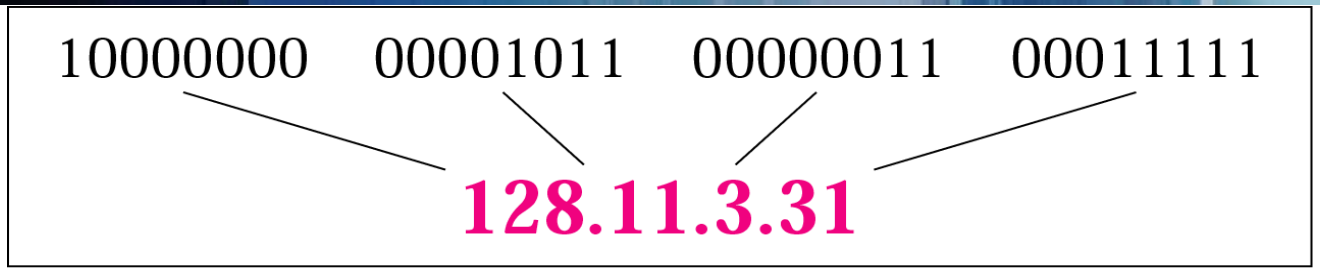
IP Addresses



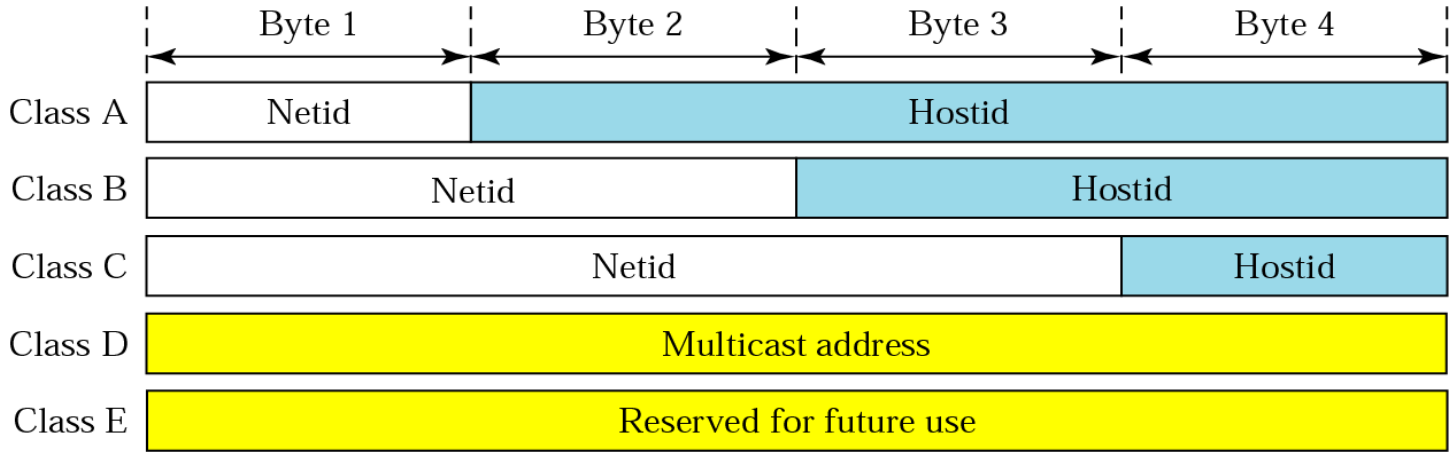
- 32-bit number
  - 4.294.967.296 addresses
- IP addresses are unique and universal
  - except private addresses

Range			Total
10.0.0.0	to	10.255.255.255	2^{24}
172.16.0.0	to	172.31.255.255	2^{20}
192.168.0.0	to	192.168.255.255	2^{16}

- Dotted decimal notation:
  - Bytes of binary notation represented as decimal separated by dot

* Figure is courtesy of B. Forouzan

Classful Addresses



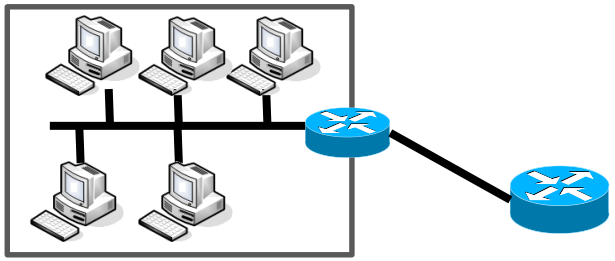
- Class A (international organisations)
  - 126 networks with 16,277,214 hosts each
- Class B (large companies)
  - 16,384 networks with 65,354 hosts each
- Class C (smaller companies)
  - 2,097,152 networks with 254 hosts each

- Inefficient use of hierarchical address space
  - Class C with 2 hosts (2/254 = 0.78% efficient)
  - Class B with 256 hosts (256/65534 = 0.39% efficient)

