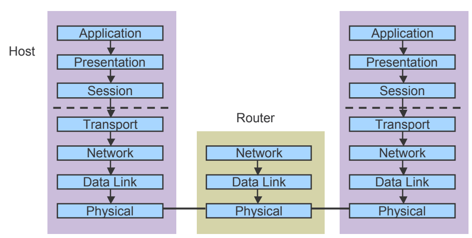
# Packet Forwarding

# Now Arriving at Layer 3

- … although layer 2 switches and layer 3 routers are similar in many ways
- … and ATM/Virtual Circuits are used at layer 2 these days

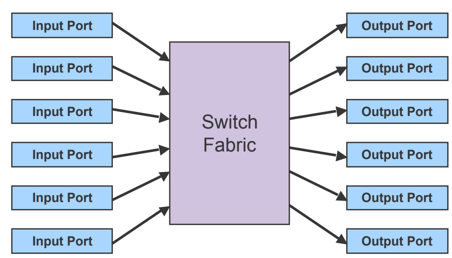


# Network Layers and Routers






# Router Design





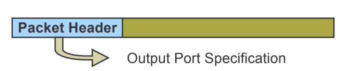

# Forwarding

- Forwarding Algorithm
  - Consult packet header
  - Consult forwarding tables
  - Decide on output port
- Three general types
  - Datagram forwarding
  - Virtual Circuits
  - Source Routing
- Differ by contents of header and tables



# Switching and Forwarding

- Forwarding
  - The task of specifying an appropriate output port for a packet
    - Datagram
    - Virtual Circuit Switching
    - Source Routing
  - Each packet contains enough information for a switch to determine the correct output port
  - Later
    - Building forwarding tables – routing.

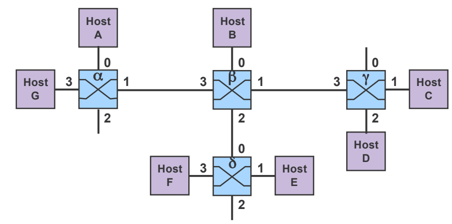


# Forwarding with Datagrams

- Connectionless
  - Each packet travels independently
- Switch
  - Translates global address to output port
  - Maintains table of translations
- Used in traditional data networks
  - i.e., Internet



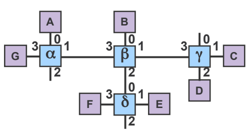
# Forwarding with Datagrams






# Routing Table

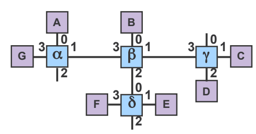
Each switch maintains a routing table that translates a host name to an output port



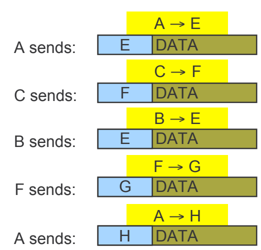
| α's Table | | β's Table | | γ's Table | | δ's Table | |
|---|---|---|---|---|---|---|---|
| A | 0 | A | 3 | A | 3 | A | 0 |
| B | 1 | B | 0 | B | 3 | B | 0 |
| C | 1 | C | 1 | C | 1 | C | 0 |
| D | 1 | D | 1 | D | 2 | D | 0 |
| E | 1 | E | 2 | E | 3 | E | 1 |
| F | 1 | F | 2 | F | 3 | F | 3 |
| G | 3 | G | 3 | G | 3 | G | 0 |



# Forwarding with Datagrams



What happens to the last packet?






