

Network Layer Introduction

Tom Kelliher, CS 325

Apr. 14, 2008

1 Administrivia

Announcements

Assignment

Read 4.4.

From Last Time

Congestion Control.

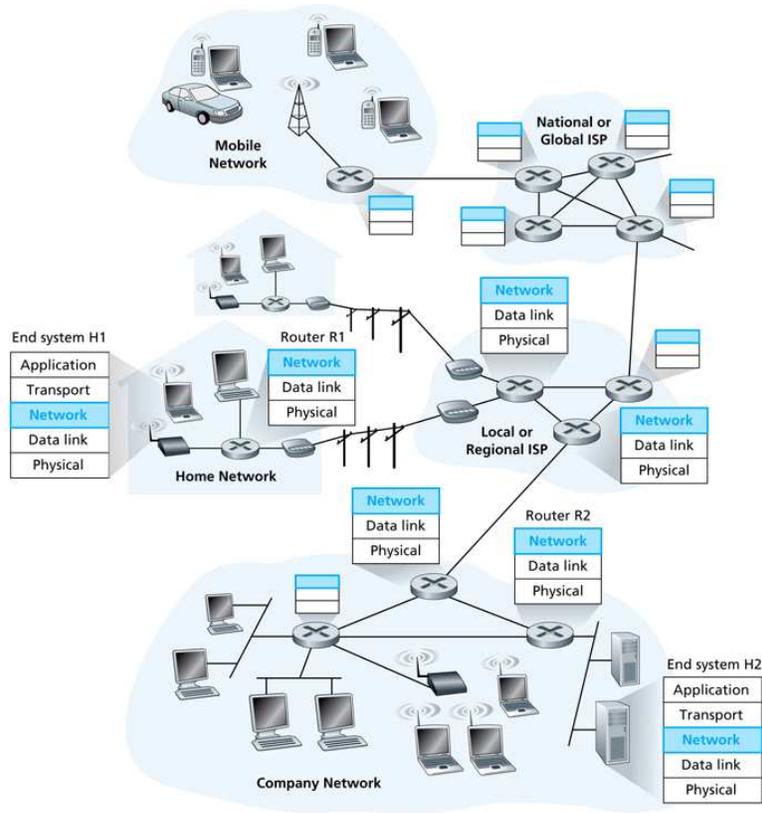
Outline

1. Introduction.
2. Virtual circuit and datagram networks.
3. What's inside a router?

Coming Up

IP.

2 Introduction



1. Network layer works end-to-end, but only host-to-host, unlike process-to-process of transport layer.
2. Dedicated routers only implement the bottom three layers of the protocol stack.

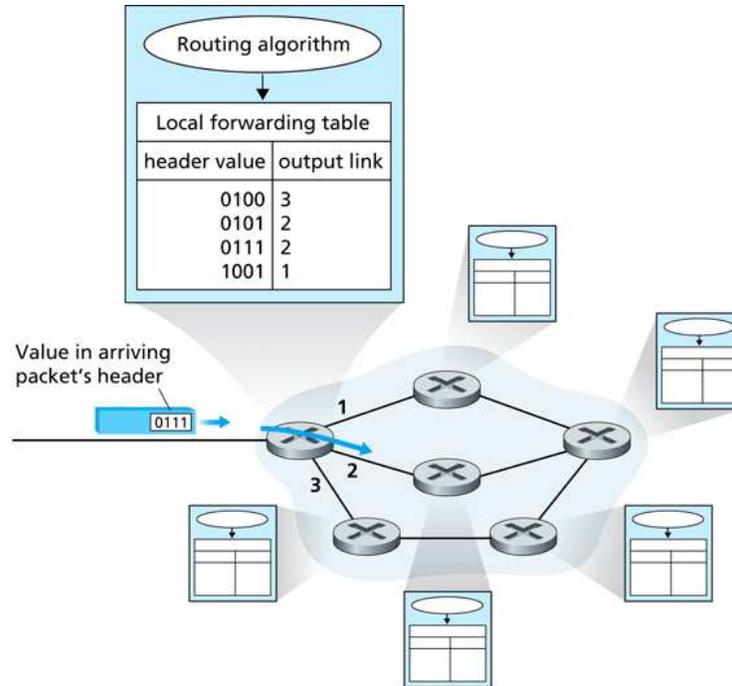
Forwarding vs. routing:

1. Both are router responsibilities.
2. Forwarding is the actual intra-router routing of a datagram from an input link directly attached to the router to an output link directly attached to the router.

