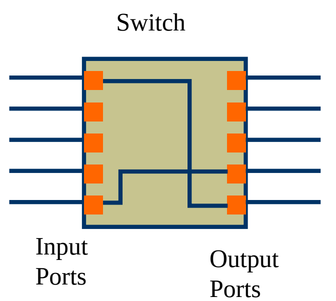
15-441 Computer Networking

Lecture 13:
Virtual Circuits, ATM, MPLS

Outline

- Circuit and Packet switching refresher
- Virtual Circuits - general
  - Why virtual circuits?
  - How virtual circuits?
- Two modern implementations
  - ATM - Teleco-style virtual circuits
  - MPLS - IP-style virtual circuits

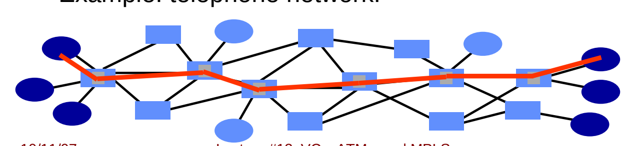
Circuit Switching



Connects (electrons or bits) ports to ports

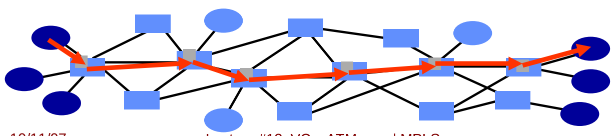
Circuit Switching

- Source first establishes a connection (circuit) to the destination.
  - Each router or switch along the way may reserve some bandwidth for the data flow
- Source sends the data over the circuit.
  - No need to include the destination address with the data since the routers know the path
- The connection is torn down.
- Example: telephone network.



Packet Switching

- Source sends information as self-contained packets that have an address.
  - Source may have to break up single message in multiple
- Each packet travels independently to the destination host.
  - Routers and switches use the address in the packet to determine how to forward the packets
- Destination recreates the message.
- Analogy: a letter in surface mail.

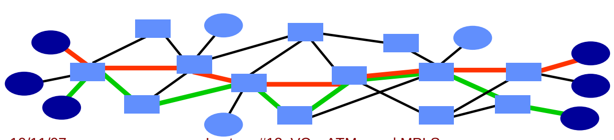


Circuit vs. Packet Switching

- Setup: initial delay in CS, not in PS
- Reservation: guaranteed BW and performance in CS, not in PS
- Queues: none in CS, while packets are buffered in PS
- Efficiency: CS wastes BW specially for bursty traffic, no waste in PS
- Lookup: simple in CS, more difficult in PS (longest-prefix lookup)
- Multiplexing: fixed in CS (TDM, FDM), Statistical in PS
- Path choice: Arbitrary in CS, depends on destination in PS.
- State: per-connection in CS (hard state), vs no state in PS
- Big Question: Can we get the advantages of Circuit switching without (all) the disadvantages?

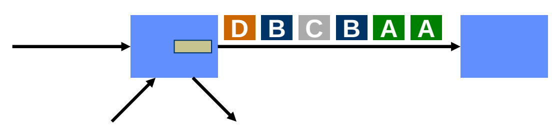
Virtual Circuits

- Each wire carries many "virtual" circuits.
  - Forwarding based on virtual circuit (VC) identifier
    - IP header: src, dst, etc.
    - Virtual circuit header: just "VC"
  - A path through the network is determined for each VC when the VC is established
  - Use statistical multiplexing for efficiency
- Can support wide range of quality of service.
  - No guarantees: best effort service
  - Weak guarantees: delay < 300 msec, …
  - Strong guarantees: e.g. equivalent of physical circuit



Packet Switching and Virtual Circuits: Similarities

- "Store and forward" communication based on an address.
  - Address is either the destination address or a VC identifier
- Must have buffer space to temporarily store packets.
  - E.g. multiple packets for some destination arrive simultaneously
- Multiplexing on a link is similar to time sharing.
  - No reservations: multiplexing is statistical, i.e. packets are interleaved without a fixed pattern
  - Reservations: some flows are guaranteed to get a certain number of "slots"