# Forwarding with Datagrams

- Analogous to following signs
- Requires globally unique addresses
- Routing is decentralized
  - A router follows global routing algorithms
  - Two packets *usually* take the same path but…
  - Each router can change its mind at any time



# Traceroute Example

- From last year's solutions

```
traceroute www.scott.aq
traceroute to www.scott.aq (203.167.246.34), 30
   hops max, 40 byte packets
1 uiuc-ewsl-vlan1.gw.uiuc.edu (130.126.160.1) 0.425
   ms 0.213 ms 0.319 ms
2 …
13 ae-0-0.bbr1.Washington1.Level3.net
   (64.159.0.229) 21.946 ms
as-2-0.bbr2.Washington1.Level3.net (209.247.10.130)
   21.351 ms 21.280 ms
```

# Datagrams

- Advantages
  - Routes around failures
  - Can send traffic immediately
- Disadvantages
  - Header requires full unique address
  - Might not be possible to deliver packet
  - Successive packets may not follow the same route
  - Global address to path translations requires significant storage



# Forwarding with Virtual Circuits

- Connection oriented
  - Requires explicit setup and teardown
  - Packets follow established route
- Why support connections in a network?
  - Useful for service notions
  - Important for telephony
- Switch
  - Translates virtual circuit ID on incoming link to virtual circuit ID on outgoing link
  - Circuit Ids can be per-link or per-switch
- Used in ATM



# Virtual Circuits

- Packet header stores:
  - Virtual Circuit ID
- Router stores:
  - Table of how to forward packets for each virtual circuit
- Note: VCID need not be global
  - Assign a VCID to a circuit for each link-link pair



# Forwarding with Virtual Circuits

- Set up
  - A virtual circuit identifier (VCI) is assigned to the circuit for each link it traverses
  - VCI is locally significant
  - <incoming port, incoming VCI> uniquely identifies VC
- Switch
  - Maintains a translation table from <incoming port, incoming VCI> to <outgoing port, outgoing VCI>
- Permanent Virtual Circuits (PVC)
  - Long-lived
- Switch Virtual Circuits (SVC)
  - Uses signaling to establish VC



# Forwarding with Virtual Circuits

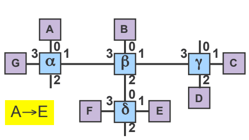



Table entries after A→E connection is set

δ

| Port IN | VCI IN | Port OUT | VCI OUT |
|---|---|---|---|
| 0 | 0 | 1 | 0 |

α

| Port IN | VCI IN | Port OUT | VCI OUT |
|---|---|---|---|
| 0 | 0 | 1 | 0 |

β

| Port IN | VCI IN | Port OUT | VCI OUT |
|---|---|---|---|
| 3 | 0 | 2 | 0 |



# Forwarding with Virtual Circuits

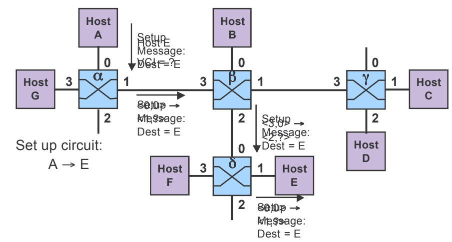
- A simple example setup protocol
  - Each host and switch maintains per-link local variable for VCI assignment
  - When setup frame leaves host/switch
    - Assign outgoing VCI
    - Increment assignment counter
  - port and circuit id combination is unique
  - switches maintain translation table from
    - incoming port/VCI pair to
    - outgoing port/VCI pair

## Forwarding with Virtual Circuits

- Assumptions
  - Circuits are simplex
    - On a duplex link, the same VCI can be used for two circuits, one in each direction
  - The same VCI can be used on different ports of the same switch
  - At setup, the lowest available VCI is used

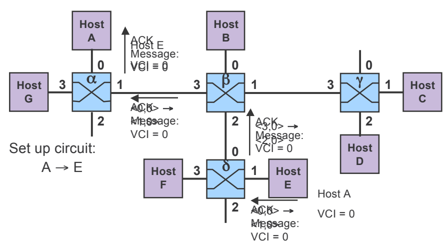


## Forwarding with Virtual Circuits



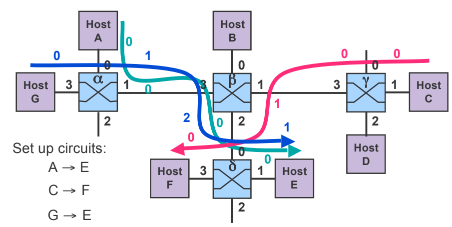



## Forwarding with Virtual Circuits






## Forwarding with Virtual Circuits






## Forwarding with Virtual Circuits

Table entries after A→E, C→F, G→E connection is set

α

| Port IN | VCI IN | Port OUT | VCI OUT |
|---|---|---|---|
| 0 | 0 | 1 | 0 |
| 3 | 0 | 1 | 1 |