* Figure is courtesy of B. Forouzan

Network Layer

Trinity College

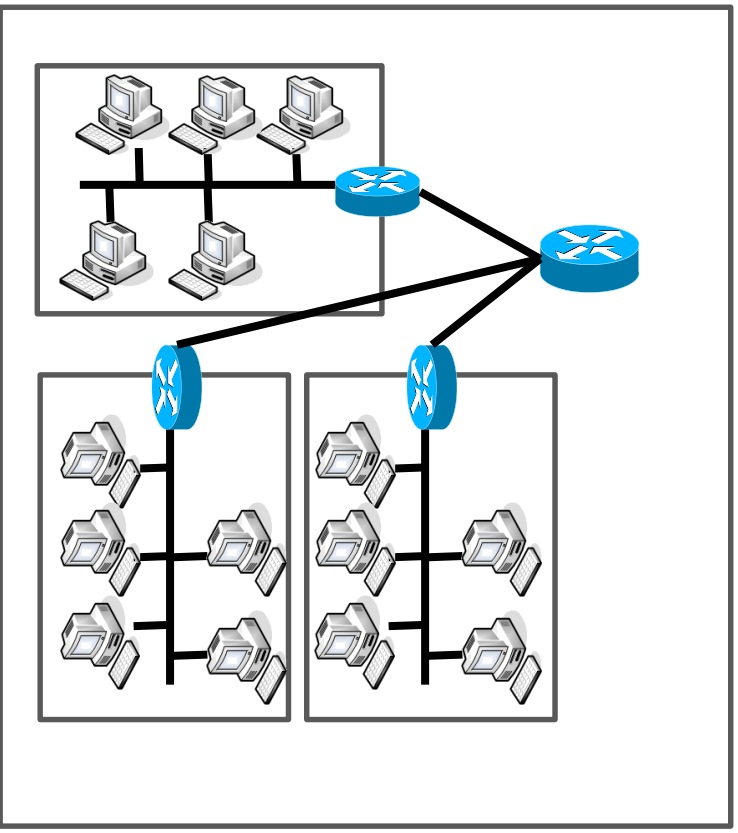


134.226.0.0

Class B

Network Layer

Trinity College

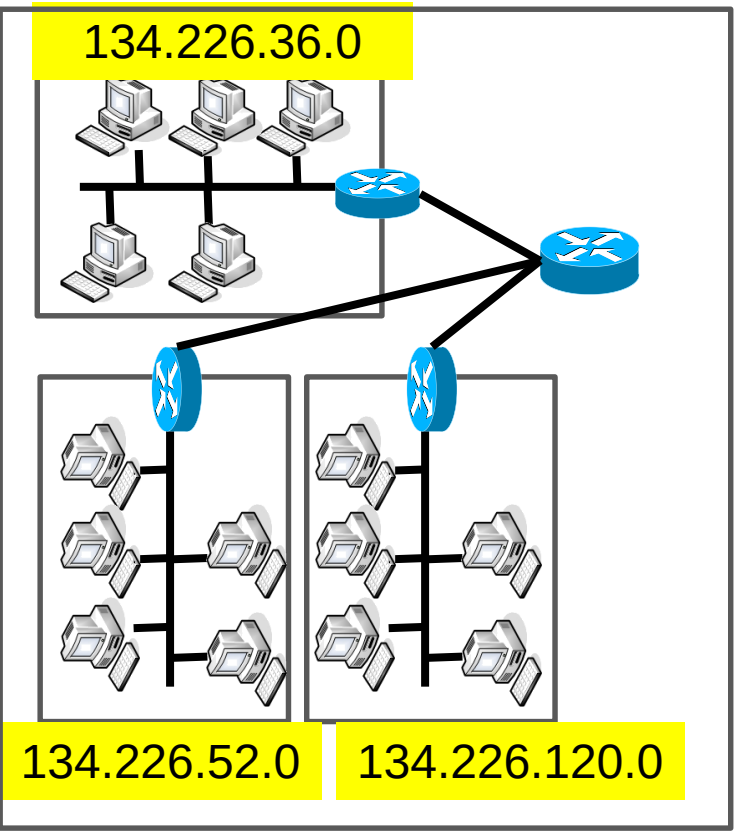


134.226.0.0

Class B

Network Layer

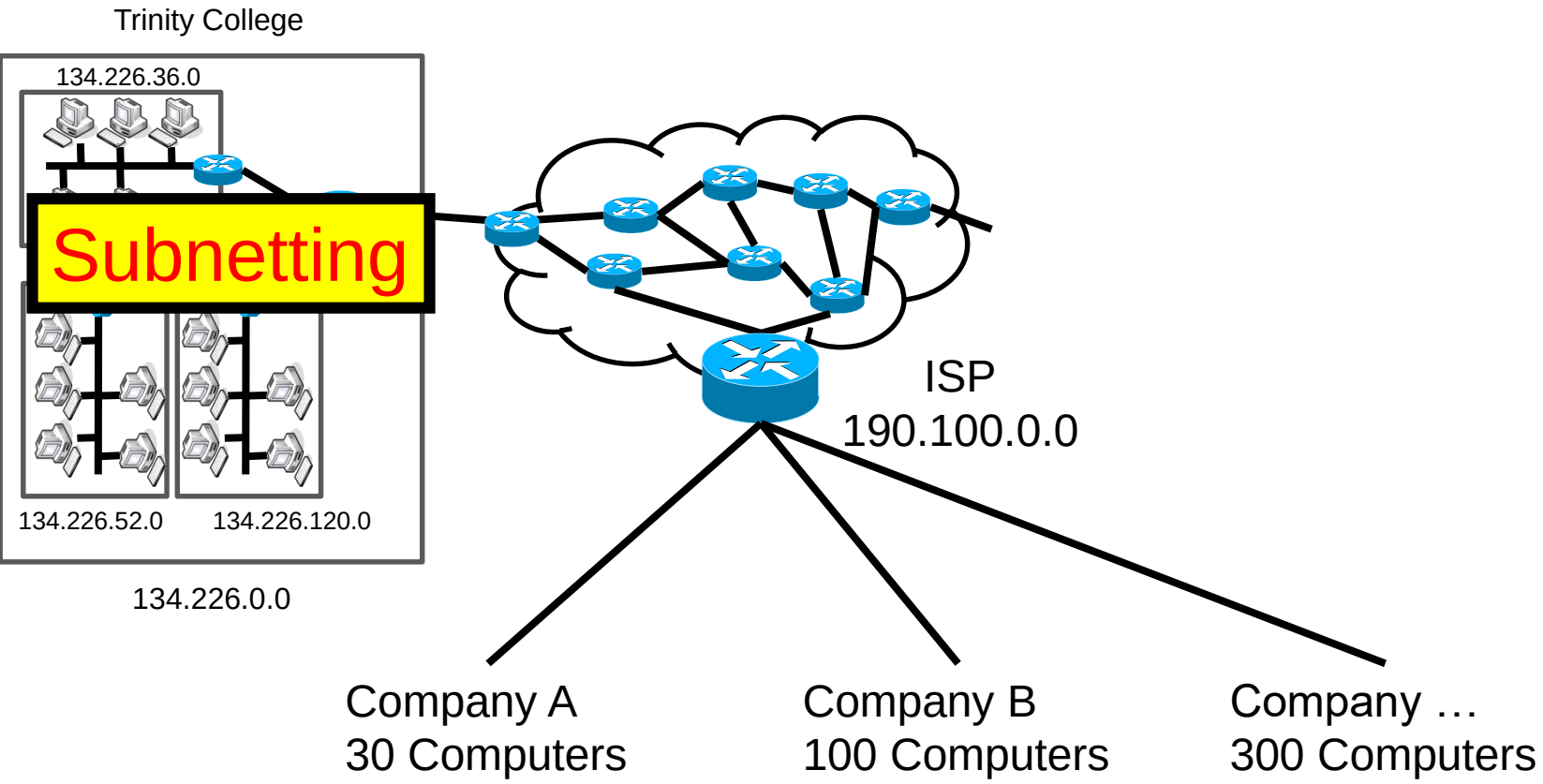
Trinity College



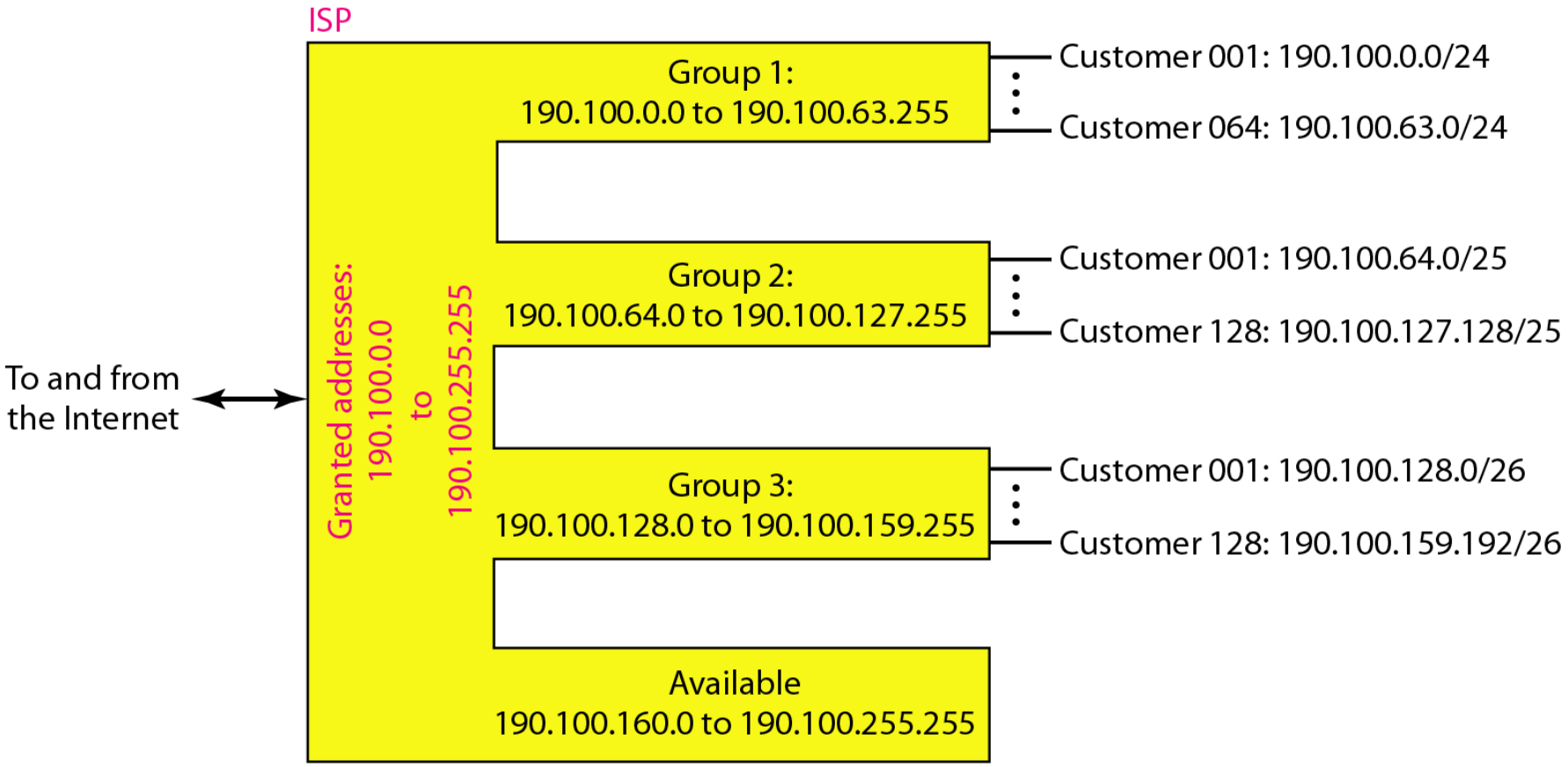
134.226.0.0

