Gigabit and Terabit Switching

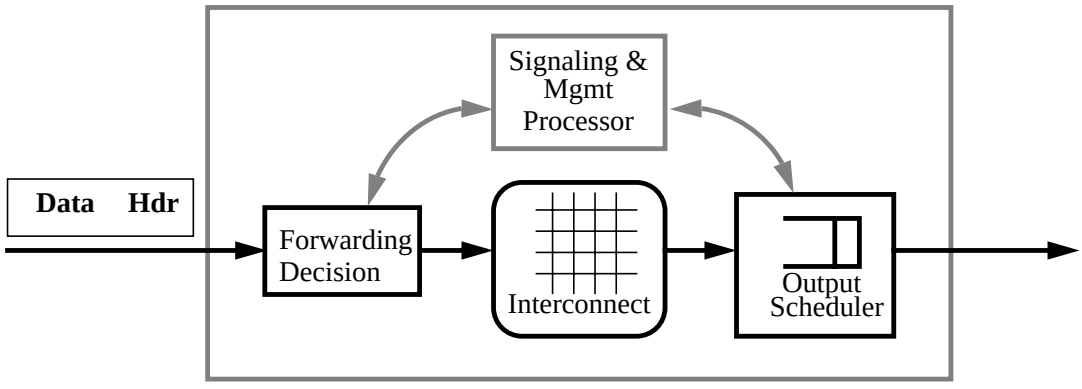
Gignet '97 — Europe: June 11, 1997.

Nick McKeown
Assistant Professor of Electrical Engineering and Computer Science

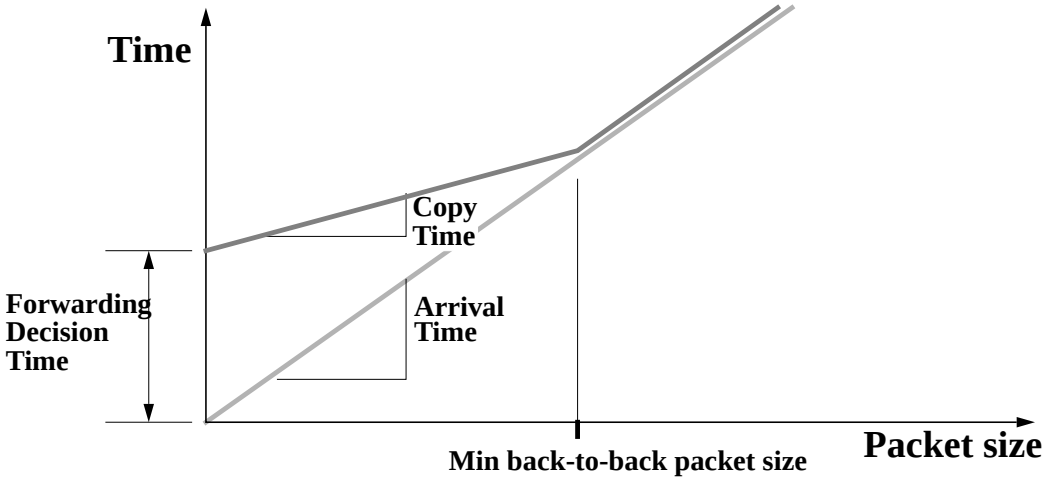
nickm@ee.stanford.edu
http://ee.stanford.edu/~nickm

The Architecture of Packet Processors

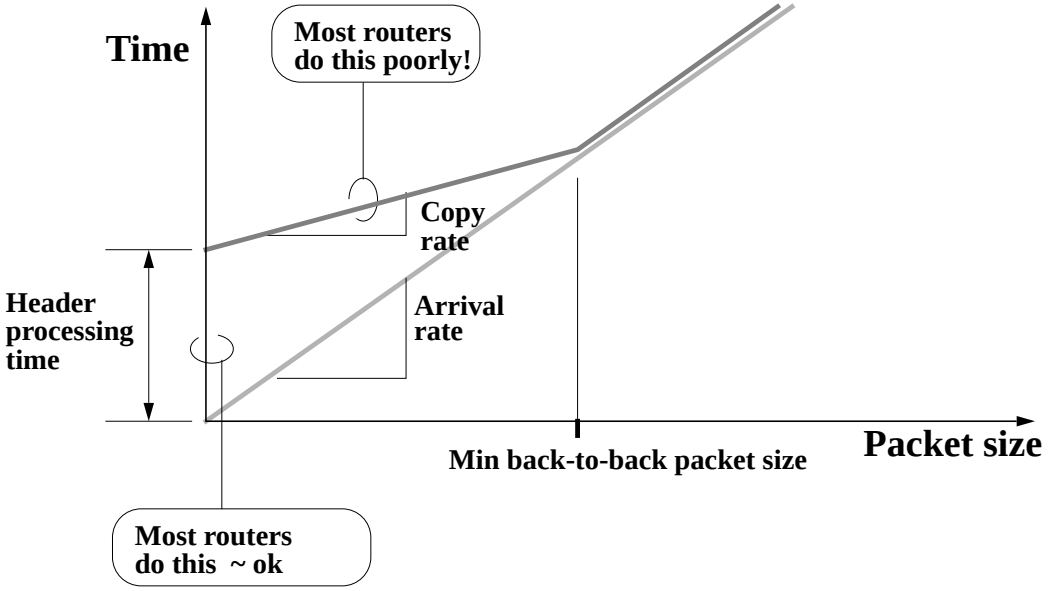
Generic Packet Processor:
(e.g. IP Router, ATM Switch, LAN Switch)



