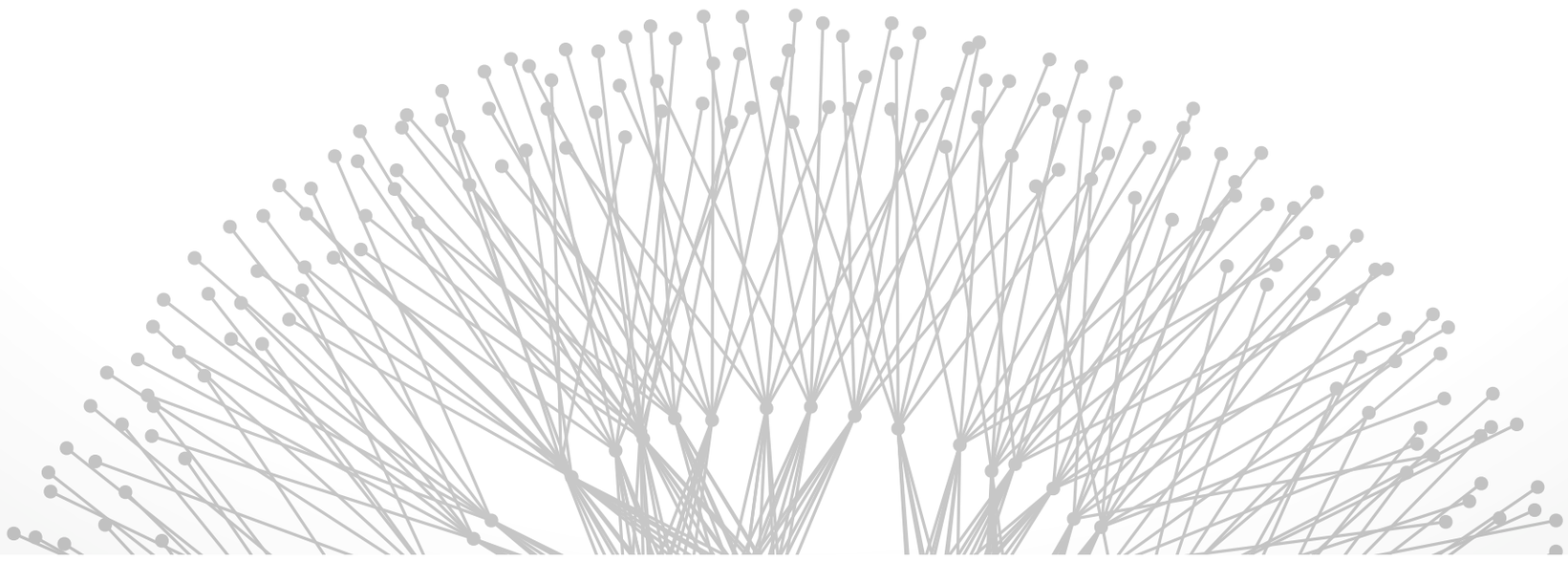
Forwarding Architecture

Brighten Godfrey

CS 538 September 18 2012



Building a fast router

Partridge: 50 Gb/sec router

A fast IP router

Good exhibition of the guts of a router and problems to be solved in router architecture