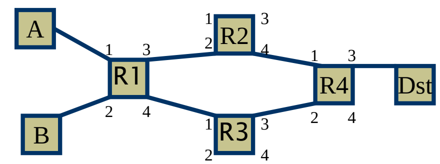
Virtual Circuits Versus Packet Switching

- Circuit switching:
  - Uses short connection identifiers to forward packets
  - Switches know about the connections so they can more easily implement features such as quality of service
  - Virtual circuits form basis for traffic engineering: VC identifies long-lived stream of data that can be scheduled
- Packet switching:
  - Use full destination addresses for forwarding packets
  - Can send data right away: no need to establish a connection first
  - Switches are stateless: easier to recover from failures
  - Adding QoS is hard
  - Traffic engineering is hard: too many packets!

Packet switched vs. VC

Payload | VCI

Payload | Dst



R1 packet forwarding table:

Dst R2

R1 VC table:

VC 1 R2

VC 2 R3

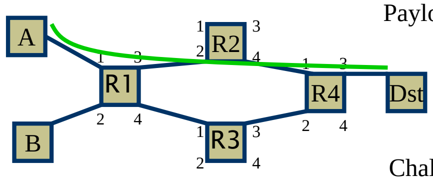
Different paths to same destination!

(useful for traffic engineering!)

Virtual Circuit

Payload | VCI

Payload



R1 VC table:

VC 5 R2

R2 VC table:

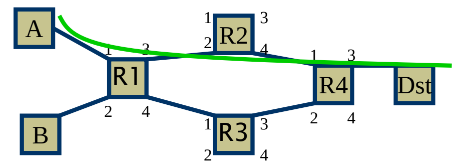
VC 5 R4

Challenges:

- How to assign IDs?

- How to set up path?

Virtual Circuit IDs/Switching: Label ("tag") Swapping



- Global VC ID allocation -- ICK! Solution: Per-link uniqueness. *Change VCI each hop.*

	Input Port	Input VCI	Output Port	Output VCI
R1:	1	5	3	9
R2:	2	9	4	2
R4:	1	2	3	5

Label ("tag") Swapping

- Result: Signalling protocol must only find per-link unused VCIs.
  - "Link-local scope"
  - Connection setup can proceed hop-by-hop.
    - Good news for our setup protocols!
- Second Challenge: How do we set a VC up?

VC setup: Permanent VCs and Switched VCs

- Permanent vs. Switched virtual circuits (PVCs, SVCs)
- Main difference is: static vs. dynamic.
- PVCs last "a long time"
  - E.g., connect two bank locations with a direct link (really expensive!) or setup a PVC that looks like a circuit
  - Administratively configured
- SVCs is temporary
  - Setup is more like a phone call
  - SVCs dynamically set up on a "per-call" basis

PVC connection setup

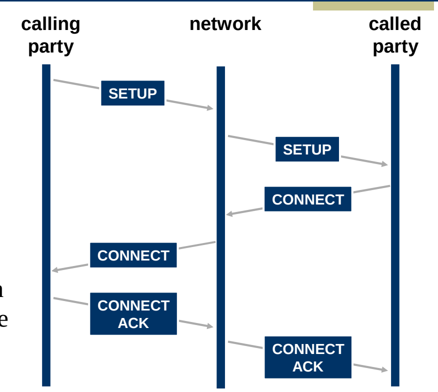
- Manual?
  - Configure each switch by hand. Ugh.
- Dedicated signaling protocol
  - E.g., what ATM uses
- Piggyback on routing protocols
  - Used in MPLS. E.g., use BGP to set up
- During connection setup, the VC tables and resources are reserved (if needed) during setup time.

SVC Connection Setup

- Hop by hop SVC setup. We now make use of label switching and VCI labeling.

- Setup VC tables along the path.

