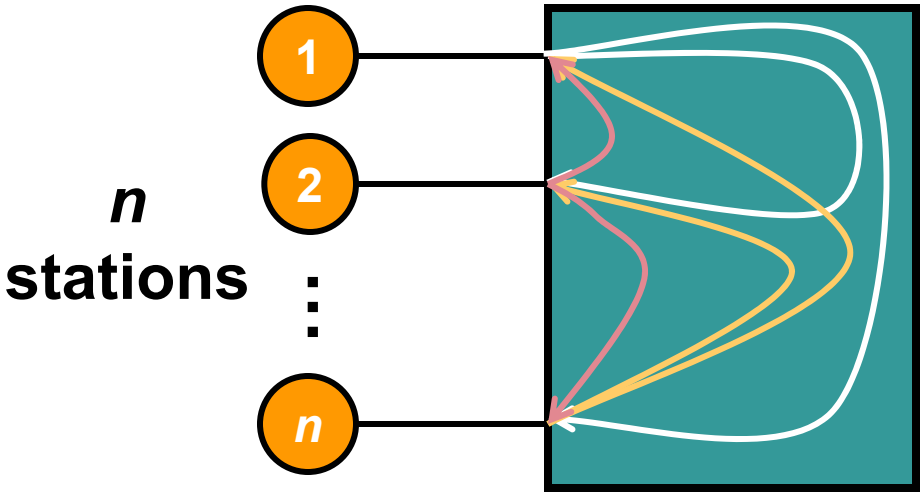
Norman Finn

Start with the proposed configuration



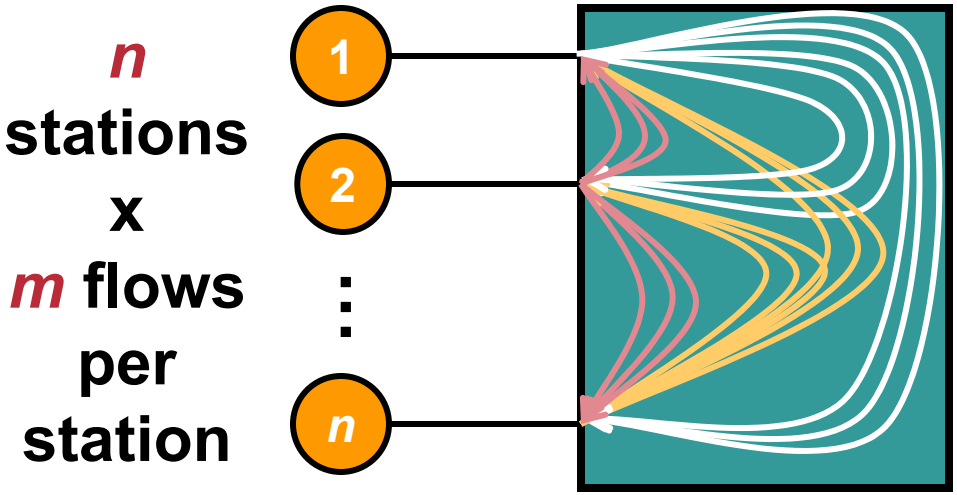
- One switch, *n* stations

On what granularity are we pausing?



- Are we pausing on a per-station basis?
- No. We can do that now, with the current Pause.

On what granularity are we pausing?



- So, we are pausing (*n* stations) • (*m* streams per station) = *M* streams.
- With small *m* and *n* this is feasible.

Per-Something Pause

Cisco.com

Per what?

- Could do per-Priority Pause.
- Could do per-VLAN Pause.
- Could do per-Flow Pause.

Per-Priority Pause

Compatibility with current devices must be maintained, else it is not 802.3!

- **Priority queues are already defined in 802.1D. They are perfectly applicable to stations, and in use in some.**
- **Better "drain" algorithms than simple priority order are in general use in bridges, and could be standardized.**
- **BUT, there are not probably enough of them to identify flows.**

Per-VLAN Pause

Compatibility with current devices must be maintained, else it is not 802.3!