Performance of IP Routers



Performance of IP Routers



Gigabit and Terabit Switching

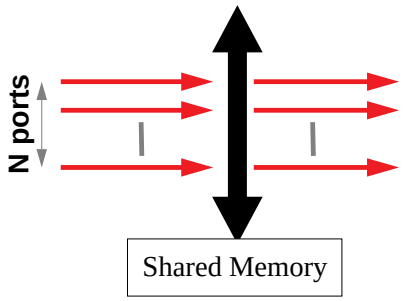
Switched Backplanes

- Input Queueing
  - — Theory
  - — Unicast
  - — Multicast
- Fast Buffering
- Speedup

An Example: *The Tiny Tera*

Should we use shared memory or input-queueing?

Shared Memory:



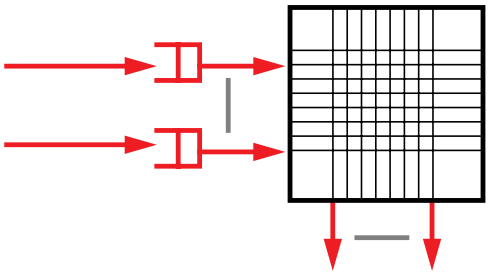
Advantages:

Highest Throughput.
Possible to control packet delay.

Disadvantages:

N-fold internal speed-up

Input Queueing:



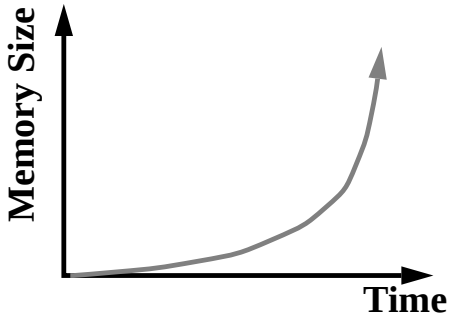
Advantages:

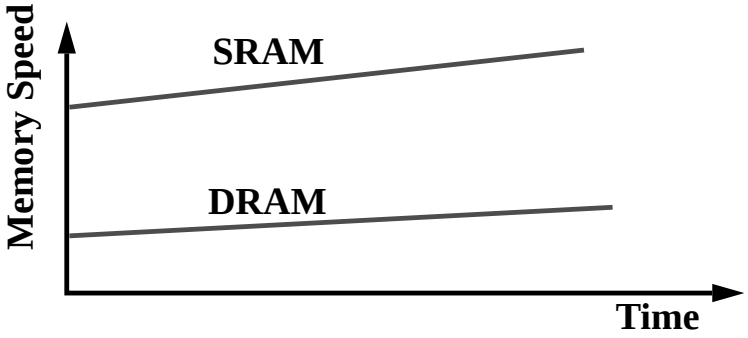
Simplicity
High Bandwidth

Disadvantages:

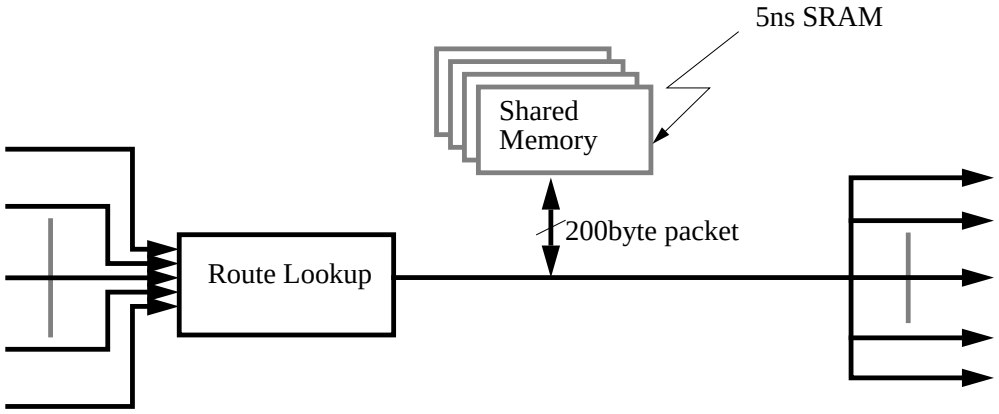
HOL Blocking
Less efficient
Difficult to control packet delay.