Class B ← Classful Addresses

Network Layer

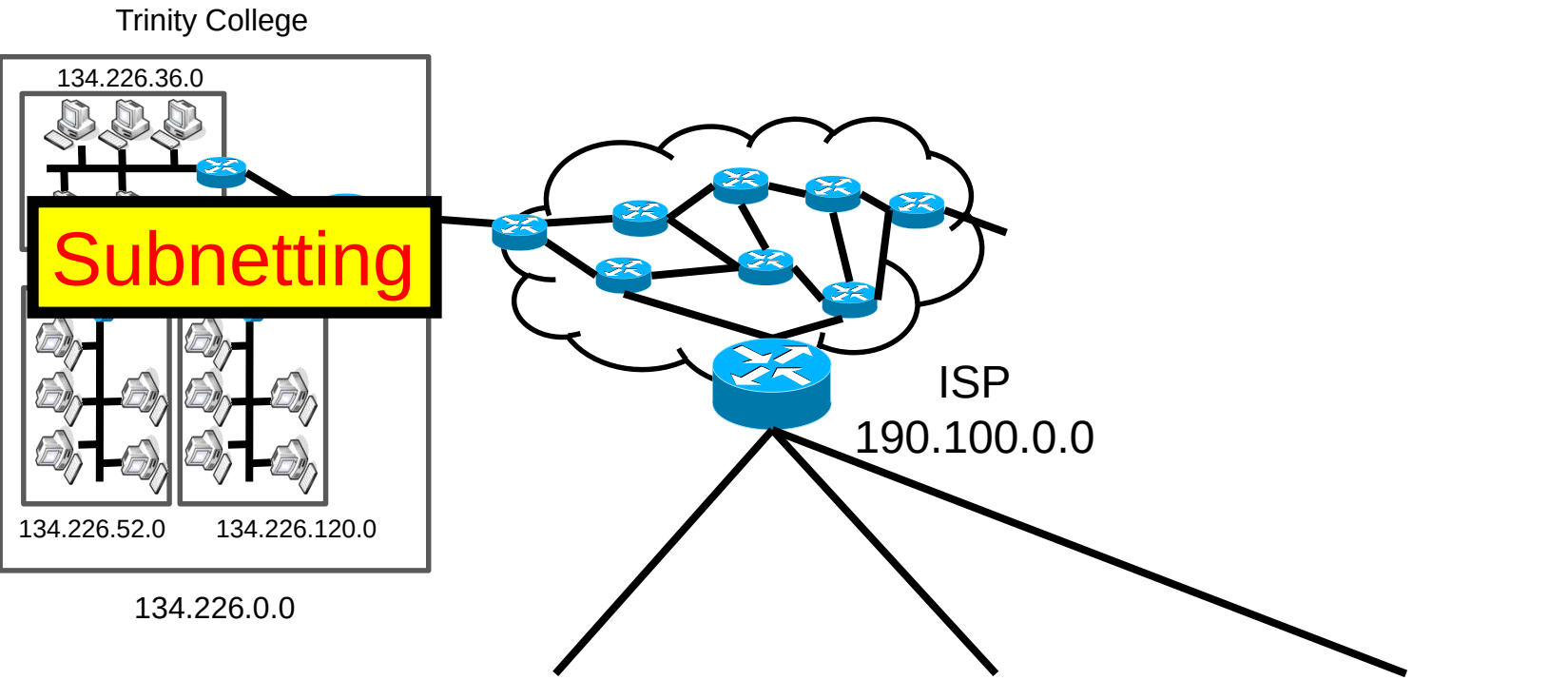


ISPs & Classless Addresses

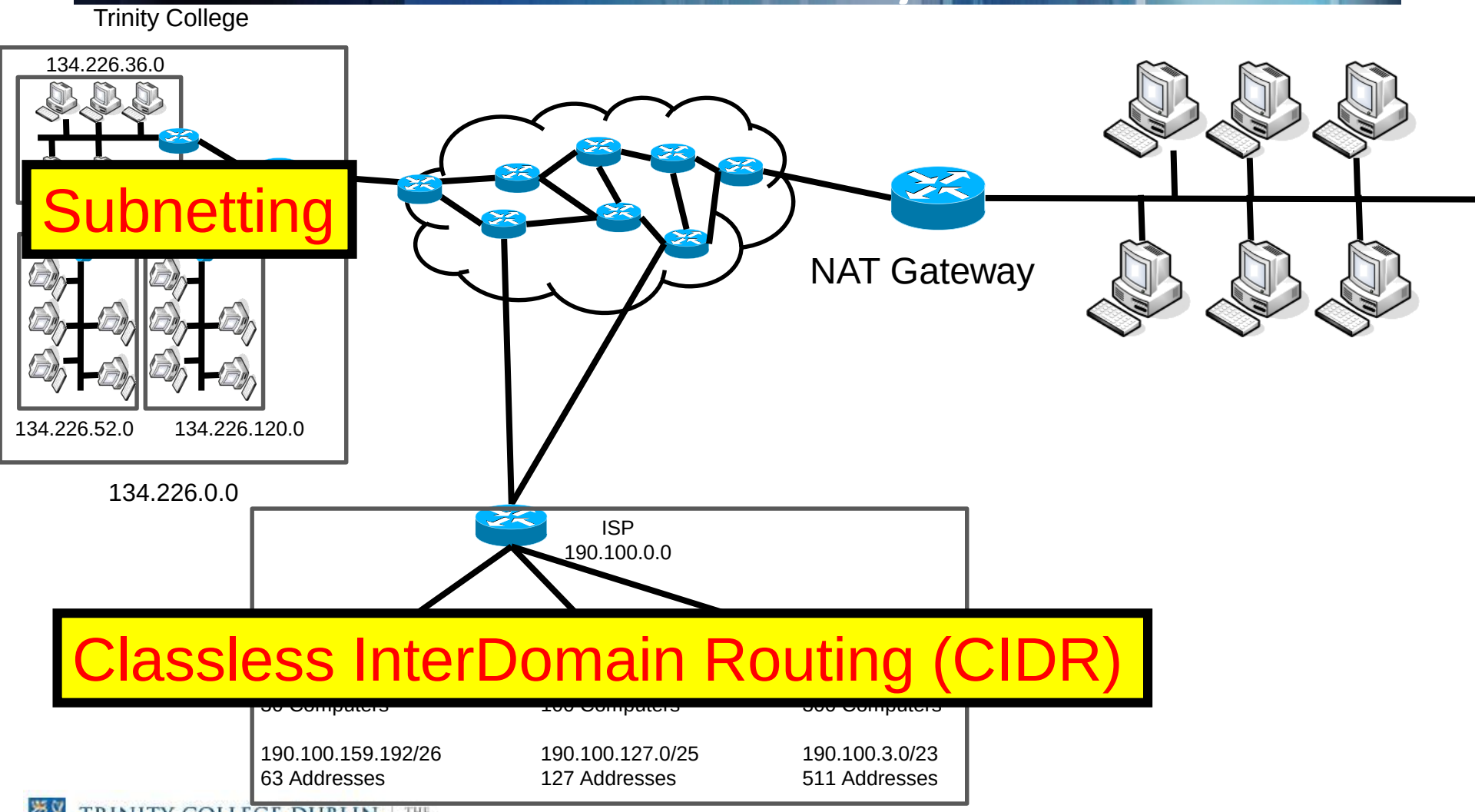


* Figure is courtesy of B. Forouzan

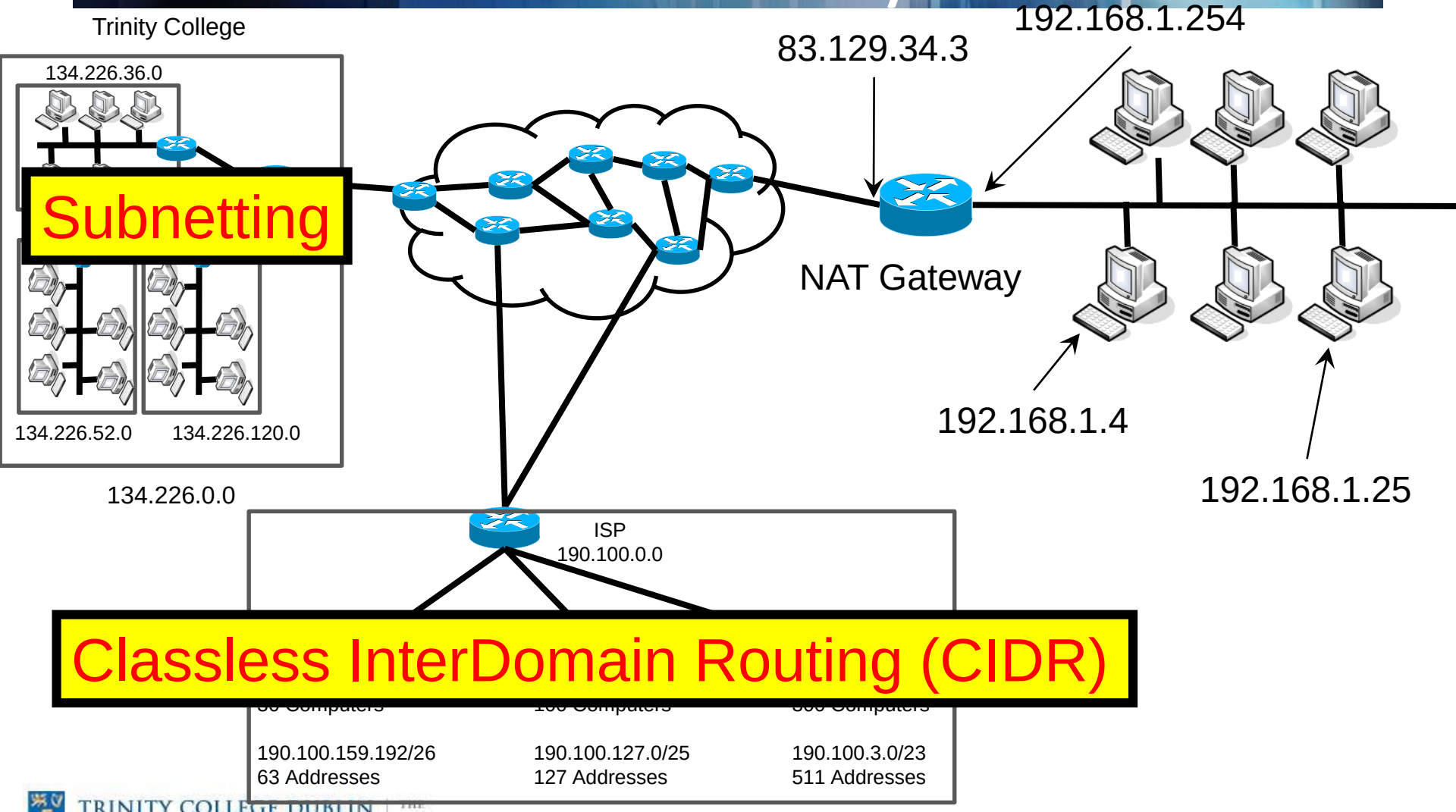
Network Layer



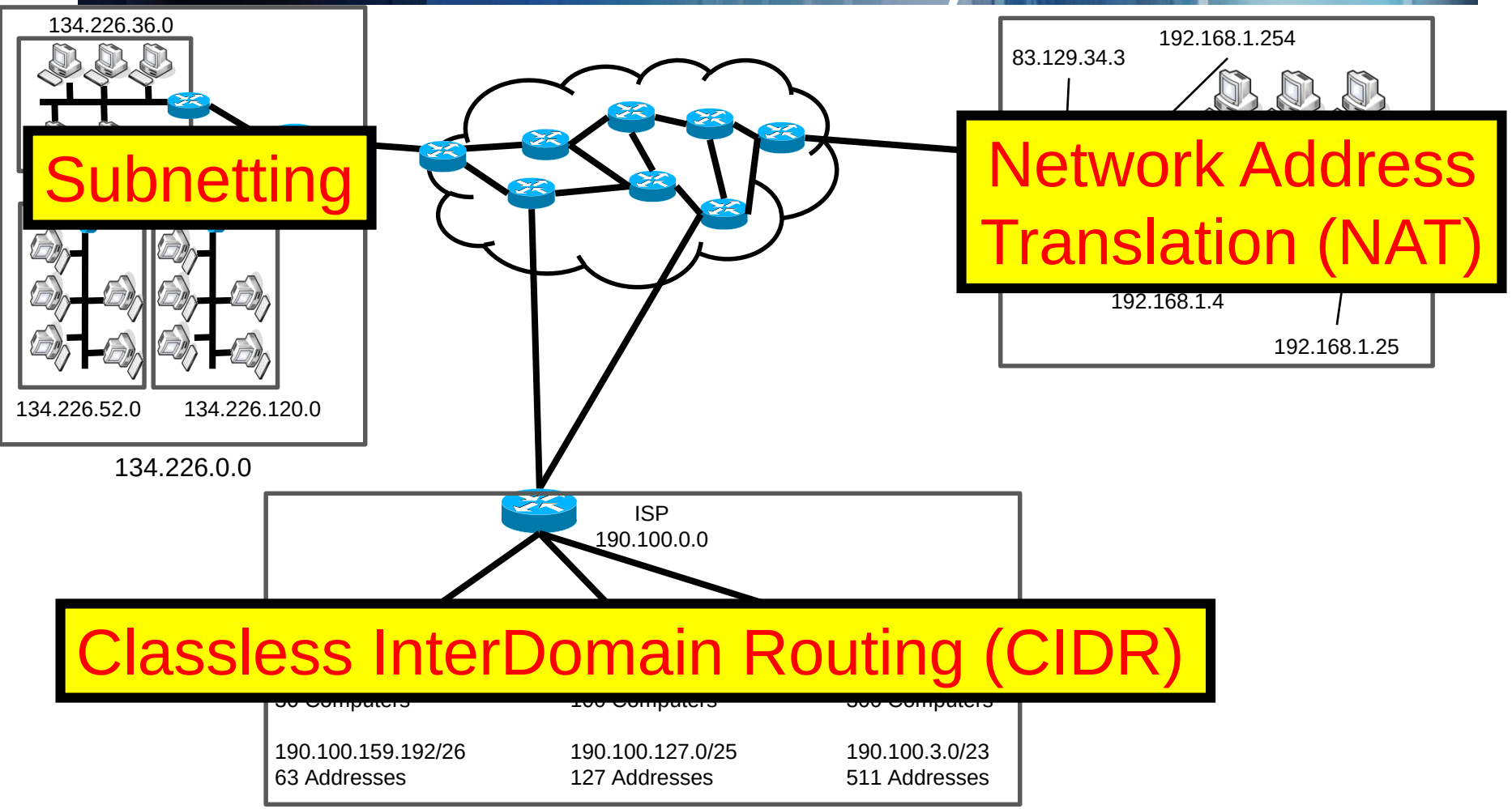
Network Layer



Network Layer



Network Layer