Routing vs. forwarding

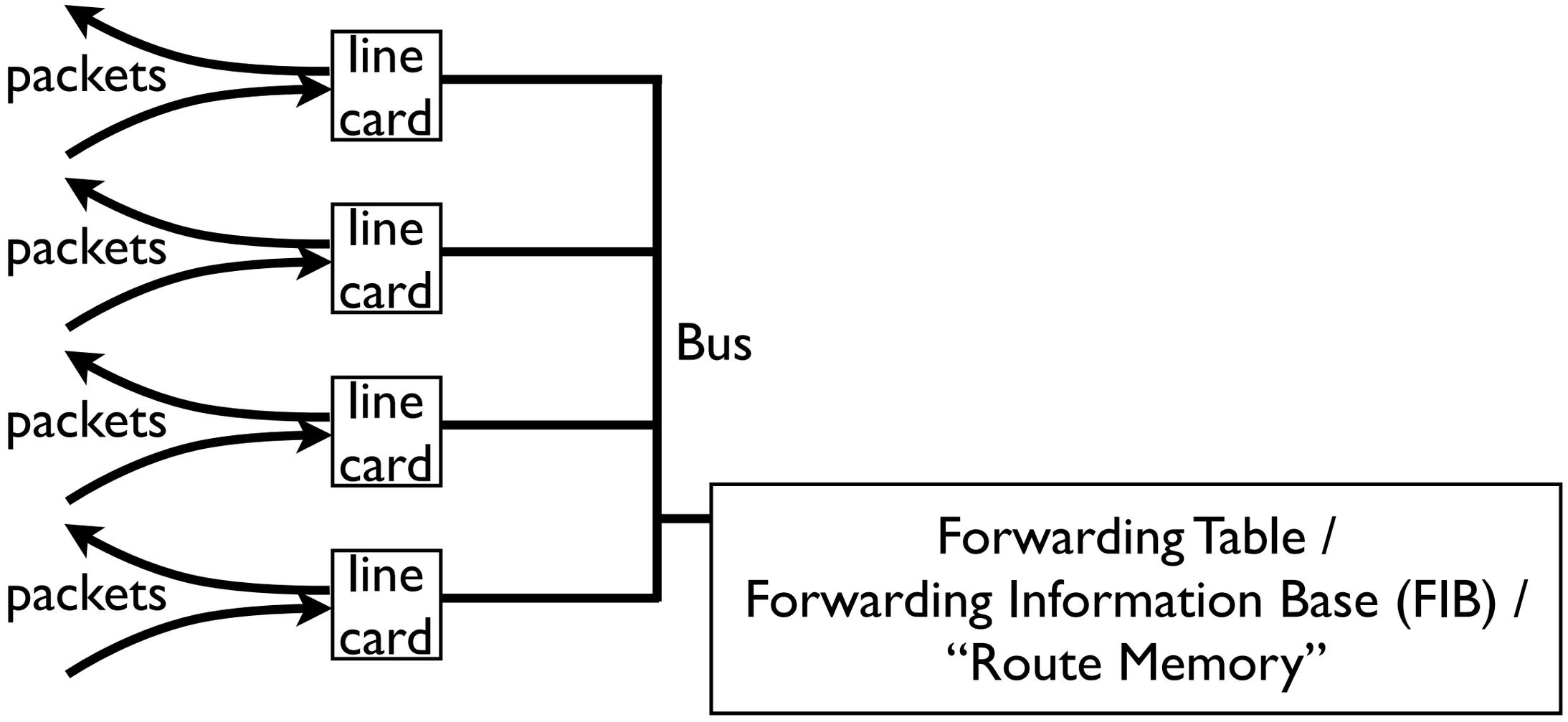
Control plane ⟵ Upcoming lectures

- Decides how data should flow through the network
- Uses operator configuration & distributed routing protocols
- Output: Forwarding table
  - Given a packet header, can find instructions for what to do with it
  - Relatively simple data structure (lookup table, tree, ...)