Cisco.com

- VLANs already defined in 802.1Q.
- BUT, VLANs are broadcast domain identifiers, not flow identifiers.
- AND, per-VLAN queuing would have to be defined for stations.
- AND, it is difficult to see how per-VLAN station queues would interoperate with per-priority bridge queues, which are orthogonal to VLANs.

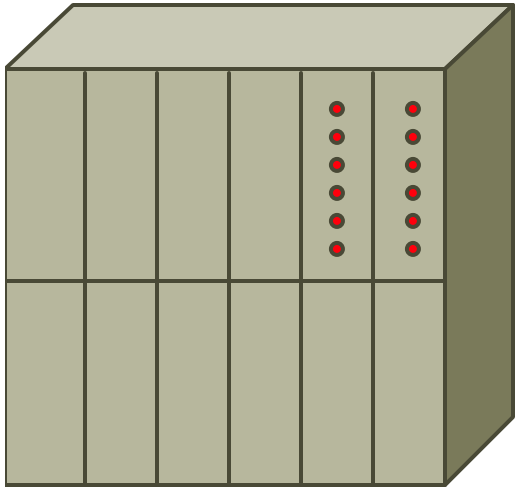
Per-Flow Pause

Compatibility with current devices must be maintained, else it is not 802.3!

Cisco.com

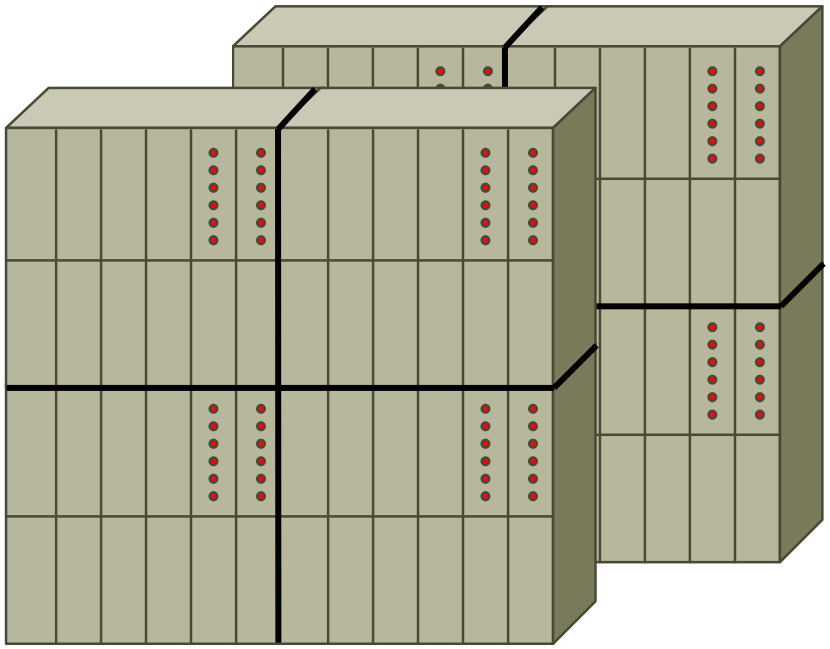
- **A "Flow" is a data stream from one process in one processor to another process in (for this case) another processor.**
- **Flows are undefined, today, but could be defined.**
- **BUT, how would one identify Flows? A new tag?**