- Resource reservation occurs during this time as well



Virtual Circuits In Practice

- ATM: Teleco approach
  - Kitchen sink. Based on voice, support file transfer, video, etc., etc.
  - Intended as IP replacement. That didn't happen. :)
  - Today: Underlying network protocol in many teleco networks. E.g., DSL speaks ATM. IP over ATM in some cases.
- MPLS: The "IP Heads" answer to ATM
  - Stole good ideas from ATM
  - Integrates well with IP
  - Today: Used inside some networks to provide VPN support, traffic engineering, simplify core.
- Other nets just run IP.
- Older tech: Frame Relay
  - Only provided PVCs. Used for quasi-dedicated 56k/T1 links between offices, etc. Slower, less flexible than ATM.

ATM: Asynchronous Transfer Mode

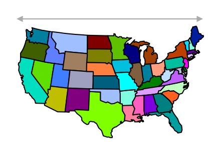
- Connection-oriented, packet-switched
  - (e.g., virtual circuits).
- Teleco-driven. Goals:
  - Handle voice, data, multimedia
  - Support both PVCs and SVCs
  - Replace IP. (didn't happen…)
- Important feature: Cell switching

Cell Switching

- Small, fixed-size cells
  [Fixed-length data][header]
- Why?
  - Efficiency: All packets the same
    - Easier hardware parallelism, implementation
  - Switching efficiency:
    - Lookups are easy -- table index.
  - Result: Very high cell switching rates.
  - Initial ATM was 155Mbit/s. Ethernet was 10Mbit/s at the same time. (!)
- What is the cell size?

Why 53 Bytes?

- Small cells favored by voice applications
  - delays of more than about 10 ms require echo cancellation
  - each payload byte consumes 125 µs (8000 samples/sec)
- Large cells favored by data applications
  - Five bytes of each cell are overhead
- France favored 32 bytes
  - 32 bytes = 4 ms packetization delay.
  - France is 3 ms wide.
  - Wouldn't need echo cancellers!
- USA, Australia favored 64 bytes
  - 64 bytes = 8 ms
  - USA is 16 ms wide
  - Needed echo cancellers anyway, wanted less overhead
- Compromise



ATM Features

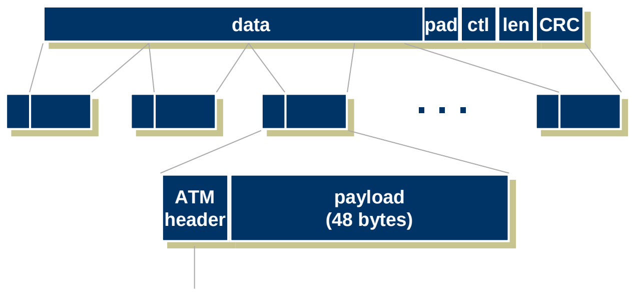
- Fixed size cells (53 bytes).
- Virtual circuit technology using hierarchical virtual circuits.
- PHY (physical layer) processing delineates cells by frame structure, cell header error check.
- Elaborate signaling stack.
  - Backwards compatible with respect to the telephone standards
- Standards defined by ATM Forum.
  - Organization of manufacturers, providers, users
- Support for multiple traffic classes by adaptation layer.
  - E.g. voice channels, data traffic

ATM Adaptation Layers

1	2	3	4	5
synchronous		asynchronous		
constant	variable bit rate			
connection-oriented			connectionless	

- **AAL 1: audio, uncompressed video**
- **AAL 2: compressed video**
- **AAL 3: long term connections**
- **AAL 4/5: data traffic**
  - **AAL5 is most relevant to us…**

AAL5 Adaptation Layer



includes EOF flag

Pertinent part: Packets are spread across multiple ATM cells. Each packet is delimited by EOF flag in cell.

ATM Packet Shredder Effect

- Cell loss results in packet loss.
  - Cell from middle of packet: lost packet
  - EOF cell: lost two packets
  - Just like consequence of IP fragmentation, but VERY small fragments!
- Even low cell loss rate can result in high packet loss rate.
  - E.g. 0.2% cell loss -> 2 % packet loss
  - Disaster for TCP
- Solution: drop remainder of the packet, i.e. until EOF cell.
  - Helps a lot: dropping useless cells reduces bandwidth and lowers the chance of later cell drops
  - Slight violation of layers
  - Discovered after early deployment experience with IP over ATM.