Memory Bandwidth





An aside: How fast can shared memory operate?



How fast can a 16 port switch run with this architecture?

5ns per packet $\times$ *2 memory operations per cell time*

$\Rightarrow$ *aggregate bandwidth is 160Gb/s*

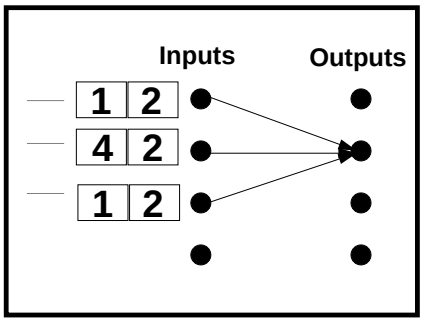
Should we use shared memory or input-queueing?

Because of a *shortage of memory bandwidth*, most multigigabit and terabit switches and routers use either:

1. Input Queueing, or
2. Combined Input and Output Queueing.

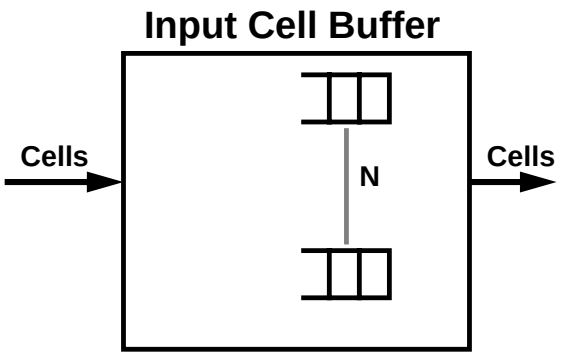
Head of Line Blocking

The Problem



$\rho_{max} = 2 - \sqrt{2} = 58\%$

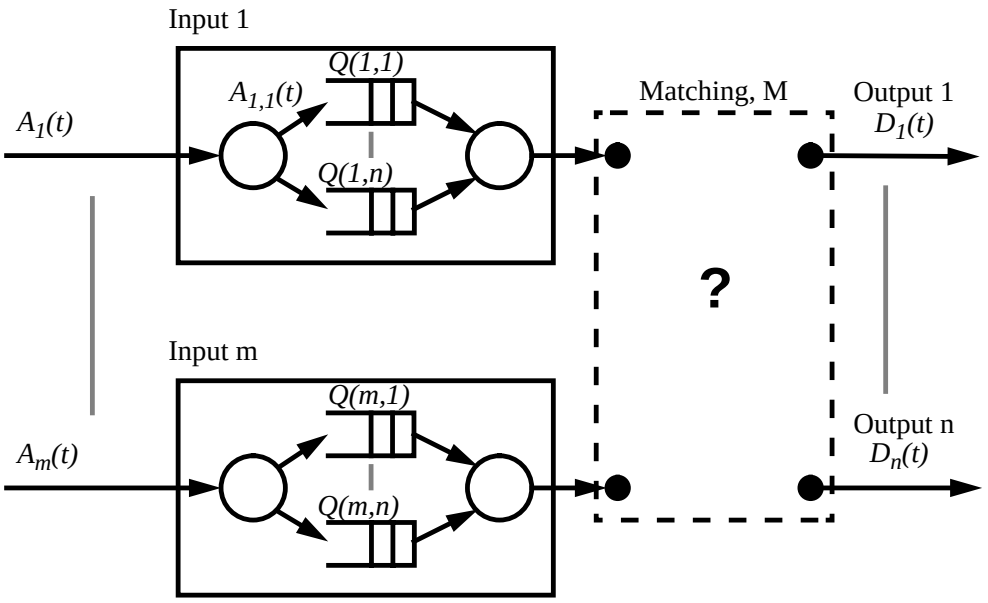
A Solution....



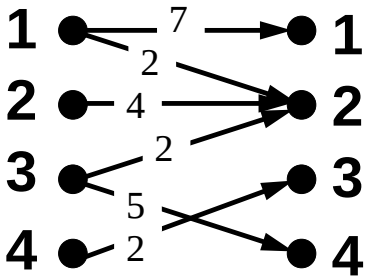
"Virtual Output Queueing"

$\rho_{max} = 100\%$

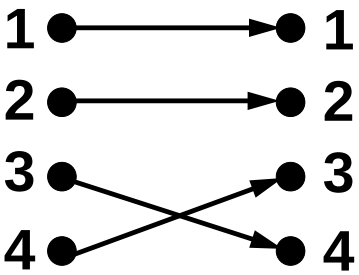
....but requires scheduling...