Today



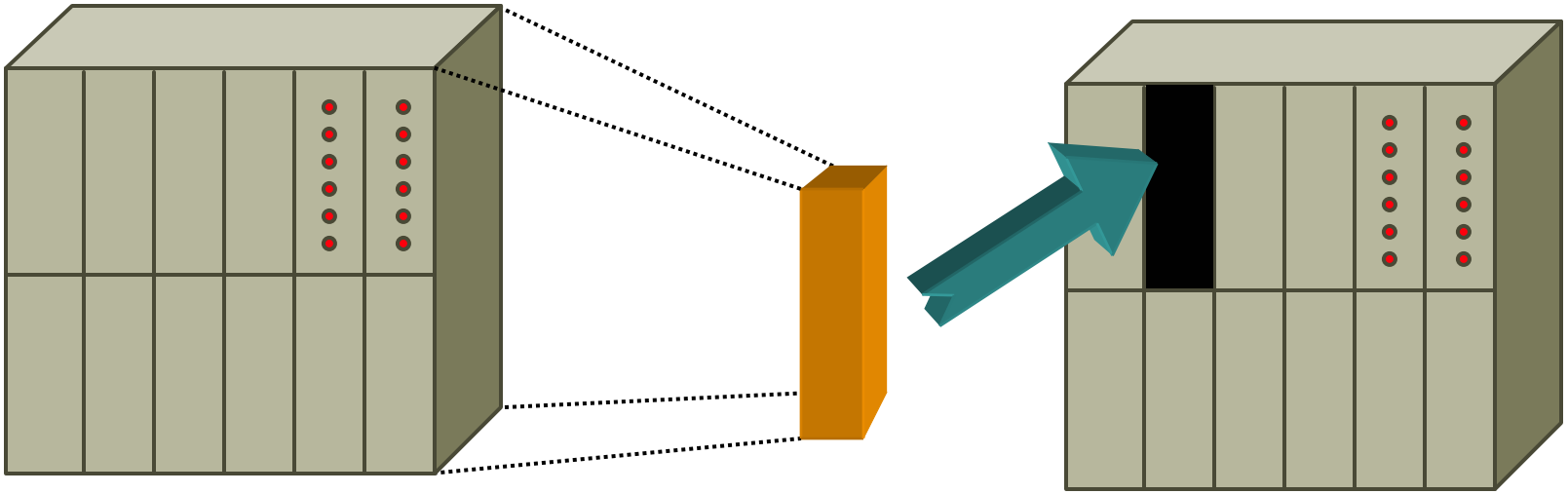
- **Several station cards connected to one switch card.**
- **One (or a very few?) processors per station card.**

What happens Tomorrow?



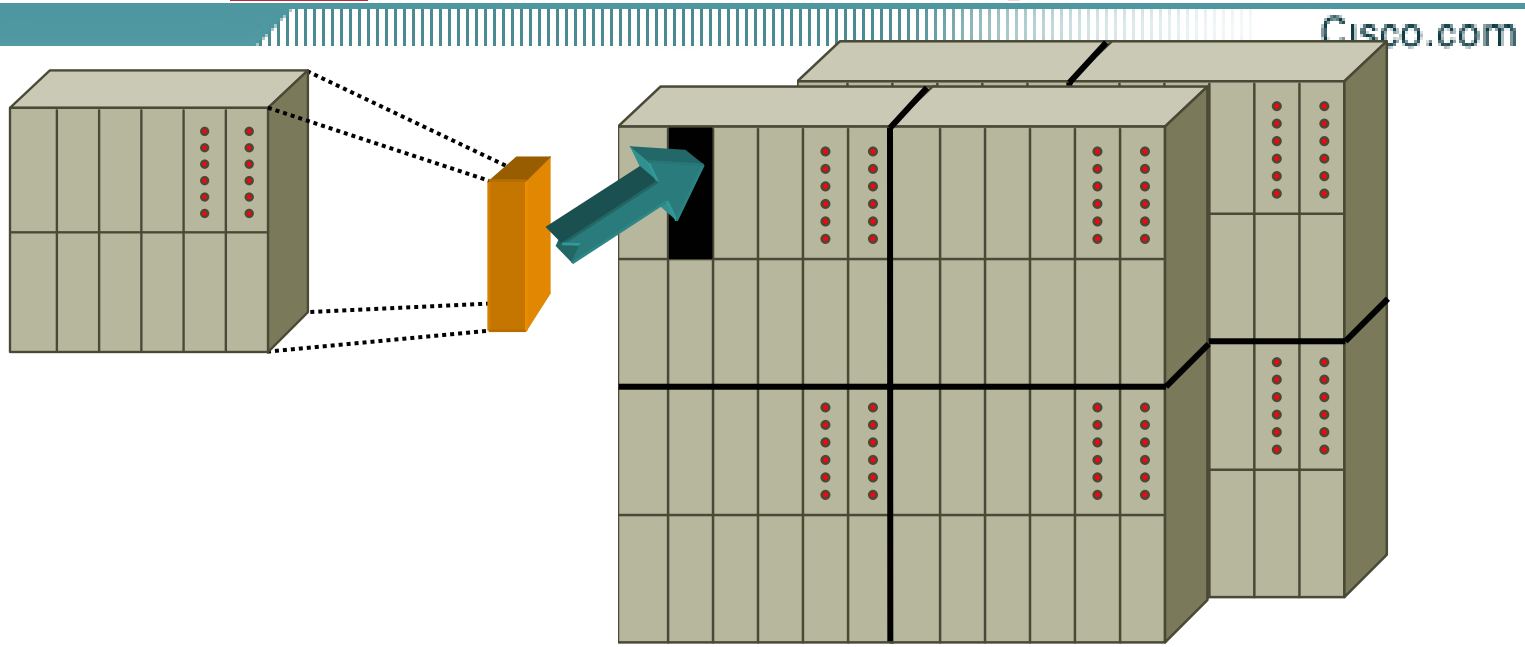
- Several chassis connected together in a rack.
- Maybe multiple racks?
- *n* • *m* goes up by a factor of 8? 100? 10,000?