The forwarding table determines this routing.

3. Routing is the process of determining the “best” path from one host to another.

The output from the routing algorithms are used to configure the forwarding tables. As such, they determine the host-to-host routes.



Routers also participate in connection setup for network-layer protocols based on the virtual circuit model, such as ATM.

Possible network service characteristics, WRT individual packets:

1. Guaranteed delivery.
2. Guaranteed delivery with bounded delay.

Possible network service characteristics, WRT packet flows:

1. In-order packet delivery.
2. Guaranteed minimal bandwidth.
3. Guaranteed maximum jitter.
4. Security services: encryption, data integrity, source authentication.

The Internet gives "best effort" service. "Best effort" is a euphemism for "No guarantees." ATM provides a couple different service levels:

Network Architecture	Service Model	Bandwidth Guarantee	No-Loss Guarantee	Ordering	Timing	Congestion Indication
Internet	Best effort	None	None	Any order possible	Not maintained	None
ATM	CBR	Guaranteed constant rate	Yes	In order	Maintained	Congestion will not occur
ATM	ABR	Guaranteed minimum	None	In order	Not maintained	Congestion indication provided

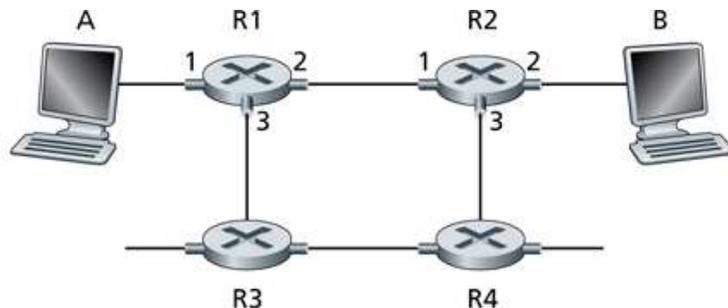
3 Virtual Circuit and Datagram Networks

Recall:

1. Virtual circuit: connection-oriented. Example: ATM.
2. Datagram: connectionless. Example: IP.

3.1 Virtual Circuit Networks

Consider the following example of a small virtual circuit network:



and R1's routing table:

Incoming Interface	Incoming VC #	Outgoing Interface	Outgoing VC #
1	12	2	22
2	63	1	18
3	7	2	17
1	97	3	87

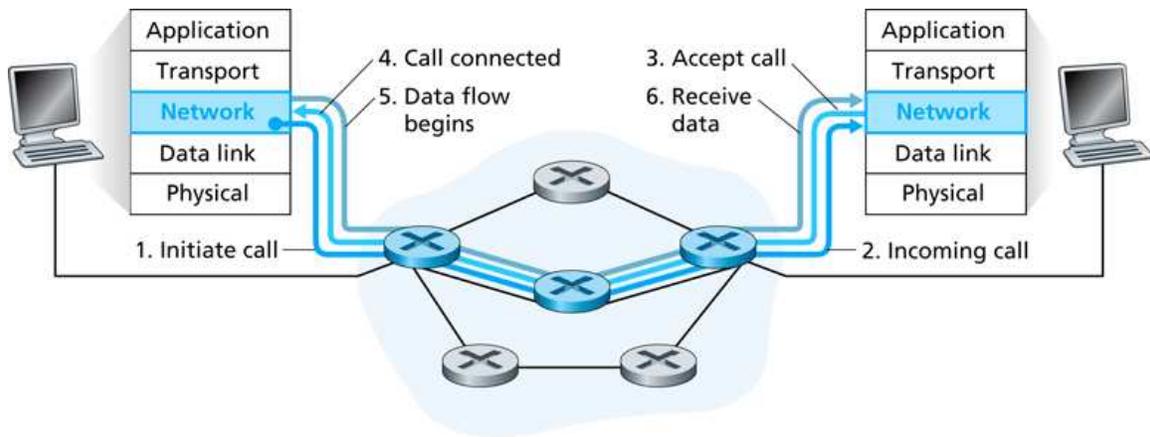
1. A virtual circuit is a sequence of links.
2. A given virtual circuit may have different virtual circuit numbers for each link along its route.