Data plane ⟵ Today

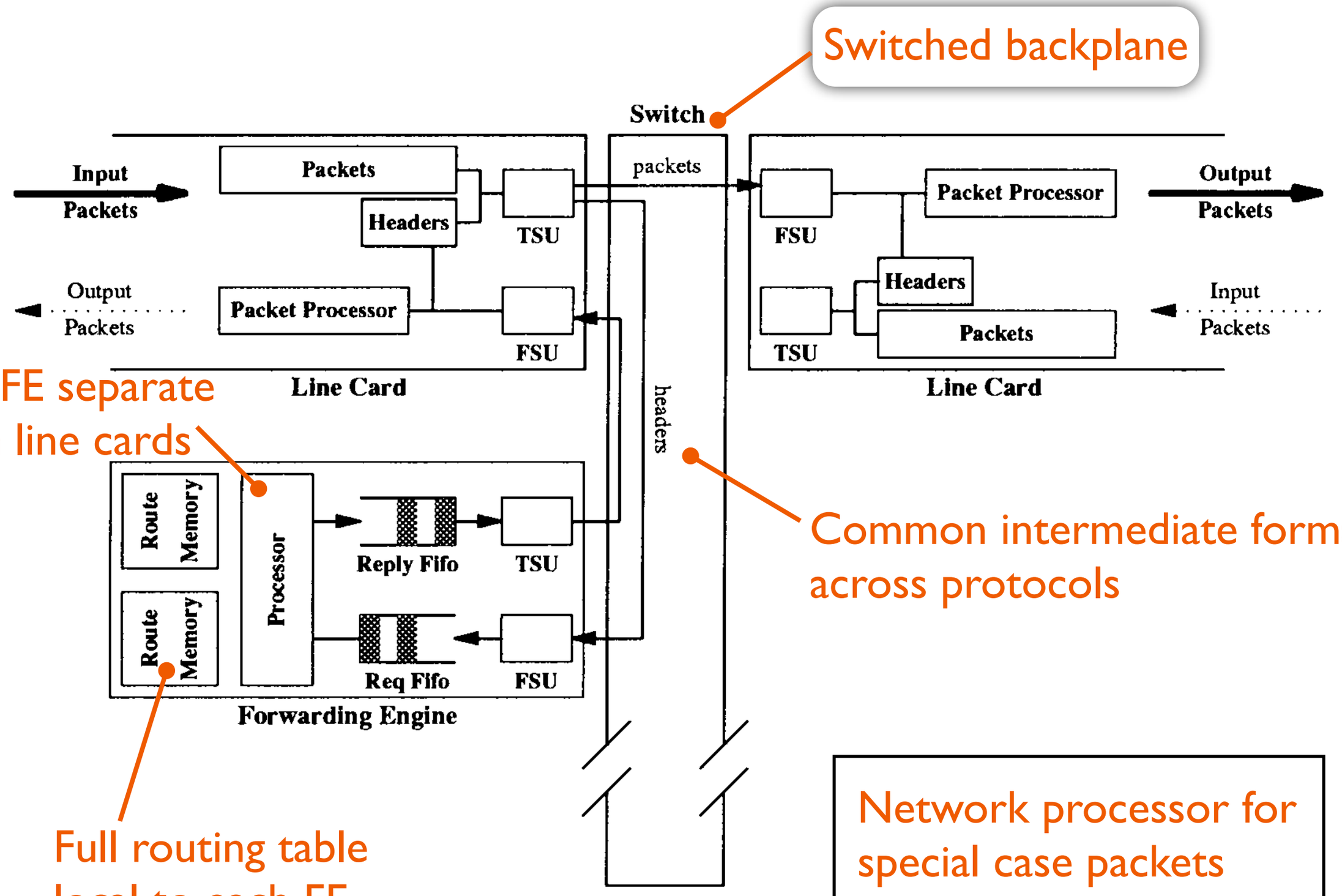
- Forwards data through the network
- Input: Forwarding table

Simplistic router diagram



What's wrong with this picture?

Inside the router



Fast FE separate from line cards

Common intermediate format across protocols

Full routing table local to each FE

Network processor for special case packets

[Partridge et al '98]

Switching fabric

Operates in epochs

- 128 bytes sent by each line card to next-hop line card
- Each line card can send to only one other card, and can receive from only one other card

Inputs ready to send...	... to outputs						
	1	2	3	4	5	6	7
1	0	0	1	0	1	0	0
2	1	1	1	1	0	0	0
3	0	1	1	1	0	0	1
4	1	0	1	0	0	1	0
5	0	0	0	0	0	0	0
6	0	0	0	0	1	0	1
7	0	1	1	0	0	1	0

Allocator assigns inputs to outputs & tells line cards

What fundamental problem is being solved?

Maximum bipartite matching

Maximize number of matched input-output pairs, such that each input & output only matched once

50 Gbit/s router uses approximate solution

Inputs ready to send...

... to outputs

	1	2	3	4	5	6	7
1	0	0	1	0	1	0	0
2	1	1	1	1	0	0	0
3	0	1	1	1	0	0	1
4	1	0	1	0	0	1	0
5	0	0	0	0	0	0	0
6	0	0	0	0	1	0	1
7	0	1	1	0	0	1	0

Beyond the 50 Gb/sec router

Many more problems to solve

Buffering (how big?)

Queueing (where? what order?)