What happens Next Week?



- **Today's chassis becomes tomorrows card.**
- **That card is put in a new chassis.**

That is, we will have to scale up!

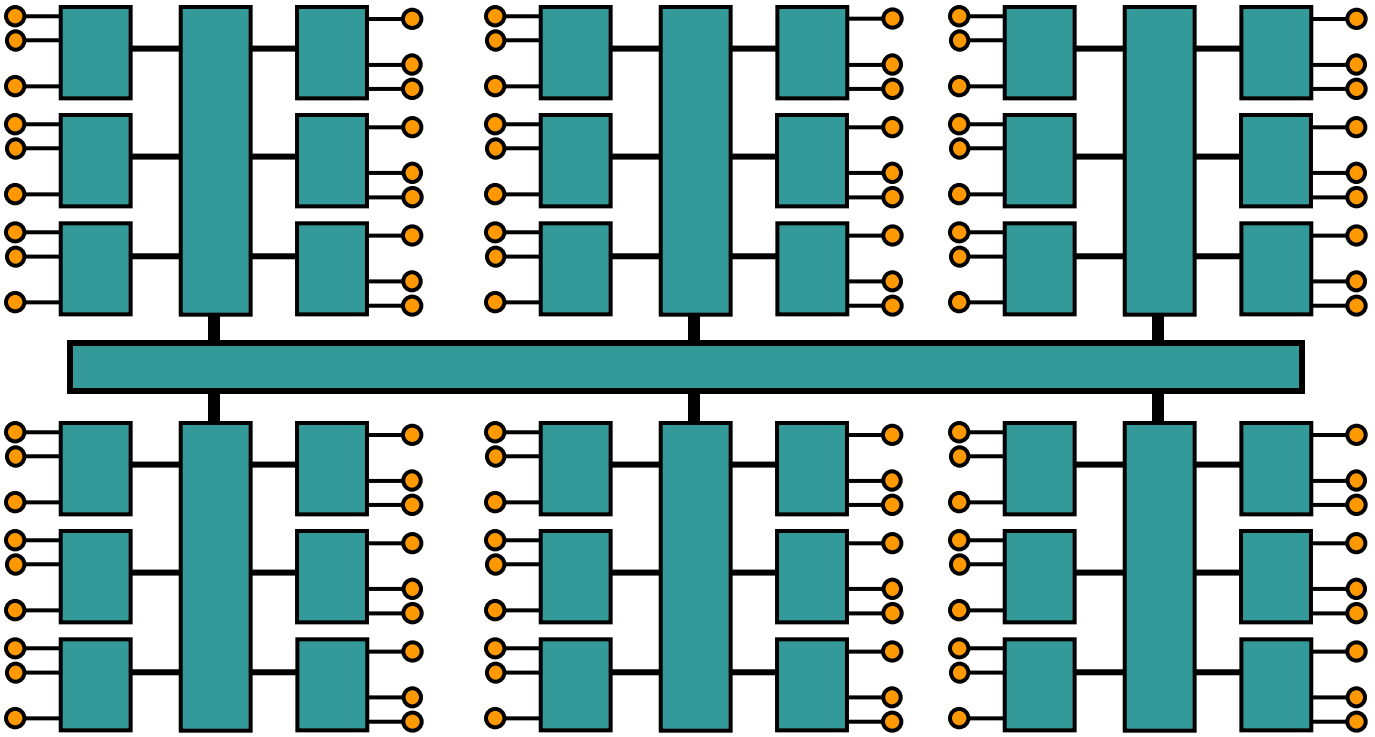


- The chassis's switch card becomes:

a card's switch chip.