Why?

Three phases in the life of a virtual circuit:

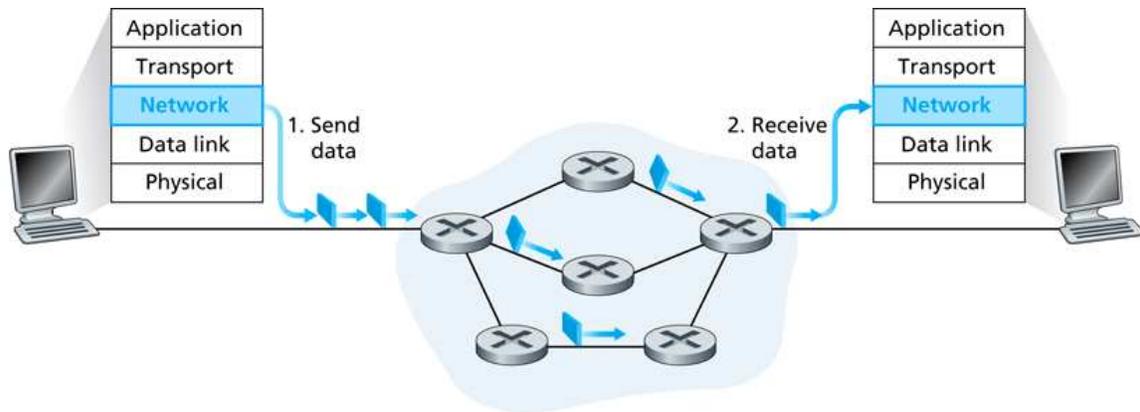
1. VC setup: The two end hosts establish state (windows, ISNs, etc.). The routers establish state (forwarding table entries, etc.), the route, and allocate resources.
2. Data transfer.
3. VC teardown. End hosts close the connection and reclaim state. The routers reclaim state and resources.

VCS created/torn down frequently.



3.2 Datagram Networks

No setup or teardown. Source just “dumps” packets into the network:



Routers must forward packets on incoming links to correct outgoing links.

Example: IP. 32 bit address. A complete forwarding table would have 2^{32} entries. Infeasible.

Instead, use hierarchical addressing and “blocks” of addresses:

Destination Address Range	Link Interface
11001000 00010111 00010000 00000000 through 11001000 00010111 00010111 11111111	0
11001000 00010111 00011000 00000000 through 11001000 00010111 00011000 11111111	1
11001000 00010111 00011001 00000000 through 11001000 00010111 00011111 11111111	2
otherwise	3

This table can be compressed to this:

Prefix Match	Link Interface
11001000 00010111 00010	0
11001000 00010111 00011000	1
11001000 00010111 00011	2
otherwise	3

1. Perform a table lookup to find the correct entry.
2. Longest prefix matching rule.
3. Smaller blocks — larger forwarding tables.

Forwarding table entries change infrequently.

1. Virtual circuit networks had their origins in telephony.

Dumb edge devices, smart core devices.

2. Datagram networks had their origins in computing

Smart edge devices. Pushing functionality to the edge increases innovation. Keep core devices simple.

Of course, we have only “best effort” service, due to this simplicity in the core.