FIB memory (how fast, large, and expensive?)

FIB algorithms (what data structure? how many memory accesses?)

Specialized functionality

Specialized functionality

Modern routers are jacks-of-all-trades:

- load balance across links
- access control
- filtering attacks
- quality of service
- accounting, traffic metering
- virtual private networks
- Protocol support: IPv4, IPv6, MPLS, ethernet, ...
- ...

Can we make forwarding flexible, extensible?

Data plane flexibility over the ages

1990s: Software routers

- Zebra, later Quagga, Click, Xorp, Open vSwitch

1997: Label switching (MPLS)

- Set up explicit paths for classes of packets

1999: Active networks

- Packet header carries (pointer to) program code

2008: Software Defined Networks

- Open interface to data plane, programmed by software controller

Scaling software routers

Efficiency vs. extensibility

Hardware routers

- Fast
- Specific functionality
- Result: many physical devices (routers, firewalls, intrusion detection, ...)

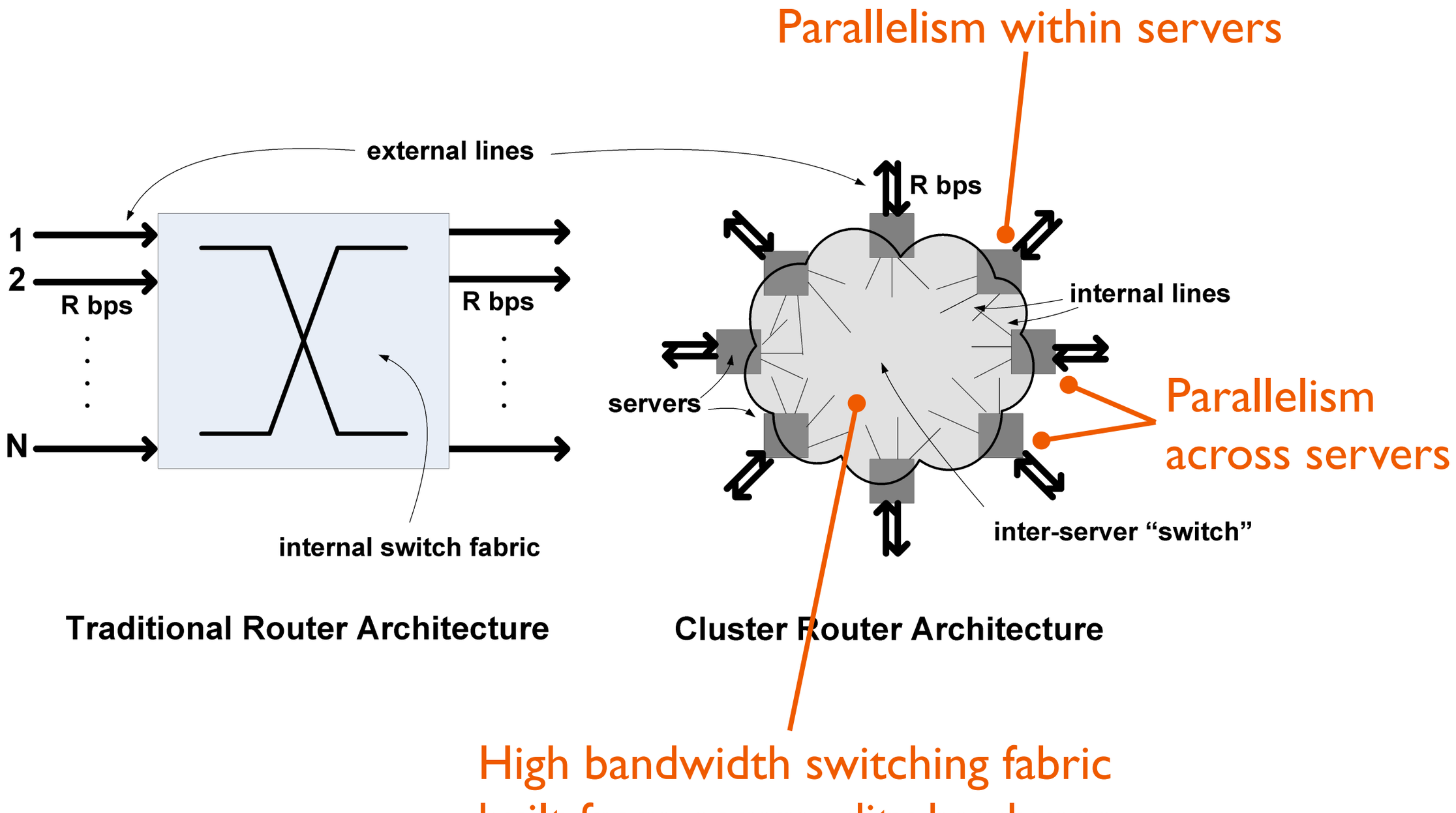
Software routers

- Slow
- Extensible

Can we get the best of both worlds?

RouteBricks approach

[Dobrescu, Egi, Argyraki, Chun, Fall, Iannaccone, Knies, Manesh, Ratnasamy, NSDI 2009]