....which is equivalent to graph matching



Request Graph



Bipartite Matching

(Weight = 18)

Practical Algorithms

1. ***i*SLIP** — Weight = 1
   — Iterative round-robin
   — Simple to implement

   Simple, fast, efficient

2. ***i*LQF** — Weight = Occupancy

   Good for non-uniform traffic. Complex!

3. ***i*OCF** — Weight = Cell Age

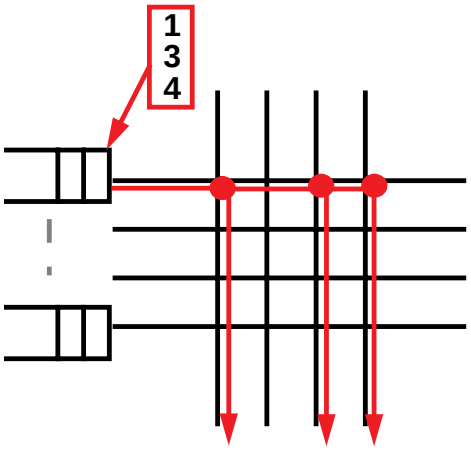
4. ***i*LPF** — Weight = Backlog

   Good for non-uniform traffic. Simple!

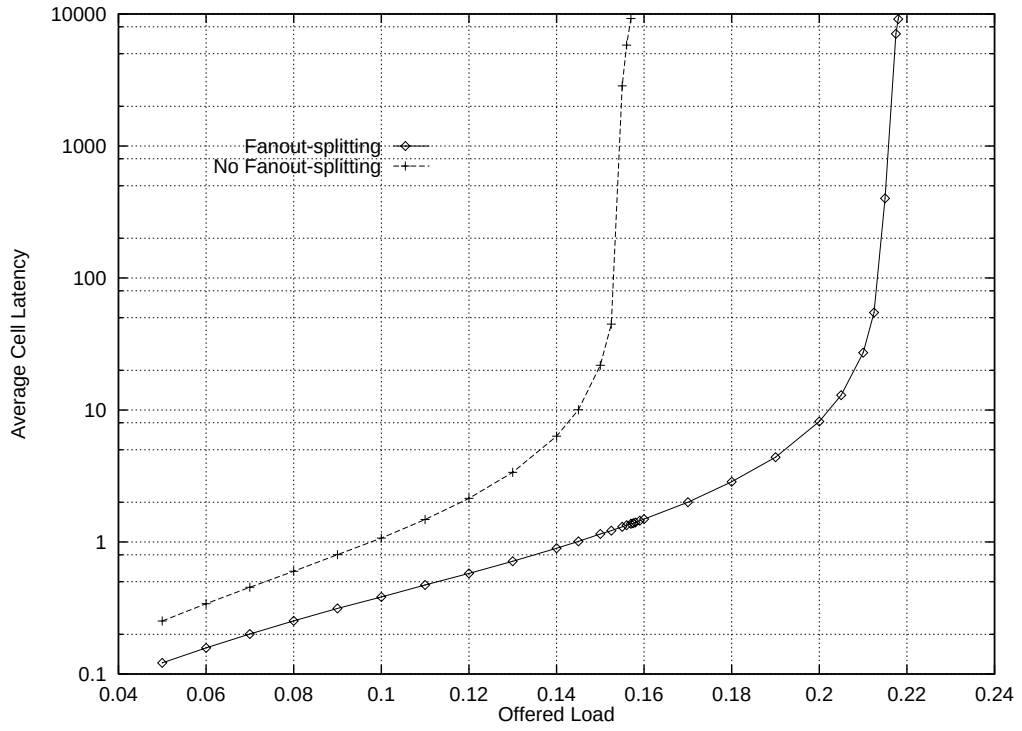
Multicast Traffic

Queue Architecture

1. Making use of the crossbar
2. Why treat multicast differently?
3. Why maintain a single FIFO queue?
4. Fanout-splitting