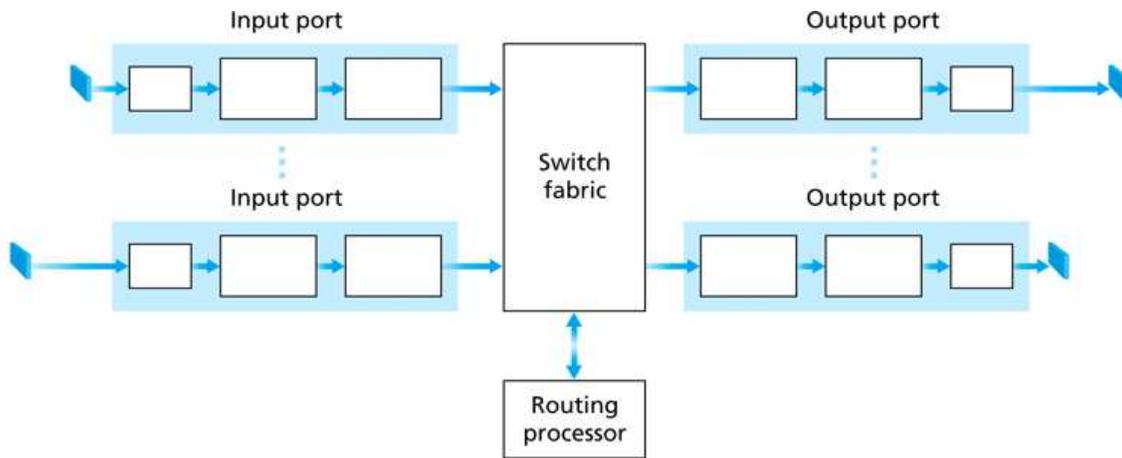
4 What’s Inside a Router?

1. Even end hosts performing a routing function:

```
bluebird:~/
* netstat -r
Kernel IP routing table
Destination  Gateway      Genmask      Flags      MSS Window  irtt  Iface
10.67.1.0    *            255.255.255.0  U          0 0      0     eth0
169.254.0.0  *            255.255.0.0   U          0 0      0     eth0
default      10.67.1.1   0.0.0.0       UG         0 0      0     eth0
```

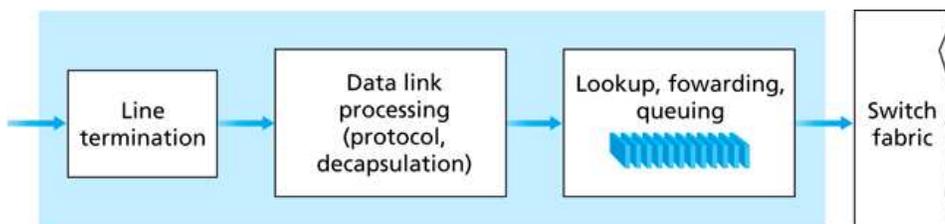
2. Servers can route, but not at backbone speeds.
3. At the backbone, servers are dedicated systems.

Block diagram of a router:



1. Input port: physical and data-link layer functions. Forwarding decision made here, possibly. Queuing possible here.
2. Switch fabric: a network within the network connecting input ports to output ports.
3. Output ports: queuing, data-link and physical layer functions.
4. Routing processor: management, computation routing algorithms, configuration of forwarding table(s).

A bit more detail on an input port:



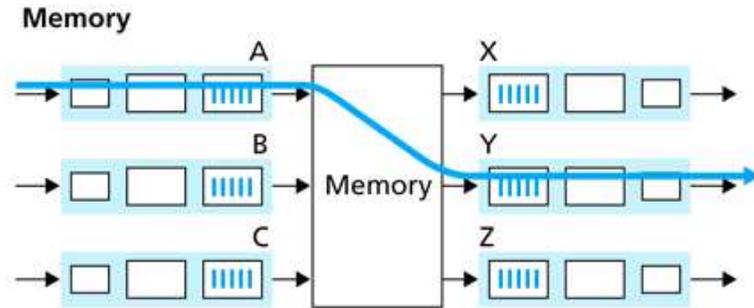
1. Decentralized forwarding frees eliminates the routing processor as a forwarding bottleneck.
2. Forwarding rate might be one million per sec.
3. Linear search infeasible. Use binary search — examine bits, from msb down.

High-end routers use CAMs.

