EE 122: Switching and Forwarding

Kevin Lai
September 23, 2002
Direct Link Network Review

- Data link layer presents a single media (e.g., single wire) network model

- Problem and solutions
  - Framing
    - character stuffing, byte counting, bit stuffing, clocked framing
  - Error detection
    - parity, checksum, CRC
  - Reliability
    - stop and go, sliding window
  - solutions also apply to similar problems in higher layers
    - problems can not be completely solved at data link layer
    - only implemented in data link layer as optimization
Limitations of Direct Link Networks

- **distance**
  - distance increases propagation delay
  - large propagation delay causes large coordination delay
  - e.g., Ethernet collision detection requires $2 \times \text{prop\_delay}$

- **number of hosts**
  - More hosts increases the probability of collisions
  - collisions decrease efficiency of link

- **bandwidth**
  - bandwidth of link is shared among all connected nodes

- **single media type**
  - different media (e.g., fiber, Ethernet, wireless) have different tradeoffs for performance, cost, etc.
Direct Link Networks v.s. Switching

Direct Link Network

Switched Network

Single link

n links

Switch

Emulates clique

laik@cs.berkeley.edu
Definitions

- **switch** (aka *bridge*)
  - does *switching*
  - operates at data link layer
  - *router* also does switching, but at network layer

- **switching** consists of
  - *forwarding*
    - read data from input links,
    - decide which output link to forward on, and
      - examine packet header or incoming circuit, and
      - look up in *forwarding table*
    - transmit it on one of the output links (unicast)
  - *routing*
    - how the switch/router builds up its forwarding table
Properties

- spans larger physical area than direct link network (DLN)
  - can connect multiple switches together
- supports more hosts than DLN
  - hosts on separate links can transmit at same time
- higher aggregate bandwidth than DLN
  - approaches \((n/2)\cdot b\) instead of \(b\), \(n =\) number of switched links, \(b =\) bandwidth of one link
- supports more than one media type
  - more expensive for bridge than router
Bridge/Router Comparison

- Router interconnects different link layer protocols more easily

Switch

<table>
<thead>
<tr>
<th>Ethernet</th>
<th>E-to-E</th>
<th>Ethernet</th>
</tr>
</thead>
<tbody>
<tr>
<td>802.11b</td>
<td>E-to-8</td>
<td>802.11b</td>
</tr>
<tr>
<td>ATM</td>
<td>E-to-A</td>
<td>ATM</td>
</tr>
<tr>
<td>SONET</td>
<td>E-to-S</td>
<td>SONET</td>
</tr>
</tbody>
</table>

Router

<table>
<thead>
<tr>
<th>Ethernet</th>
<th>E-to-IP</th>
<th>IP-to-E</th>
<th>Ethernet</th>
</tr>
</thead>
<tbody>
<tr>
<td>802.11b</td>
<td>8-to-IP</td>
<td>IP-to-8</td>
<td>802.11b</td>
</tr>
<tr>
<td>ATM</td>
<td>A-to-IP</td>
<td>IP-to-A</td>
<td>ATM</td>
</tr>
<tr>
<td>SONET</td>
<td>S-to-IP</td>
<td>IP-to-S</td>
<td>SONET</td>
</tr>
</tbody>
</table>

\(O(n^2)\) converters
\(n = \text{different link types}\)

\(O(n)\) converters
Forwarding Techniques

- packet switching
  - aka [packet|datagram|connectionless]
    [switching|forwarding]
- source routing
- virtual circuit switching
  - aka virtual circuit forwarding
- circuit switching
- despite names, all ways for switch to decide which output port to forward data
Packet Switching

- Data is separated into packets
- Each packet is forwarded independently of previous packets
  - packets between two hosts can follow different paths
- On link failure, adjoining switches select new route and continue forwarding packets
- Statistical multiplexing
  - any one host may use 100% of a link’s bandwidth
Statistical Multiplexing v.s. Resource Reservations

- Reserve explicit amount of resources (e.g., bandwidth)
  - get exactly that amount
- Statistical multiplexing: get whatever is available

Advantage

Problem

10Mb/s / 10Mb/s

10Mb/s / 10Mb/s

congestion, packet loss

low utilization
Packet Switching Operation

- Each switch maintains a forwarding table
  - forwarding entry: (address, output port)

- Upon packet arrival
  - input port forwards the packet to the output port whose address matches packet’s destination address
    - exact match

- longest prefix match
  - forwarding entry: (address prefix, output port)
  - forward packet to the output port whose address matches packet’s destination address in the longest number of bits
Packet Switching Properties