IP over ATM

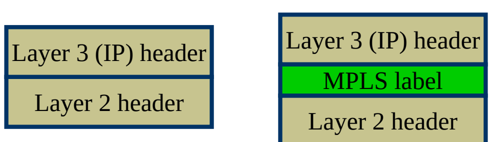
- When sending IP packets over an ATM network, set up a VC to destination.
  - ATM network can be end to end, or just a partial path
  - ATM is just another link layer
- Virtual connections can be cached.
  - After a packet has been sent, the VC is maintained so that later packets can be forwarded immediately
  - VCs eventually times out
- Properties.
  - – Overhead of setting up VCs (delay for first packet)
  - – Complexity of managing a pool of VCs
  - + Flexible bandwidth management
  - + Can use ATM QoS support for individual connections (with appropriate signaling support)

ATM Discussion

- At one point, ATM was viewed as a replacement for IP.
  - Could carry both traditional telephone traffic (CBR circuits) and other traffic (data, VBR)
  - Better than IP, since it supports QoS
- Complex technology.
  - Switching core is fairly simple, but
  - Support for different traffic classes
  - Signaling software is very complex
  - Technology did not match people's experience with IP
    - deploying ATM in LAN is complex (e.g. broadcast)
    - supporting connection-less service model on connection-based technology
  - With IP over ATM, a lot of functionality is replicated
- Currently used as a datalink layer supporting IP.

MPLS: Multi Protocol Label Switching

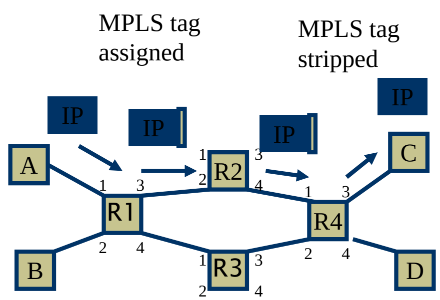
- Selective combination of VCs + IP
  - Today: MPLS useful for traffic engineering, reducing core complexity, and VPNs
- Core idea: Layer 2 carries VC label
  - Could be ATM (which has its own tag)
  - Could be a "shim" on top of Ethernet/etc.:
  - Existing routers could act as MPLS switches just by examining that shim -- no radical re-design. Gets flexibility benefits, though not cell switching advantages



MPLS + IP

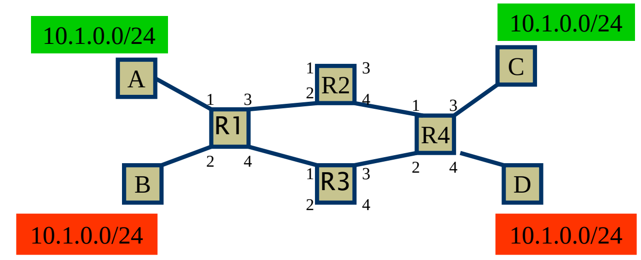
- In MPLS, a label is associated with the packet when it enters the network and forwarding is based on the label in the network core.
  - Label is swapped (as ATM VCIs)
- Potential advantages.
  - Packet forwarding can be faster
  - Routing can be based on ingress router and port
  - Can use more complex routing decisions
  - Can force packets to followed a pinned route

MPLS core, IP interface



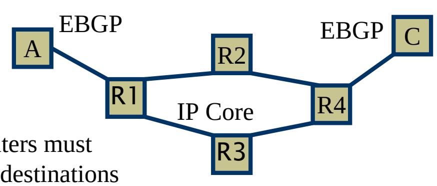
MPLS forwarding in core

MPLS use case #1: VPNs



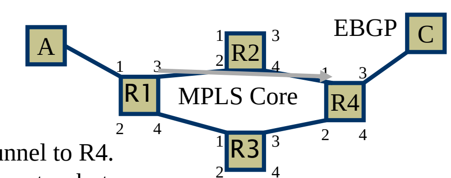
MPLS tags can differentiate green VPN from orange VPN.