Summary: Network Layer & Addresses

- Doted-decimal notation
- Classful addresses
  - Classes A, B, and C for networks, D for multicast
- Subnetting
- Classless Inter-Domain Routing (CIDR)
  - / notation = significant bits in subnet mask
- Network address translation (NAT)

Internet Protocol (IP)

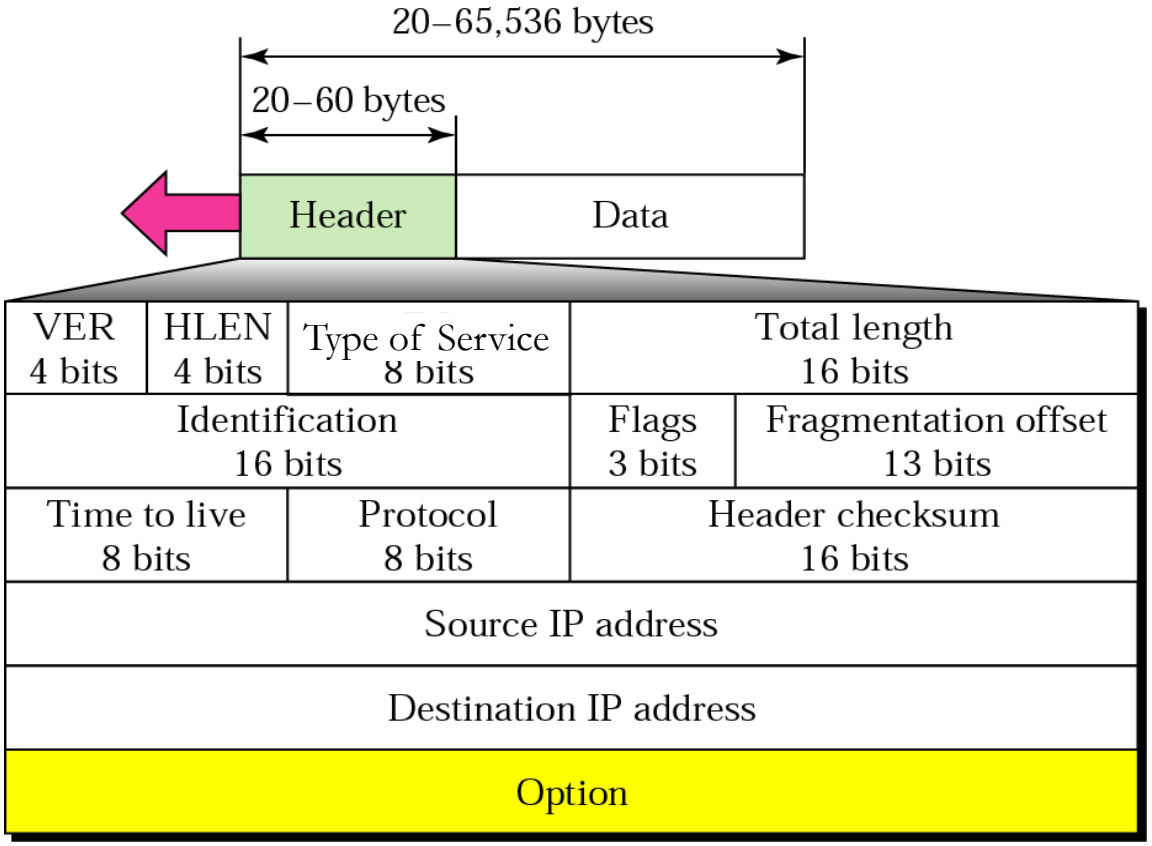
Fragmentation