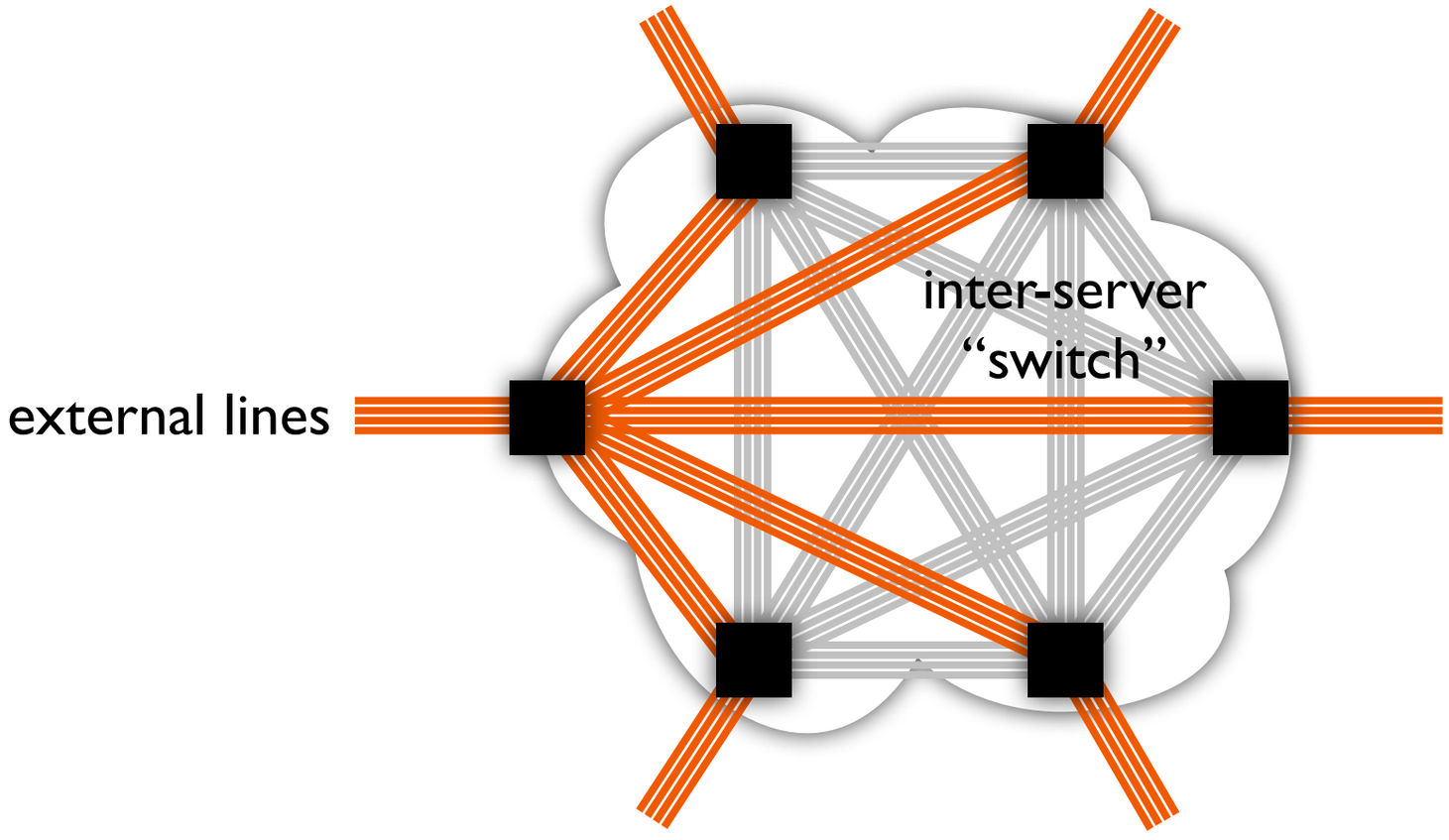


Switching fabric challenges

Handle any traffic pattern: for example, all input traffic at a server might go to any one output server