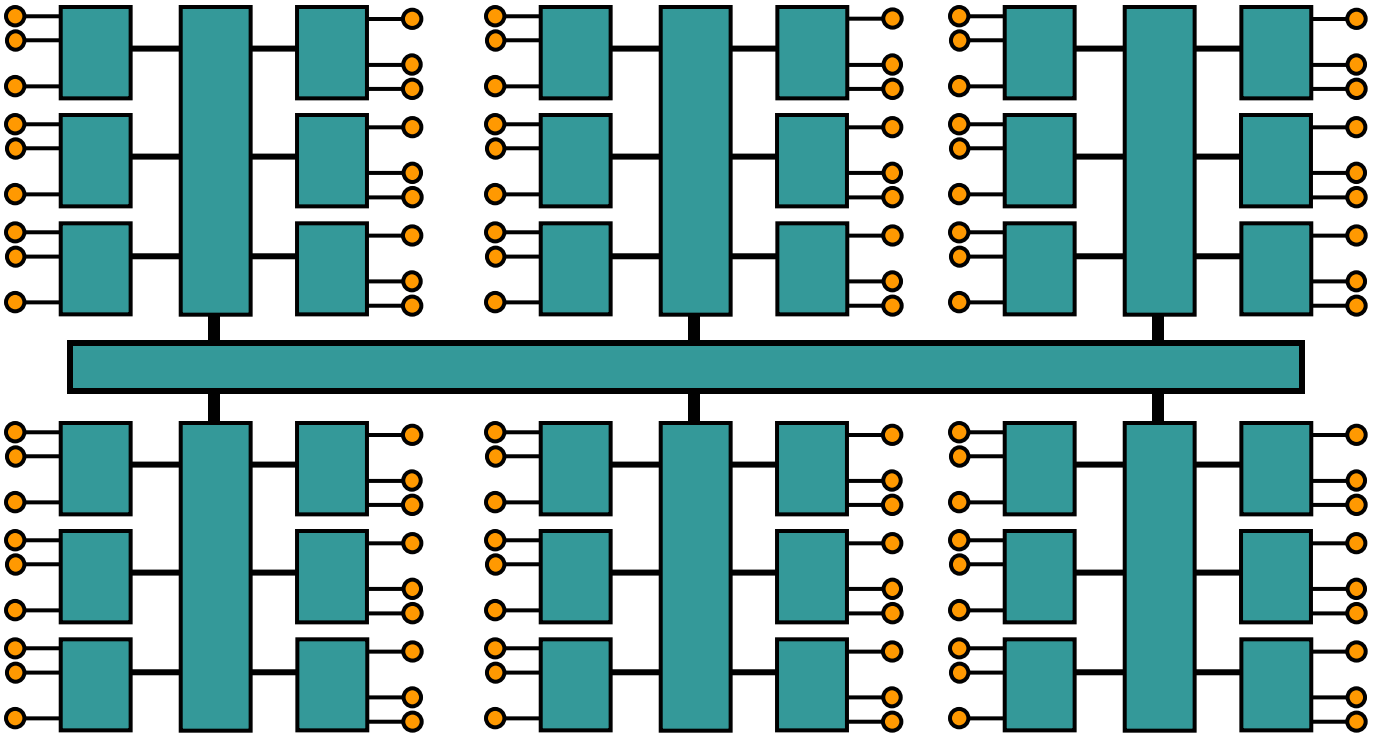
a rack's switch chassis.