IP Datagram Format

- The unit of IP data delivery is called a datagram
- Datagrams can have different sizes
- Includes header area and data area

- Header area generally 20 bytes
  - but can be more depending on options

IP Datagram

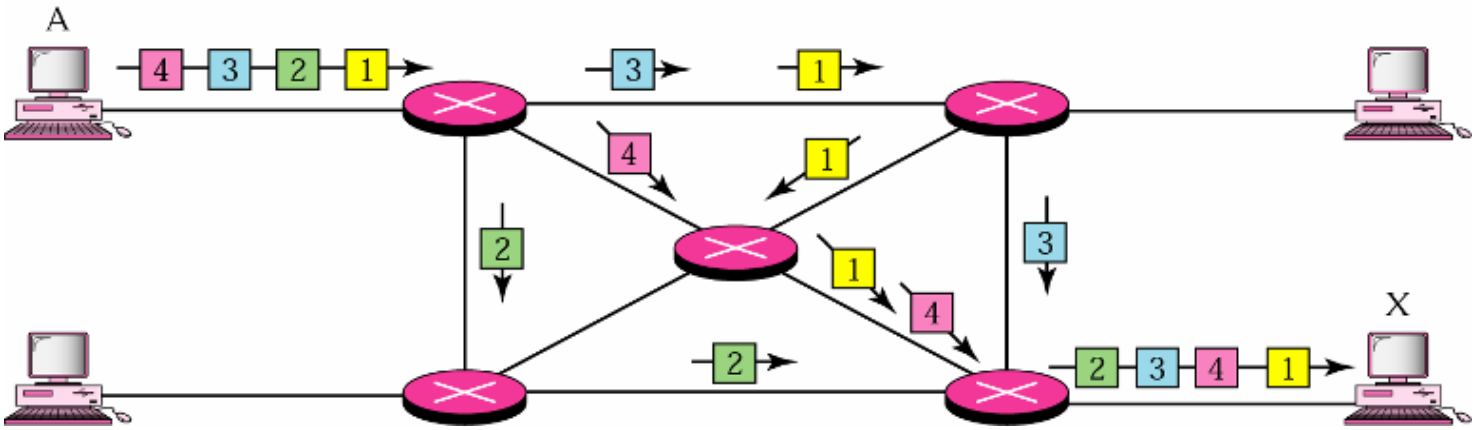


- The total length field defines the total length of the datagram including the header.