Low degree: we're using commodity hardware

Naïve approach:



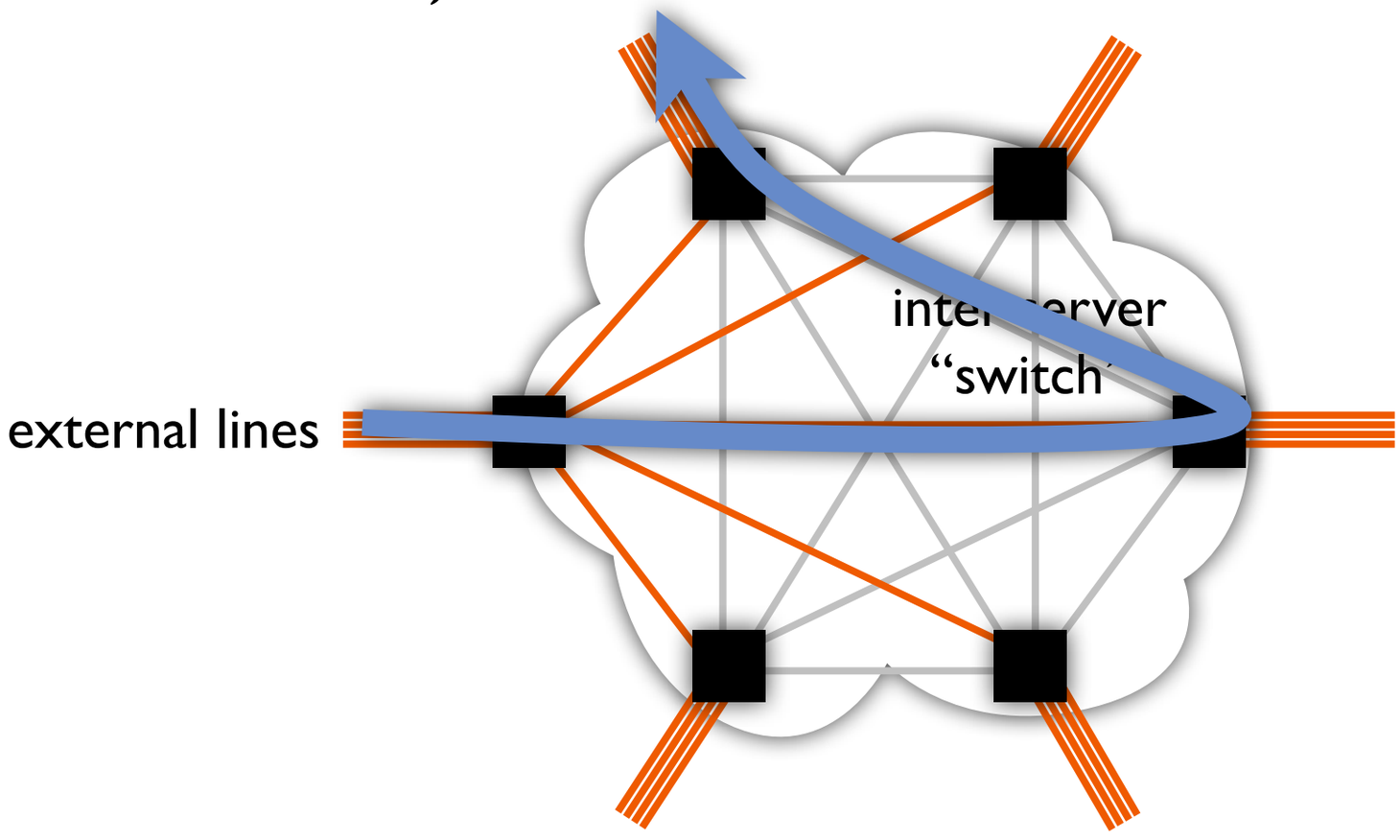
Useless: might as well just use one server!