β

| Port IN | VCI IN | Port OUT | VCI OUT |
|---|---|---|---|
| 1 | 0 | 2 | 1 |
| 3 | 0 | 2 | 0 |
| 3 | 1 | 2 | 2 |

δ

| Port IN | VCI IN | Port OUT | VCI OUT |
|---|---|---|---|
| 0 | 0 | 1 | 0 |
| 0 | 1 | 3 | 0 |
| 0 | 2 | 1 | 1 |

γ

| Port IN | VCI IN | Port OUT | VCI OUT |
|---|---|---|---|
| 1 | 0 | 3 | 0 |



## Forwarding with Virtual Circuits

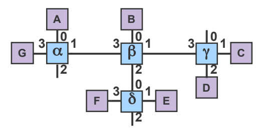
- Analogous to a game of following a sequence of clues
- Advantages
  - Header (for a data packet) requires only virtual circuit ID
    - Connection request contains global address
  - Can reserve resources at setup time
- Disadvantages
  - Typically must wait one RTT for setup
  - Cannot dynamically avoid failures, must reestablish connection
  - Global address path information still necessary for connection setup

## Forwarding with source routing

- Packet header specifies directions
  - One direction per switch
    - Absolute
      - Port name
      - Next switch name
    - Relative
      - Turn clockwise 3 ports
  - Switches may delete or rotate directions within packet headers
- No state stored at switch!



## Forwarding with Source Routing




What happens to the last packet?

A sends: A → E: 1 | 2 | 1 | DATA

C sends: C → F: 3 | 2 | 3 | DATA

B sends: B → E: 2 | 1 | DATA

F sends: F → G: 0 | 2 | 3 | DATA

A sends: A → H: 1 | 1 | 2 | 1 | DATA



## Forwarding with Source Routing

- Analogous to following directions
- Advantages
  - Simple switches
  - Fast and cheap
- Disadvantages
  - Hosts must know entire topology
  - Changes must propagate to all hosts
  - Headers might get large



## NOTE

- ATM slides not covered in lecture, provided as extra material



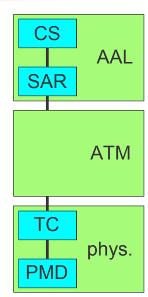
## ATM

- Defined by the ATM Forum
  - Formed October 1991
  - Joint effort of the telephony and data network industry
- High-Level Overview
  - Virtual circuit routing
  - Fixed length frames (aka cells)
  - Standard define 3 layers



## ATM

- ATL Adaptation Layer (AAL)
  - Convergence sub-layer (CS) supports different application service models
  - Segmentation and reassembly (SAR)
    - Supports variable-length frames
- ATM Layer
  - Virtual circuits maintenance
  - Cell header generation
  - Flow control
- Physical Layer
  - Transmission convergence (TC)
    - Error detection, Framing
  - Physical medium dependent (PMD) sublayer
    - encoding



## ATM Details

- Where is ATM used?
  - Common in WANs
  - Can also be used in LANs
- What is ATM built on?
  - Typically implemented on SONET
- Design
  - Connection establishment
    - Signaling (Q.2931)
  - Virtual circuits
  - Virtual Paths
    - Bundles of virtual circuits
    - Share common route
    - Optimizes forwarding



## ATM Cells

- Cell specification
  - 53-bytes
  - 48-byte payload
  - 1-byte CRC
  - 4-byte header

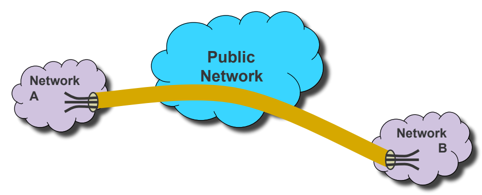


## ATM Rationale

- Why hierarchical connections?
  - Setup
    - New virtual circuits can follow existing virtual path routes
  - Forwarding
    - Virtual Path Identifier (VPI)
      - Used between switches
    - Virtual Circuit Identifier (VCI)
      - Used for last hop
  - Routing around failures
    - Need only change virtual path once for 64K virtual circuits