IP Datagram Header Fields

Length in bit	Name	Description
4	Version	Version of IP (4, 6)
4	HLEN	Length of the header
8	Service Type	Sender's preference for services e.g. low latency, etc
16	Total Length	Total length of datagram in bytes/octects
16	Identification	Unique identification for each datagram
4	Flags	Control fragmentation of datagrams
12	Fragment Offset	Used in fragmentation of datagrams
8	TTL	Time to live; decremented at each hop it passes
8	Protocol	Higher-layer protocol such as TCP, UDP, etc
16	Header Checksum	Checksum to verify correctness of header information
32	Source address	IP address of the sender
32	Dest. address	IP address of the destination

IP Service Model



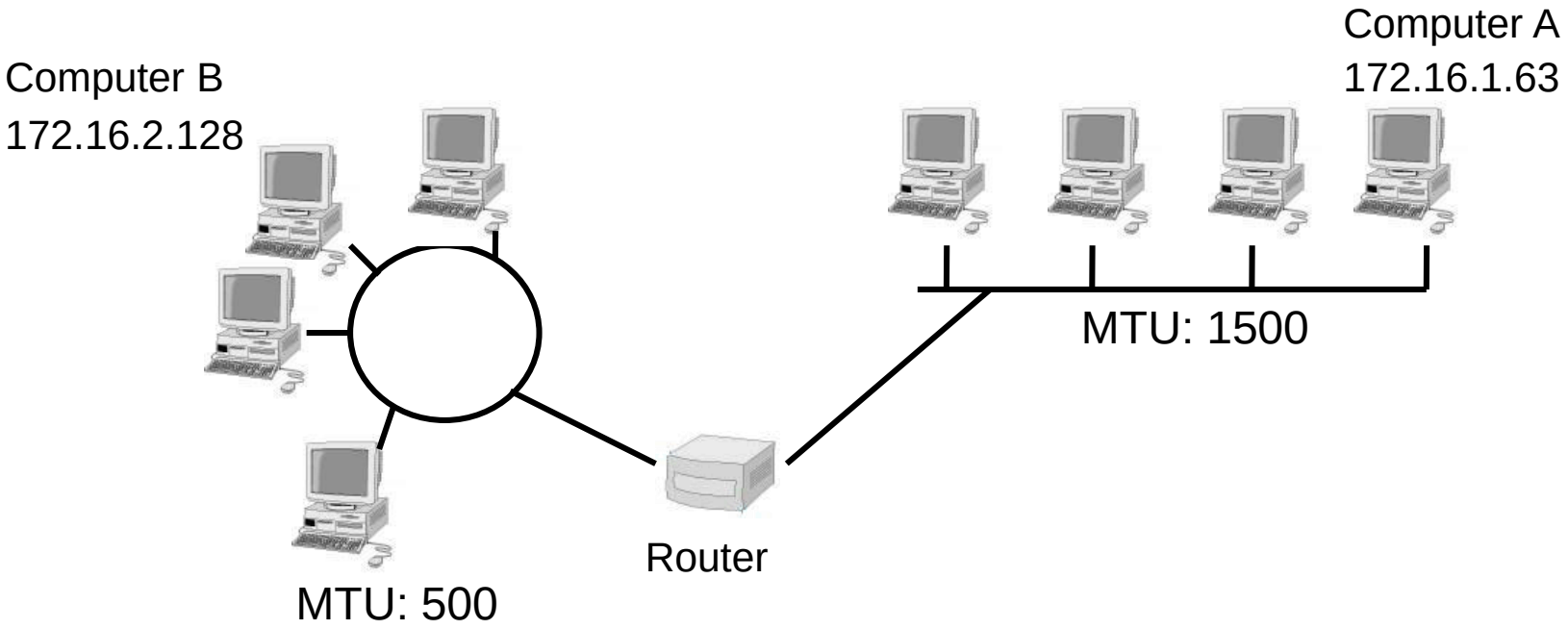
- Connection-less Communication
  - No state is kept about individual packets
- Order not guaranteed
- Best-effort delivery (unreliable service)
  - Packets may be lost
  - Packets may be delivered out of order
  - Duplicate copies of a packet may be delivered
  - Packets can be delayed for a long time

Maximum Transmission Unit (MTU)

- Maximum size of a data unit depends on underlying hardware architecture
- Maximum frame size determines *Maximum Transmission Unit (MTU)*

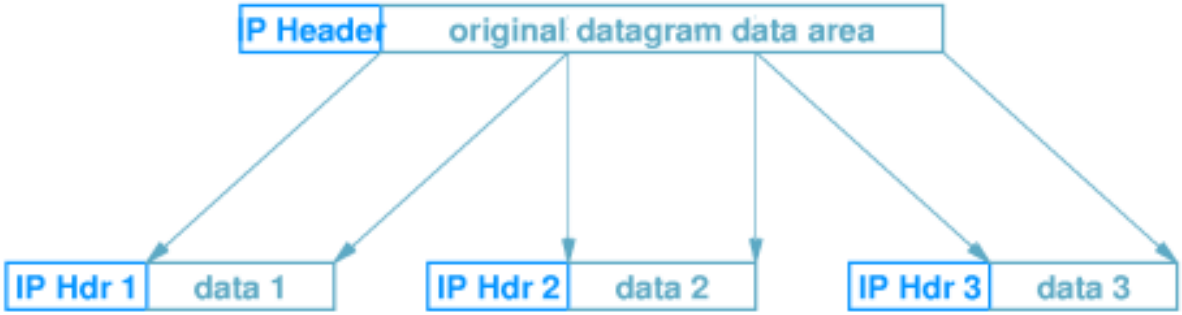
Protocol	*MTU*
Hyperchannel	65,535
Token Ring (16 Mbps)	17,914
Token Ring (4 Mbps)	4,464
FDDI	4,352
Ethernet	1,500
X.25	576
PPP	296

IP & Network Architectures



- IP datagrams may be larger than most MTUs of underlying network architectures
- Maximum frame size limits maximum size of an IP datagram for a network