Fanout-Splitting



Multicast Traffic

1. ***Residue Concentration***

2. ***Tetris-based schedulers***

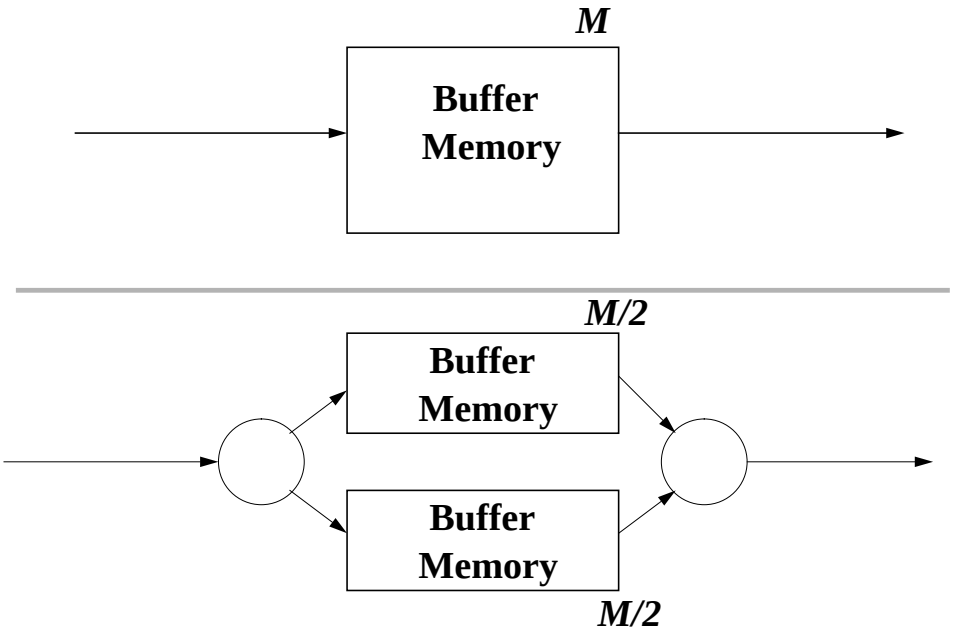
Gigabit and Terabit Routing

Switched Backplanes

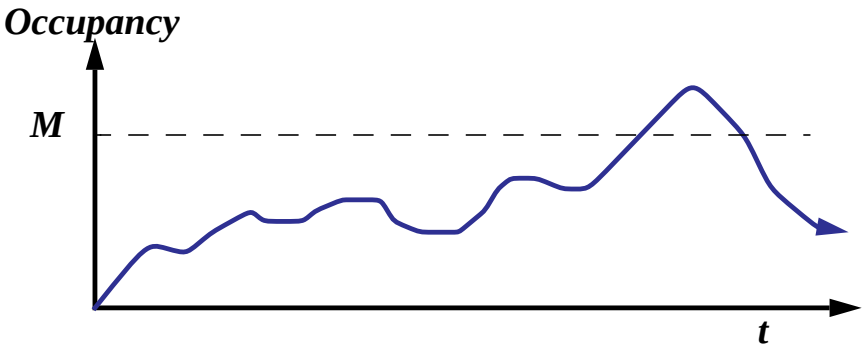
- Input Queueing
  - — Theory
  - — Unicast
  - — Multicast
- Fast Buffering
- Speedup