Low degree solution

Just one link out for each link in

Total out b/w enough, but doesn't go where we need

Solution (Valiant load balancing): send packet to random intermediate node, then on to destination



VLB guarantees & questions

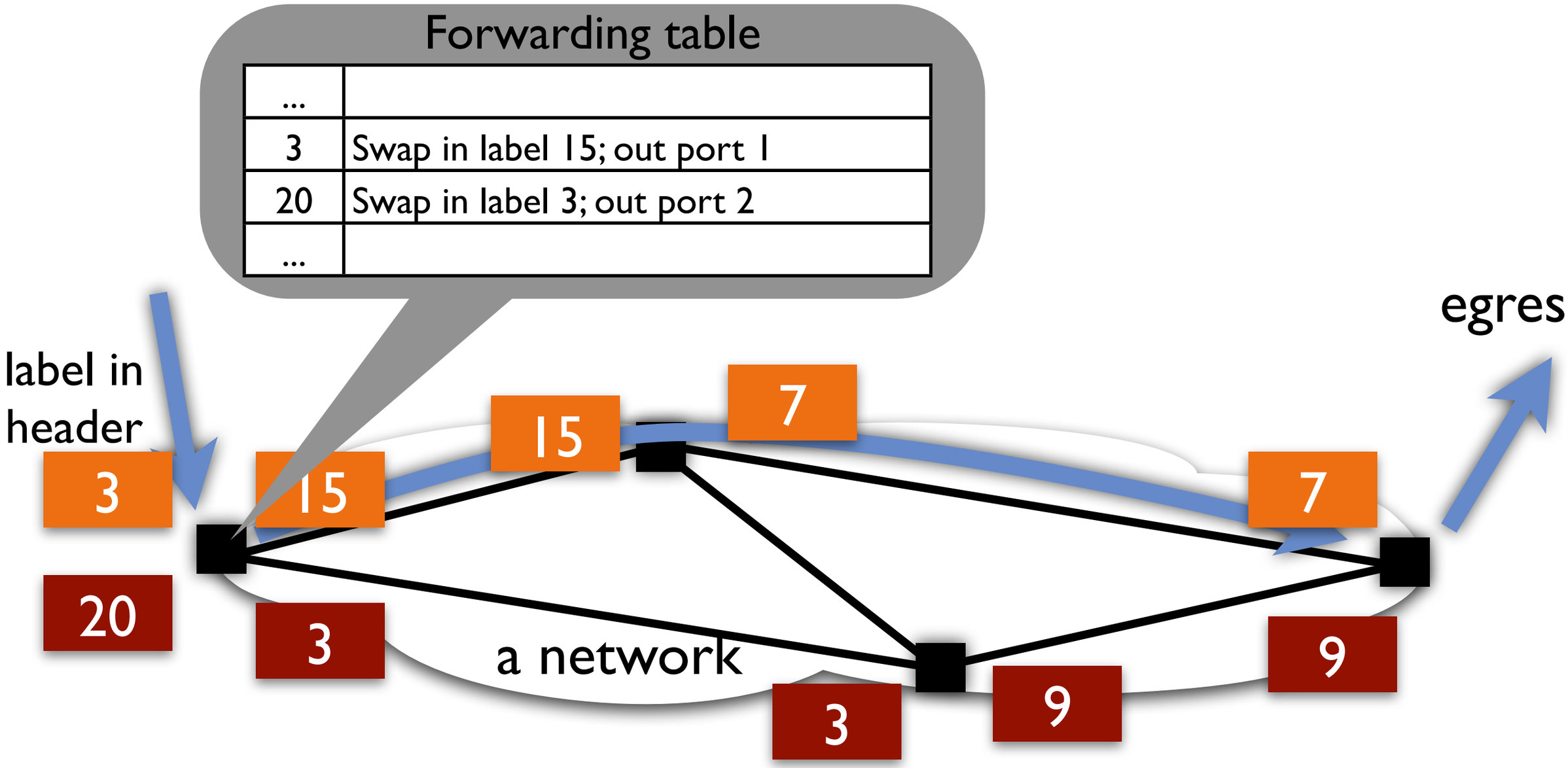
Guaranteed to nearly full throughput for any traffic demands!