And, when we scale up …



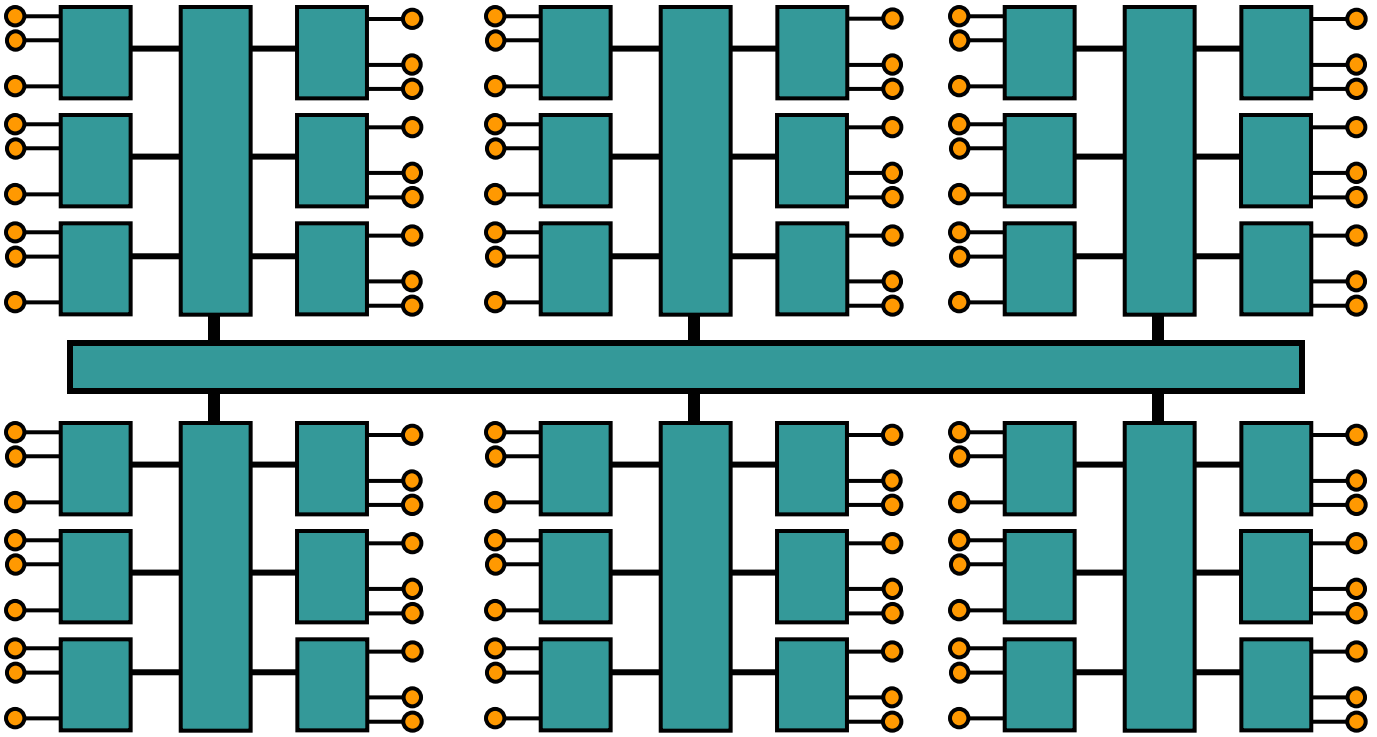
- *n* • *m* becomes very large.

And, when we scale up …



- There will no longer be only one switch between any two stations.

And, when we scale up …



- 8 Priorities will definitely not be sufficient for the number of flows per processor.

An impossible set of constraints

- The pressure to scale up is inevitable. The ability of 802.3 (via 802.1 bridges) to scale up is a large factor in its success.
- Holding to a one switch, no feedback from stations, limitation is not possible.
- Stations' protocol stacks must remain compatible with existing 802.3 stacks.
- Stations' MAC stacks must remain compatible with 802.1D/Q/ad bridges.

Not to mention

- 802.3 is not the place to talk about queues, much less a switch, even less a network of switches.

Is this a job for 802.3?

Summary:
The End-to-End Flow Control Problem

- Layer 2 solutions to EtEFC exist (ATM, Fibre Channel, others).
- Therefore, an IEEE 802 solution to EtEFC could certainly be developed.
- Such a solution should be at least as interoperable with 802.3 as FDDI, 802.5, or 802.11 have been and are.
- .3 is NOT the place to develop EtEFC.


CISCO SYSTEMS
EMPOWERING THE
INTERNET GENERATION