Fragmentation

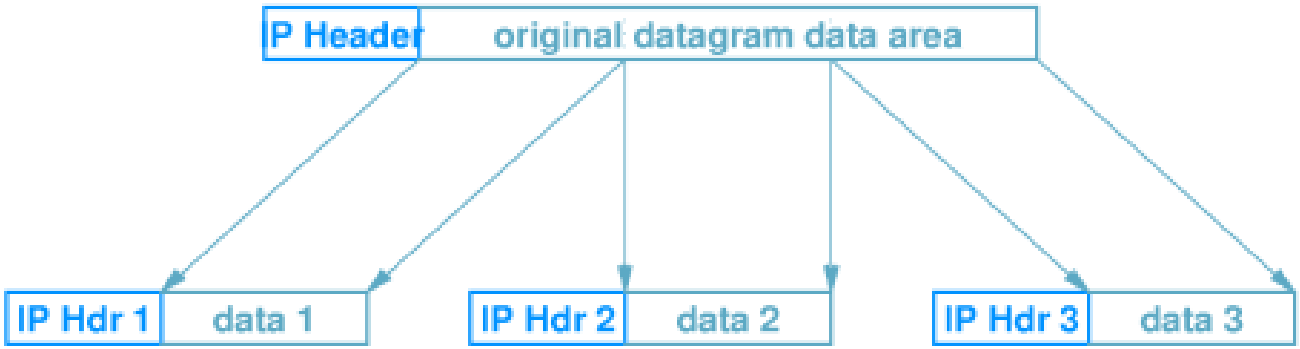


- Possible techniques:
  - Limit datagram size to smallest MTU of any network
  - Adjust datagram size as packet progresses through networks
    - Router detects datagram larger than network MTU and splits into pieces

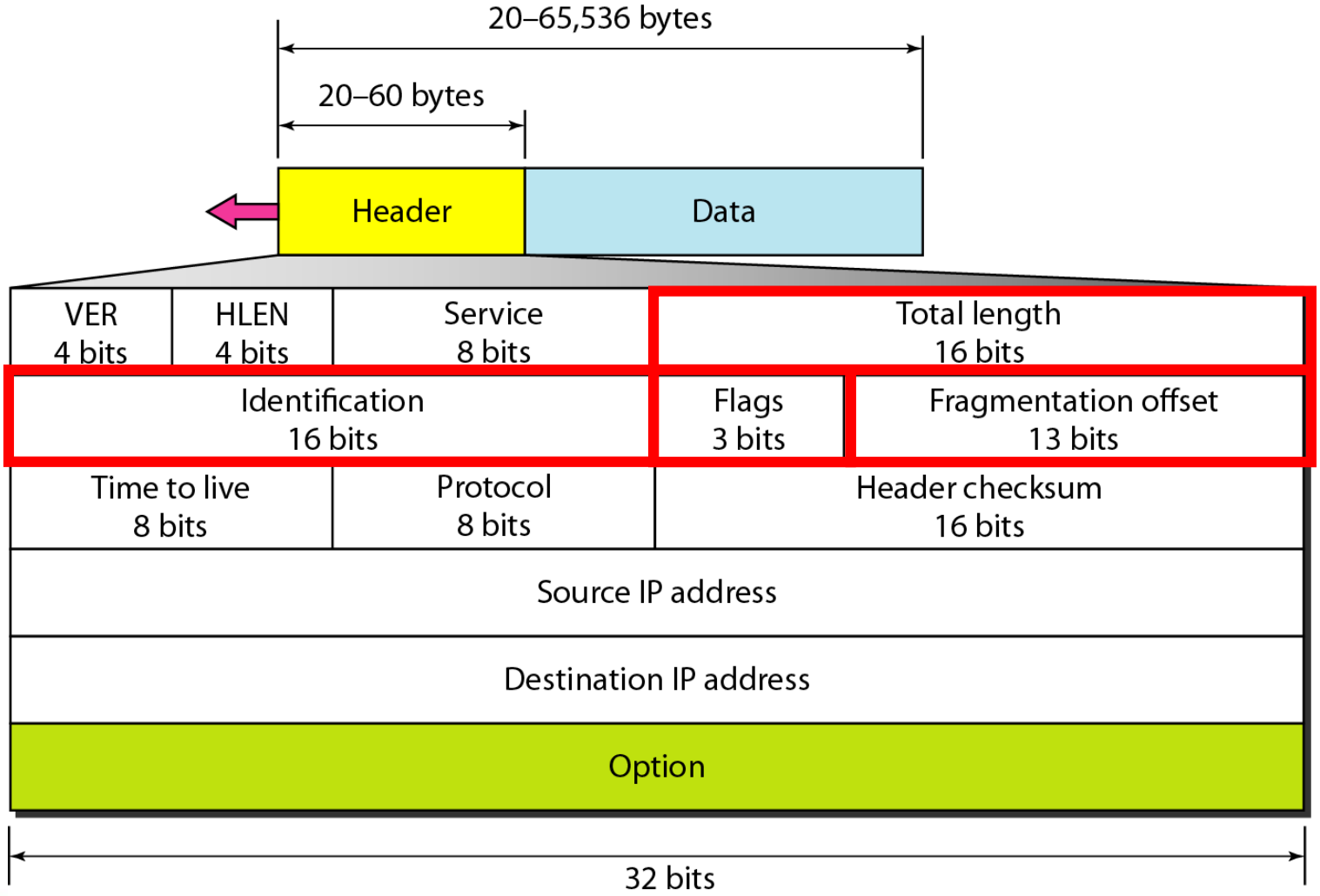
- IP Strategy
  - Fragment when necessary (Datagram > MTU)
  - Try to avoid fragmentation at source host
  - Re-fragmentation is possible
  - Delay reassembly until destination host
  - Do not recover from lost fragments
    - If one fragment is lost all fragments are discarded

Fragmentation (details)

- Each fragment is an independent datagram
  - Includes all header fields
  - Bit in header indicates datagram is a fragment
  - Other fields have information for reconstructing original datagram
  - FRAGMENT OFFSET gives original location of fragment