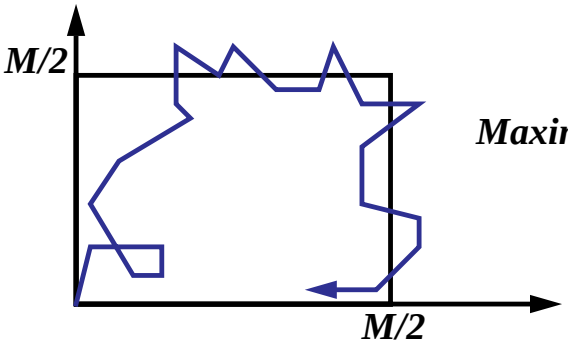
An Example: *The Tiny Tera*

Fast Buffering
Ping-pong Memory



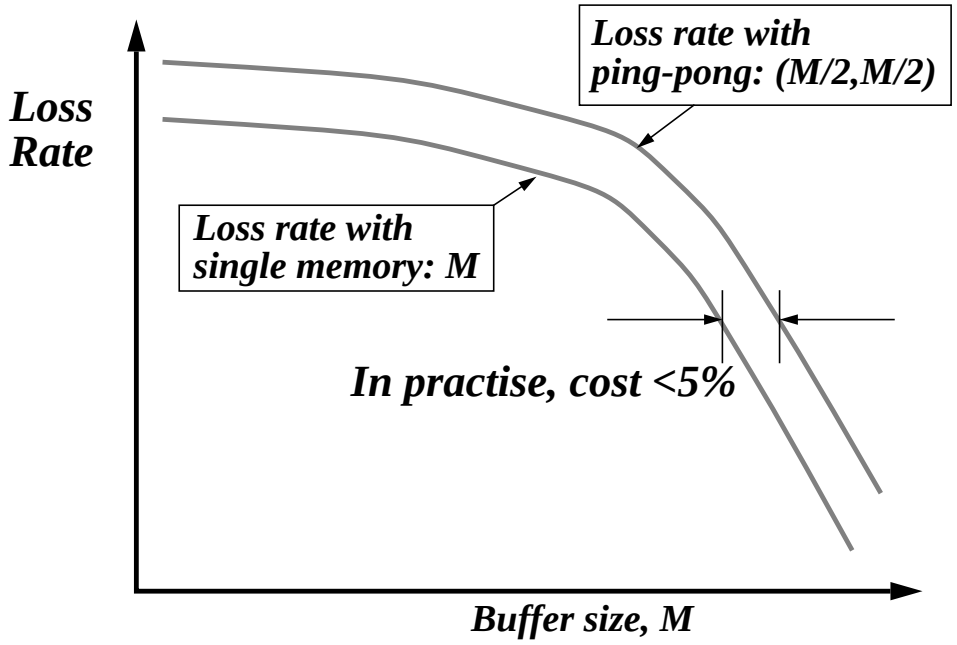
Fast Buffering
Ping-pong Memory





Maximum "cost" = M/2

Fast Buffering
Ping-pong Memory



Gigabit and Terabit Routing

Switched Backplanes

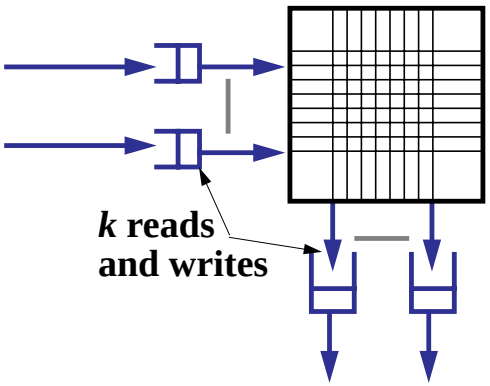
- Input Queueing
  - — Theory
  - — Unicast
  - — Multicast
- Fast Buffering
- Speedup

An Example: *The Tiny Tera*

Matching Output Queueing with Input- and Output- Queueing

How much speedup is enough?

Combined Input- and Output-Queueing:



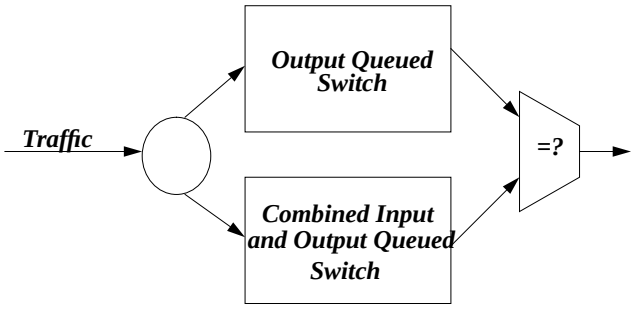
Matching Output Queueing with Input- and Output- Queueing

How much speedup is enough?

Conventional wisdom suggests:

A speedup $k \geq 2$ ***leads to high throughput***

Matching Output Queueing with Input- and Output- Queueing



Fact ***To match output queueing, with FIFO input queues:***
k = N

Fact ***To match output queueing, with virtual output queues:***
k = 2

Improving the Performance of IP Routers

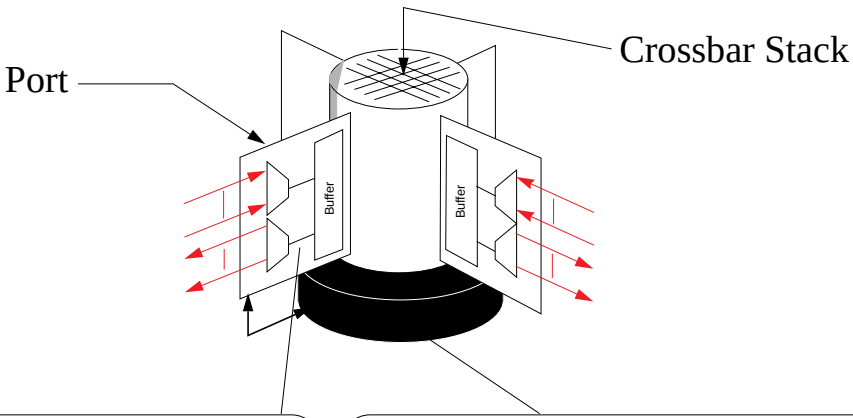
Switched Backplanes

- Input Queueing
  - — Theory
  - — Unicast
  - — Multicast
- Fast Buffering
- Speedup