- **Expensive forwarding**
  - forwarding table size depends on number of different destinations
  - must lookup in forwarding table for every packet

- **Robust**
  - link and router failure may be transparent for end-hosts

- **High bandwidth utilization**
  - statistical multiplexing

- **No service guarantees**
  - Network allows hosts to send more packets than available bandwidth → congestion → dropped packets
Source Routing

- Each packet specifies the sequence of routers, or alternatively the sequence of output ports, from source to destination
Source Routing (cont’d)

- Gives the source control of the path
- Not scalable
  - Packet overhead proportional to the number of routers
  - Typically, require variable header length which is harder to implement
- Hard for source to have complete information
- Loose source routing → sender specifies only a subset of routers along the path
Virtual Circuit (VC) Switching

- Packets not switched independently
  - establish virtual circuit before sending data

- Forwarding table entry
  - (input port, input VCI, output port, output VCI)
  - VCI – Virtual Circuit Identifier

- Each packet carries a VCI in its header

- Upon a packet arrival at interface i
  - Input port uses i and the packet’s VCI \( v \) to find the routing entry \((i, v, i', v')\)
  - Replaces \( v \) with \( v' \) in the packet header
  - Forwards packet to output port \( i' \)
VC Forwarding: Example

source

destination

in in-VCI out out-VCI

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>5</td>
<td>4</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

in in-VCI out out-VCI

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

in in-VCI out out-VCI

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>11</td>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

source

destination

in in-VCI out out-VCI

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>4</td>
<td>4</td>
</tr>
</tbody>
</table>

in in-VCI out out-VCI

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

laik@cs.berkeley.edu
VC Forwarding (cont’d)

- A signaling protocol is required to set up the state for each VC in the routing table
  - A source needs to wait for one RTT (round trip time) before sending the first data packet

- Can provide per-VC QoS
  - When we set the VC, we can also reserve bandwidth and buffer resources along the path
Virtual Circuit Switching Properties

- Less expensive forwarding
  - forwarding table size depends on number of different circuits
  - must lookup in forwarding table for every packet

- Much higher delay for short flows
  - 1 RTT delay for connection setup

- Less Robust
  - end host must spend 1 RTT to establish new connection after link and router failure

- Flexible service guarantees
  - either statistical multiplexing or resource reservations
Circuit Switching

- Packets not switched independently
  - establish circuit before sending data

- Circuit is a dedicated path from source to destination
  - e.g., old style telephone switchboard, where establishing circuit means connecting wires in all the switches along path
  - e.g., modern dense wave division multiplexing (DWDM) form of optical networking, where establishing circuit means reserving an optical wavelength in all switches along path

- No forwarding table
Circuit Switching Properties

- Cheap forwarding
  - no table lookup
- Much higher delay for short flows
  - 1 RTT delay for connection setup
- Less robust
  - end host must spend 1 RTT to establish new connection after link and router failure
- Must use resource reservations
## Forwarding Comparison

<table>
<thead>
<tr>
<th></th>
<th>Pure Packet Switching</th>
<th>Virtual Circuit Switching</th>
<th>Circuit Switching</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forwarding Cost</td>
<td>high</td>
<td>low</td>
<td>none</td>
</tr>
<tr>
<td>Bandwidth Utilization</td>
<td>high</td>
<td>flexible</td>
<td>low</td>
</tr>
<tr>
<td>Resource Reservations</td>
<td>none</td>
<td>flexible</td>
<td>yes</td>
</tr>
<tr>
<td>Robustness</td>
<td>high</td>
<td>low</td>
<td>low</td>
</tr>
</tbody>
</table>
Routing

- Update forwarding/routing tables
- Manual configuration
  - simple, error prone, work for administrator
- Learning bridges
  - all that is needed for single bridge
- Spanning Tree
  - necessary for multiple bridges
- Described in internetworking section
  - Distance Vector
  - Link State
Learning Bridges

<table>
<thead>
<tr>
<th>Host</th>
<th>Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
</tr>
</tbody>
</table>

![Diagram of a network with hosts and ports]
Learning Bridge Problem
Spanning Tree

As if. I am

Uh, no. B0/1 is.

I'm root

Cha. B0/3 is.

B1/2 is root

B1/1 is root

Get out. I am.

B1/2 is root

B1/1 is root

H1

H0
Summary

- **Switching**
  - overcome limitations of direct link networks

- **Forwarding techniques**
  - packet switching
  - source routing
  - virtual circuit switching
  - circuit switching

- **Routing techniques**