Switching fabric architectures:

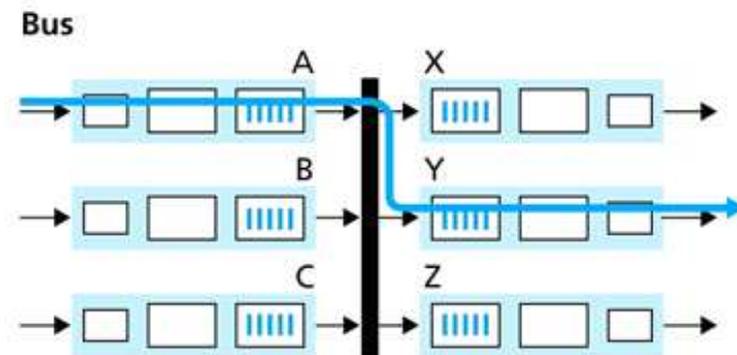
1. Memory-based — what a server router uses, with the general purpose CPU handling the route processing. Depending upon the memory architecture, one or several packets can be forwarded at-a-time.

Some dedicated routers use this architecture.



2. Bus-based — A single bus connecting all input and output ports. Only one packet can be forwarded at-a-time.

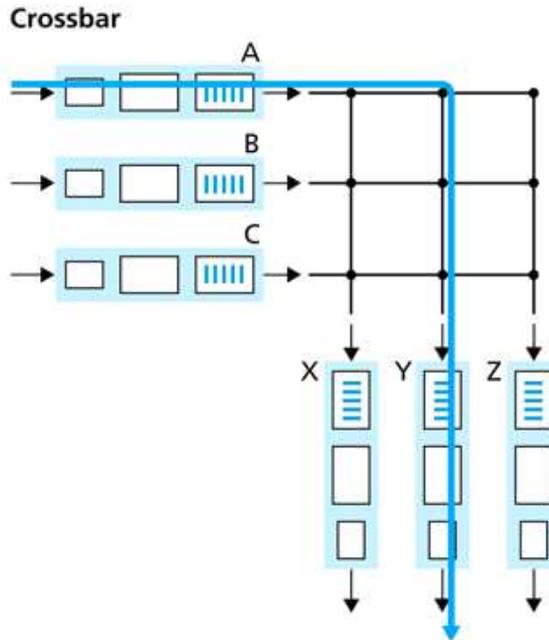
Higher performance than memory-based. Not uncommon in mid-level (enterprise) dedicated routers.



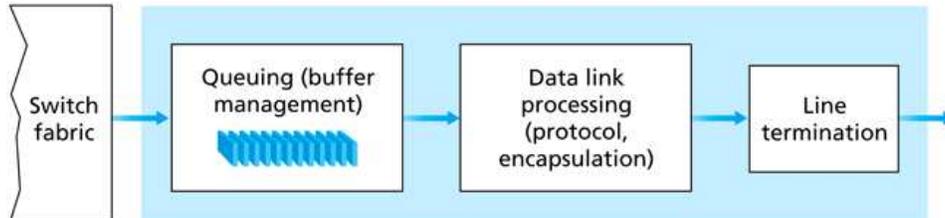
3. Crossbar-based — A two-dimensional collection of buses, with input ports aligned along one dimension and output ports aligned along the other. Multiple packets can be forwarded at-a-time.

More sophisticated configurations, such as omega networks, are coming into use.

Highest performance. Backbone routers would tend to use this architecture.