An Example: *The Tiny Tera*

The Tiny Tera

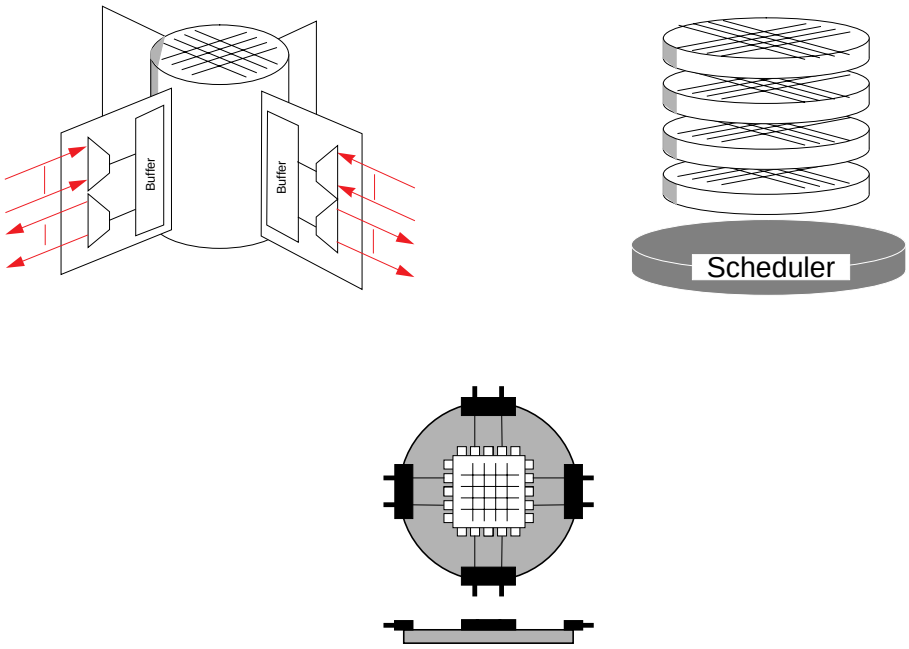
"Soda can" switch core
32x32 switch, ~16Gbps per port
Aggregate bandwidth: 0.5Tbps



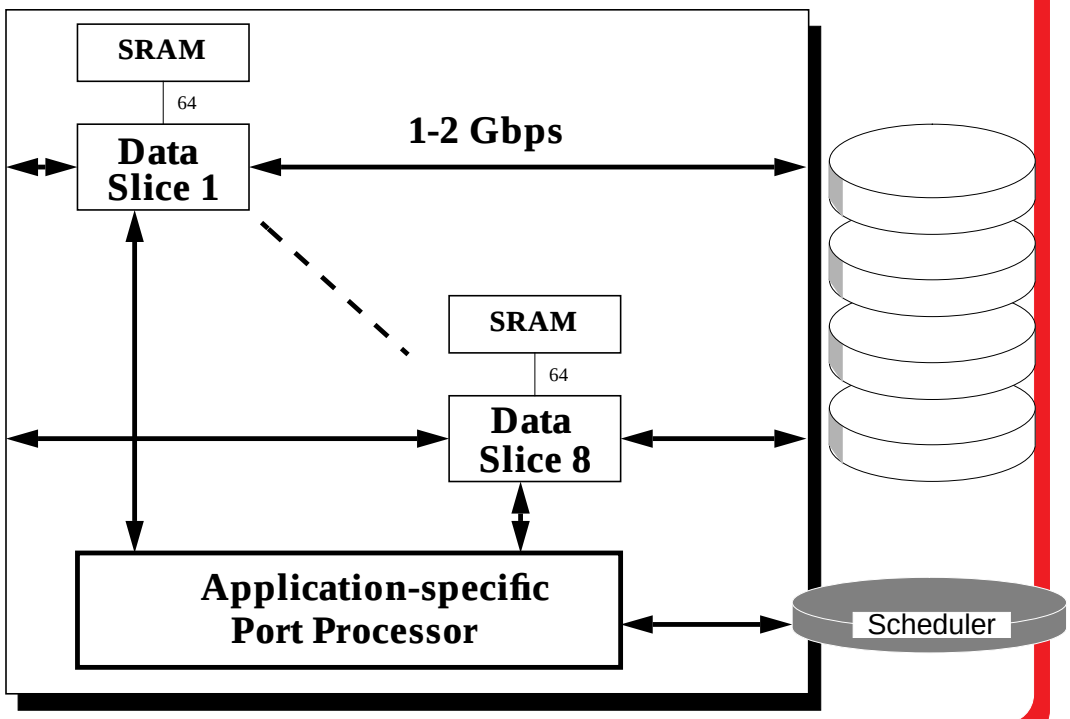
Requires high speed chip-to-chip links.

Schedulers must be fast, fair and efficient.