MPLS use case #2: Reduced State Core



A-> C pkt

Internal routers must know all C destinations



R1 uses MPLS tunnel to R4. R1 and R4 know routes, but R2 and R3 don't.

MPLS use case #3: Traffic Engineering

- As discussed earlier -- can pick routes based upon more than just destination
- Used in practice by many ISPs, though certainly not all.

MPLS Discussion

- MPLS packet forwarding: implementation of the label is technology specific.
  - Could be ATM VCI or a short extra “MPLS” header
- Supports stacked labels.
  - Operations can be “swap” (normal label swapping), “push” and “pop” labels.
    - VERY flexible! Like creating tunnels, but much simpler -- only adds a small label.
- Currently mostly used for traffic engineering and network management.
  - LSPs (Label Switched Path) can be thought of as “programmable links” that can be set up under software control
  - …on top of a simple, static hardware infrastructure

Take Home Points

- Costs/benefits/goals of virtual circuits
- Cell switching (ATM)
  - Fixed-size pkts: Fast hardware
  - Packet size picked for low voice jitter. Understand trade-offs.
  - Beware packet shredder effect (drop entire pkt)
- Tag/label swapping
  - Basis for most VCs.
  - Makes label assignment link-local. Understand mechanism.
- MPLS - IP meets virtual circuits
  - MPLS tunnels used for VPNs, traffic engineering, reduced core routing table sizes

--- Extra Slides ---

Extra information if you’re curious.

ATM Traffic Classes

- Constant Bit Rate (CBR) and Variable Bit Rate (VBR).
  - Guaranteed traffic classes for different traffic types.
- Unspecified Bit Rate (UBR).
  - Pure best effort with no help from the network
- Available Bit Rate (ABR).
  - Best effort, but network provides support for congestion control and fairness
  - Congestion control is based on explicit congestion notification
    - Binary or multi-valued feedback
  - Fairness is based on Max-Min Fair Sharing.

(small demands are satisfied, unsatisfied demands share

LAN Emulation

- Motivation: making a non-broadcast technology work as a LAN.
  - Focus on 802.x environments
- Approach: reuse the existing interfaces, but adapt implementation to ATM.
  - MAC - ATM mapping
  - multicast and broadcast
  - bridging
  - ARP
- Example: Address Resolution "Protocol" uses an ARP server instead of relying on

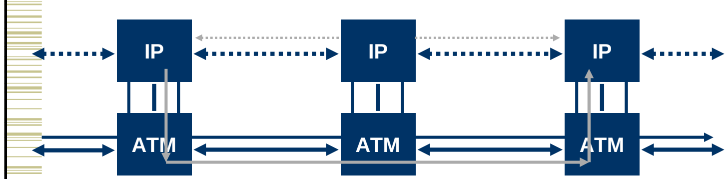
Further reading - MPLS

- MPLS isn't in the book - sorry. Juniper has a few good presentations at NANOG (the North American Network Operators Group; a big collection of ISPs):
  - http://www.nanog.org/mtg-0310/minei.html
  - http://www.nanog.org/mtg-0402/minei.html
  - Practical and realistic view of what people are doing _today_ with MPLS.

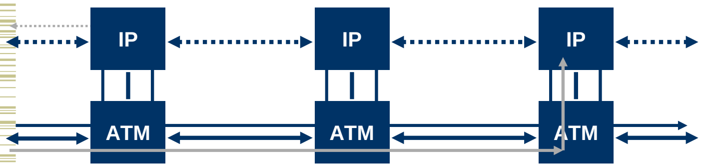
IP Switching

- How to use ATM hardware without the software.
  - ATM switches are very fast data switches
  - software adds overhead, cost
- The idea is to identify flows at the IP level and to create specific VCs to support these flows.
  - flows are identified on the fly by monitoring traffic
  - flow classification can use addresses, protocol types, ...
  - can distinguish based on destination, protocol, QoS
- Once established, data belonging to the flow bypasses level 3 routing.
  - never leaves the ATM switch
- Interoperates fine with "regular" IP routers.
  - detects and collaborates with neighboring IP switches