## ATM Rationale






## ATM Rationale

- Why fixed-length frames?
  - Hardware
    - Simpler processing for known frame sizes
    - Parallelization of processing stages
  - Is there an optimal length?
    - Small cells
      - High header-to-data overhead
      - Large frames must be fragmented
    - Large cells
      - Low utilization for small messages



## ATM Rationale

- Why short cells?
  - Better queueing behavior
  - Reduced granularity of preemption
    - High priority cell waits for one cell
      - Long cell: potentially long wait
      - Short cell: limited wait
    - Limits end-to-end jitter (variance in latency)
  - Shorted store-and-forward delay
    - Switches typically store whole frame, then forward
    - Short cells enable first part of a fragmented frame to be sent while the rest is still arriving

# ATM Rationale

- Queueing Behavior Example
  - Consider 4KB vs. 53B cells, 100Mbps Link
  - Preemption
    - High priority cell arrives just as switch starts sending low-priority cell
    - 4KB: high-priority cell must wait for 328μs
    - 53B: high priority cell must wait for 4μs
  - Queueing
    - Two 4KB frames arrive simultaneously at time 0
    - 4KB: link is idle until all data arrives at time 328μs, 8KB left to send
    - 53B: First 53B is sent at time 4μs. At time 328μs, 4KB left to send



# ATM Rationale

- Why 53-byte cells?
  - US wanted 64-bytes
    - Digital encoding for voice
      - 1 frame = 64Kbps (8-bit samples, 8Khz)
      - Collect one sample per frame
      - With 64-byte cells, no need for echo cancellation
    - Latency – 1 cell = 6msec
      - Not detectable by humans
  - Europe wanted 32-byte
    - Shorter distances, no need for echo cancellation
  - Compromise – 48-bytes of data!
- Problems with 53-byte (48-bytes of data) cells
  - Not a power of 2!



# ATM and LANs

- Comments
  - Switched networks have better performance then shared media
  - Shared media performance is increasing (100-MBps and Gigabit Ethernet)
- ATM in a LAN
  - ATM doesn't look like a traditional LAN
    - Specifically, no native support for broadcast and multicast
  - Solution
    - Redesign protocols that require broadcast/multicast
    - Make ATM behave more like a shared medium
      - LAN Emulation (LANE)

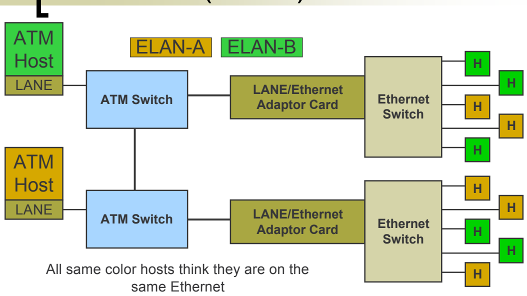


# Structure of LANE

- ATM network can have multiple Emulated LAN's (ELAN's)
  - Each ELAN corresponds to a single network
- Networks do not have to be geographically oriented
  - Hosts can move between buildings, but remain on the same network
- Access Control Lists (ACL's) on LANE servers
  - Control which hosts can join which ELAN's



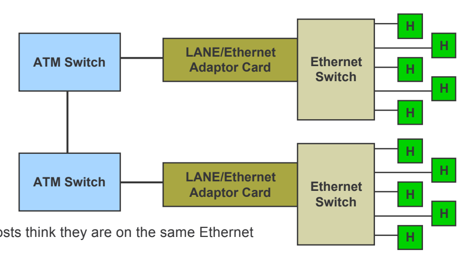
# ATM Local Area Network Emulation (LANE)




All same color hosts think they are on the same Ethernet



# ATM Local Area Network Emulation (LANE)




All hosts think they are on the same Ethernet

# ATM/LANE Protocol Layers