High Bandwidth Parallel Datapath

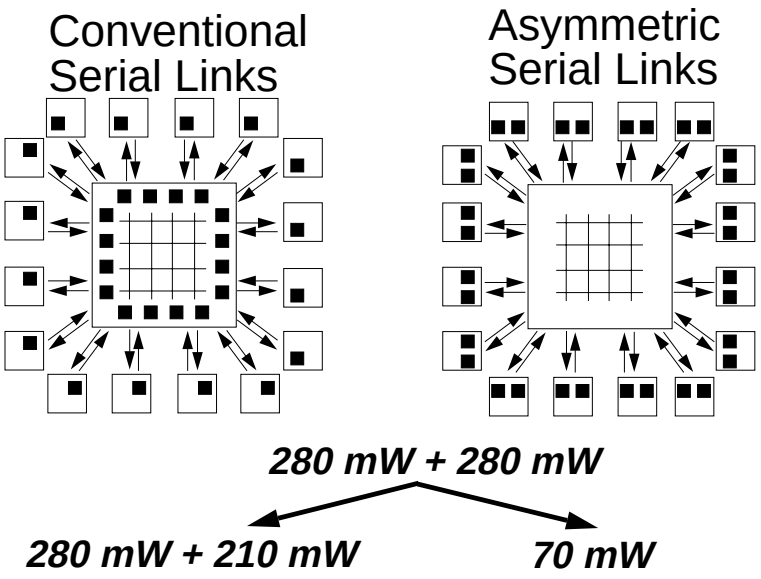


The Tiny Tera Port Architecture

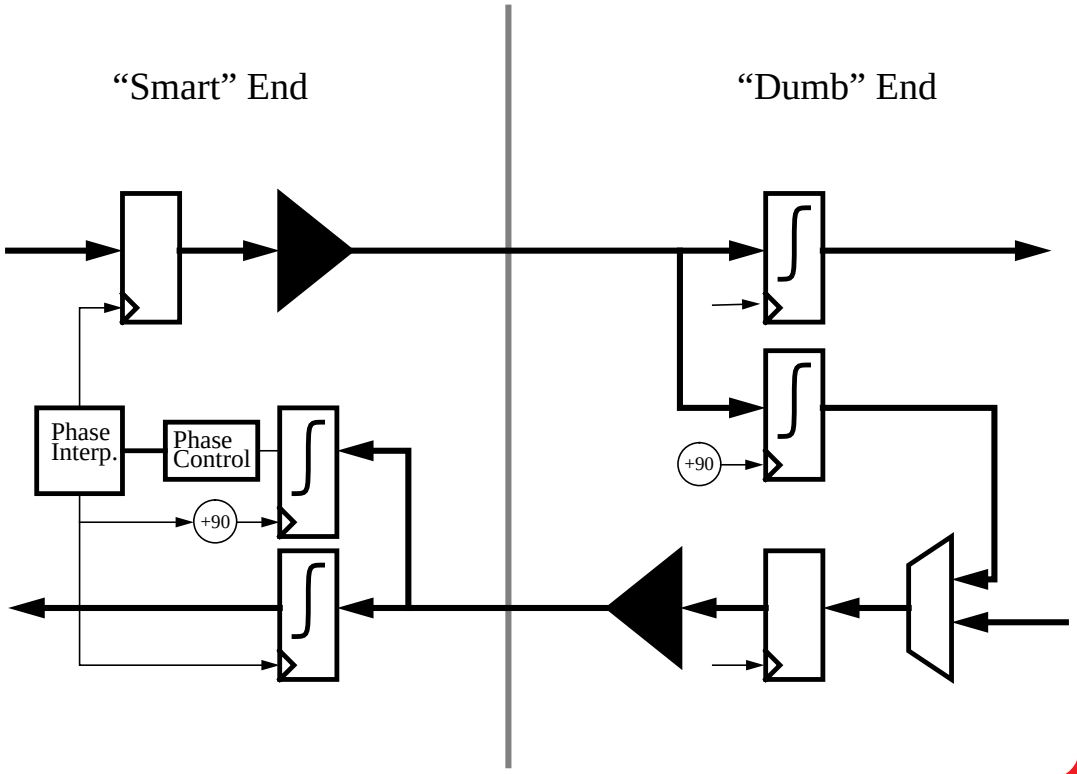


High-Speed Serial Links

2 Gbps with currently available CMOS technology.



Asymmetric High Speed Serial Links



The Tiny Tera

http://tiny-tera.stanford.edu/tiny-tera/

32 ports, 16 Gb/s per port.
***Input*-queued architecture.**
High bandwidth *parallel* datapath.
Efficient unicast *and* multicast.
Four priority levels.
Fixed *and* variable length packets.
***Asymmetric* high-speed serial links.**

'The fastest damn switch we can build...'

Gigabit and Terabit Switching

Switched Backplanes

- Input Queueing
  - — Theory
  - — Unicast
  - — Multicast
- Fast Buffering
- Speedup

An Example: *The Tiny Tera*