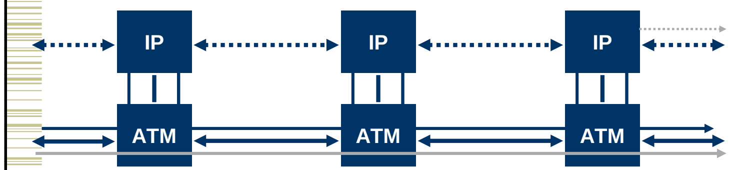
IP Switching Example



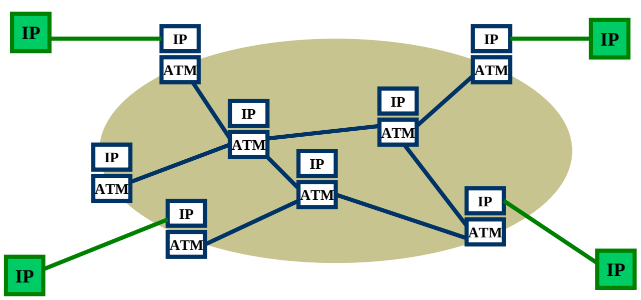
IP Switching Example



IP Switching Example



Another View

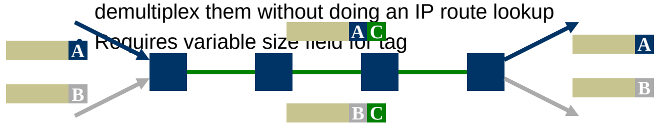


IP Switching Discussion

- IP switching selectively optimizes the forwarding of specific flows.
  - Offloads work from the IP router, so for a given size router, a less powerful forwarding engine can be used
  - Can fall back on traditional IP forwarding if there are failures
- IP switching couples a router with an ATM switching using the GSMP protocol.
  - General Switch Management Protocol
- IP switching can be used for flows with different granularity.
  - Flows belonging to an application .. Organization
  - Controlled by the classifier

An Alternative Tag Switching

- Instead of monitoring traffic to identify flows to optimize, use routing information to guide the creation of “switched” paths.
  - Switched paths are set up as a side effect of filling in forwarding tables
- Generalize to other types of hardware.
- Also introduced stackable tags.
  - Made it possible to temporarily merge flows and to demultiplex them without doing an IP route lookup
  - Requires variable size field for tag

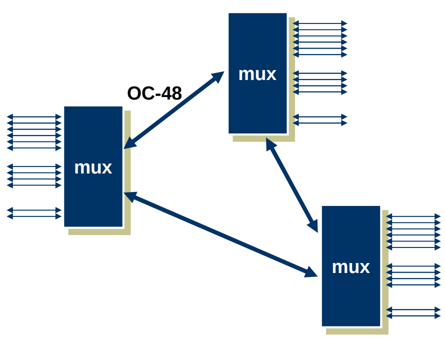


IP Switching versus Tag Switching

- Flows versus routes.
  - tags explicitly cover groups of routes
  - tag bindings set up as part of route establishment
  - flows in IP switching are driven by traffic and detected by “filters”
    - Supports both fine grain application flows and coarser grain flow groups
- Stackable tags.
  - provides more flexibility
- Generality
  - IP switching focuses on ATM
  - not clear that this is a fundamental difference

Packets over SONET

- Same as statically configured ATM pipes, but pipes are SONET channels.
- Properties.
  - – Bandwidth management is much less flexible
  - + Much lower transmission overhead (no ATM headers)



Internal BGP (I-BGP)

- R3 can tell R1 and R2 prefixes from R4
- R3 can tell R4 prefixes from R1 and R2
- R3 cannot tell R2 prefixes from R1

R2 can only find these prefixes through a *direct connection* to R1
Result: I-BGP routers must be fully connected (via TCP)!
- contrast with E-BGP sessions that map to physical links