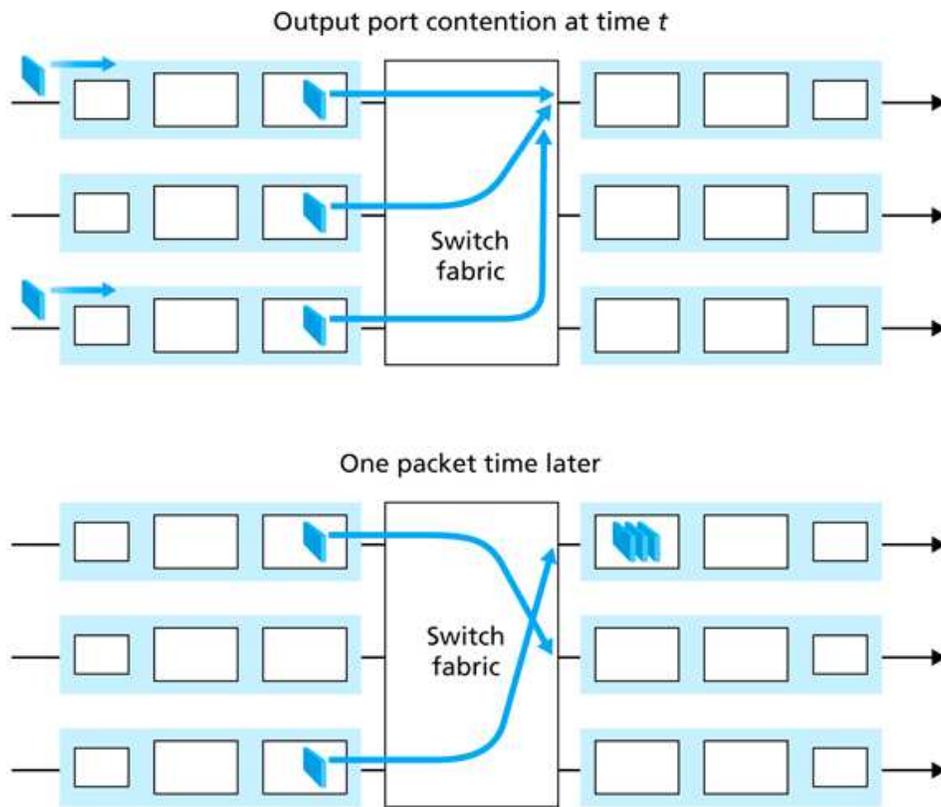


Output port details:

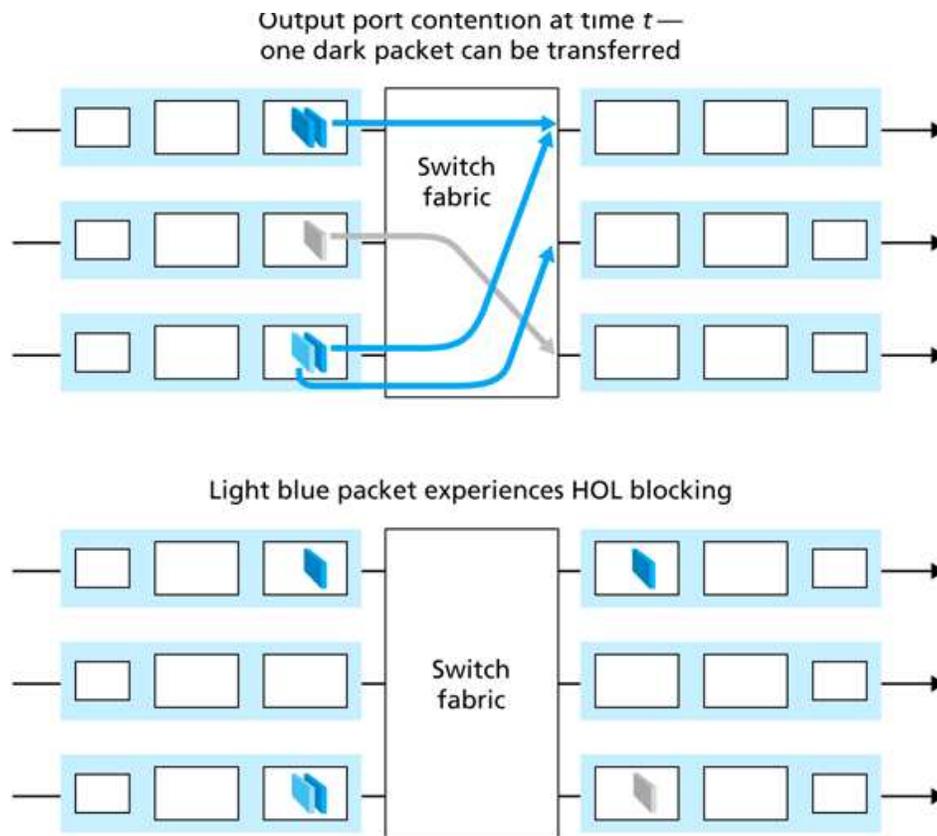


Switching fabric can dump packets into an output port more quickly than it can off-load them into the link, hence the need for queuing.

Switching fabric simultaneously forwarding three packets to the same output port:



Head-of-the-line (HOL) blocking:



Second packet in bottom input port blocked, even though its output port is available, because first packet is blocked.

HOL blocking can result in serious queueing problems.