- "nearly" = 2x. Why?
- So, switch fabric needs to be 2x as fast as external links to provide guarantees

Why does sending to a random intermediate node work?

MPLS

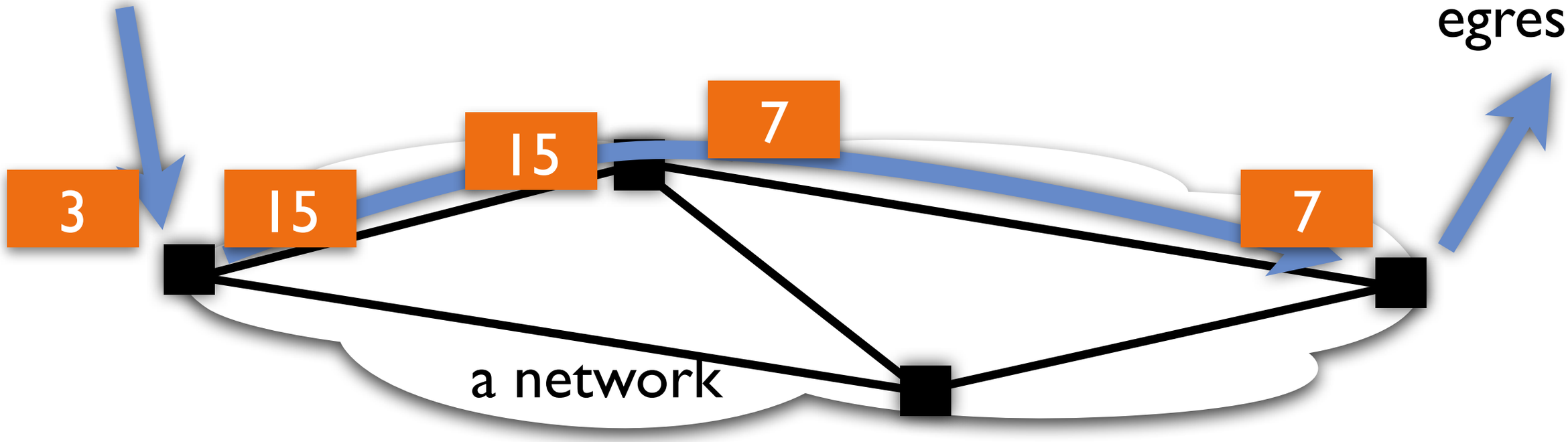
MPLS design



Why is this more flexible than shortest path routing?

MPLS design

Ingress:
Traffic classification, label packets ("forwarding equivalence class")



Control plane constructs label-switched paths and coordinates labels

Can also stack labels = concatenate paths

MPLS motivation

In the design doc

- High performance forwarding
- Minimal forwarding requirements, so can interface well with many types of media such as ATM
- Flexible control of traffic routing

What matters today?

Flexibility. Widely used to achieve:

- Virtual Private Network (VPN) service along dedicated paths between enterprise sites
- Control backup paths with MPLS Fast ReRoute
- Traffic engineering (load balancing)

Announcements

Today: Assignment 1 due

Thursday: Intradomain routing

Next Tuesday

- Project proposals due
- Proposal format in syllabus
- Need a partner? Check Piazza or hang around after class
  - https://piazza.com/class#fall2012/cs538/7

BIG DATA & MACHINE LEARNING

AT NEUSTAR

Join Neustar Chief Technology Officer, **Dr. Mark Bregman,** as he discusses Big Data and Machine Learning at Neustar. He will also discuss how Data Analytics will help make the Internet more efficient, IP Geolocation more accurate, and marketing campaigns more profitable.

Location:
1404 Siebel Center

Time:
September 20th
6:00–7:30pm