Fragmentation & IPv4 Header

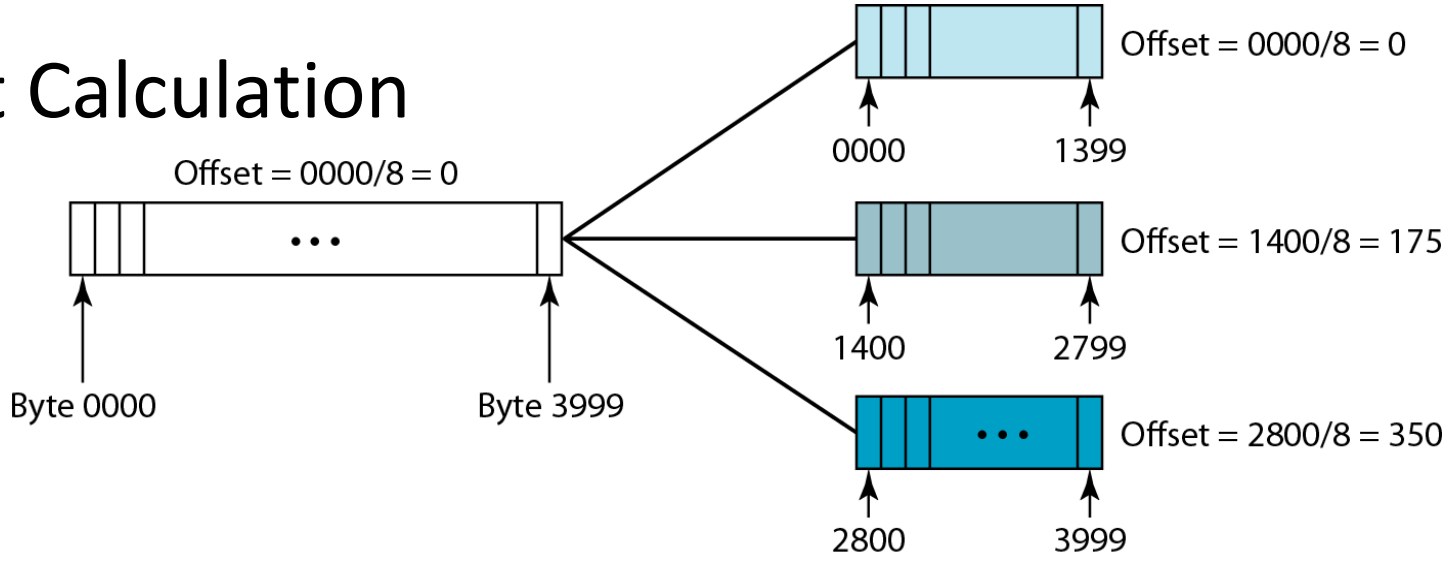


* Figure is courtesy of B. Forouzan

Header Fields

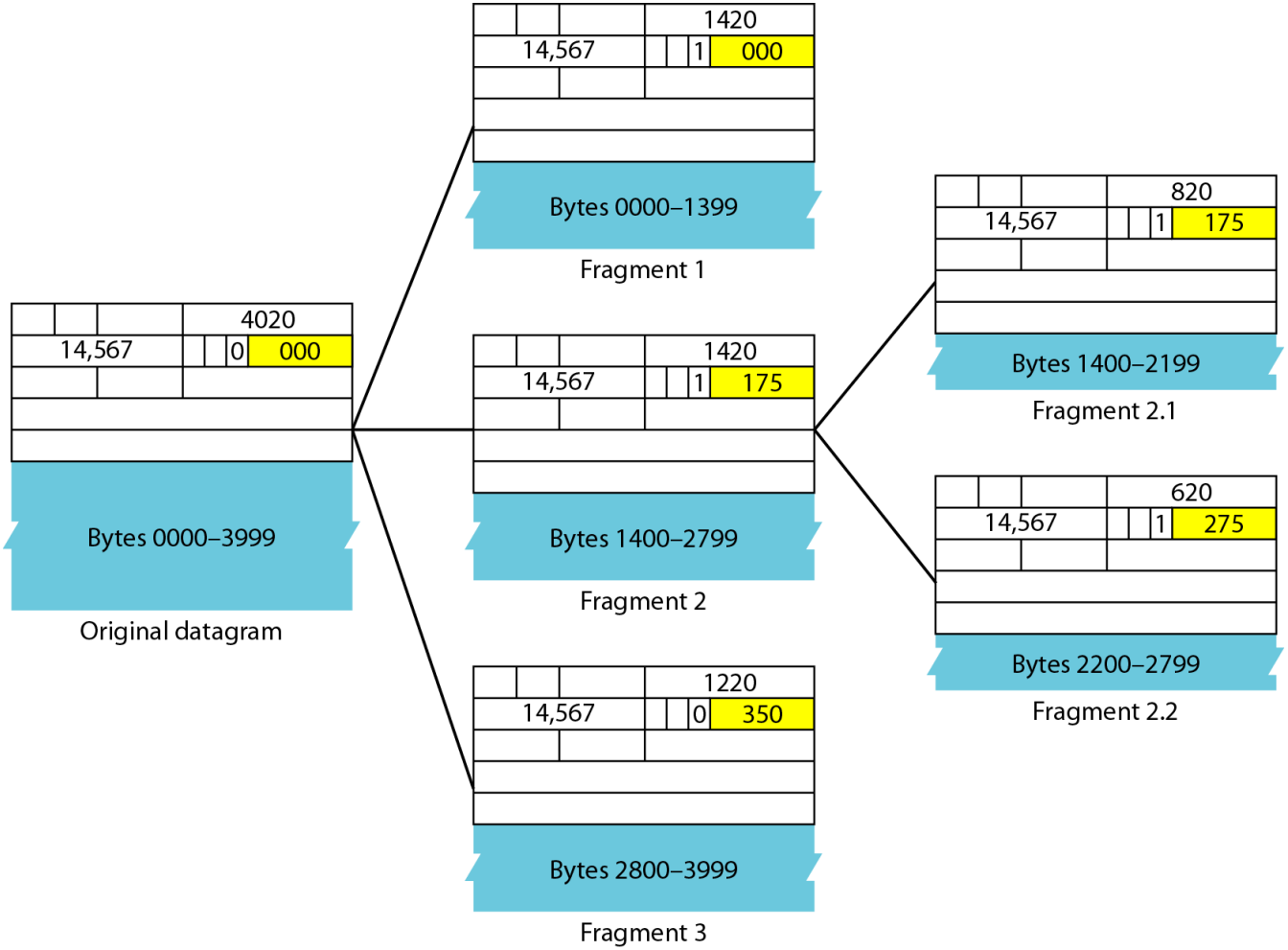
- "Do not fragment"-Request
- More Fragments

- Offset Calculation



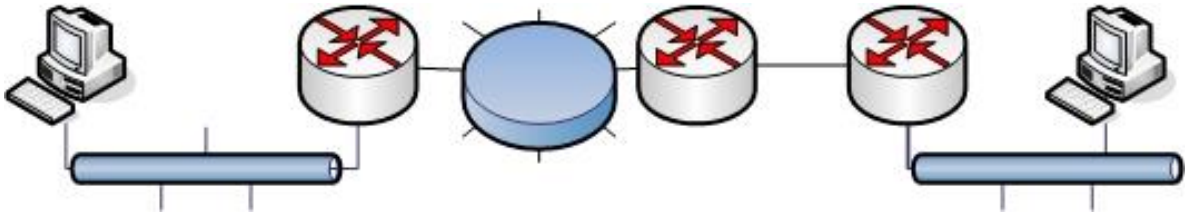
* Figure is courtesy of B. Forouzan

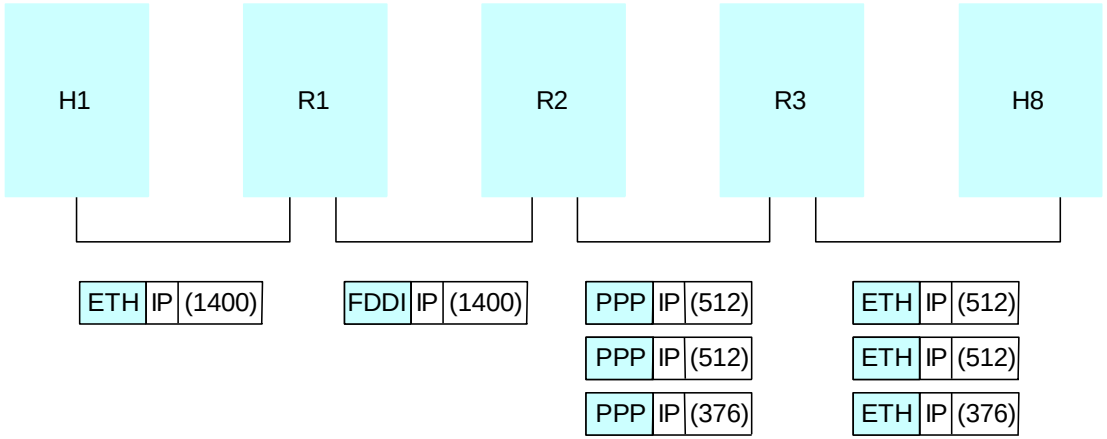
Fragmentation Example I



* Figure is courtesy of B. Forouzan

Fragmentation Example II





Start of header				
Ident= x			0	Offset= 0
Rest of header				
1400 data bytes				

Start of header				
Ident= x			1	Offset= 0
Rest of header				
512 data bytes				

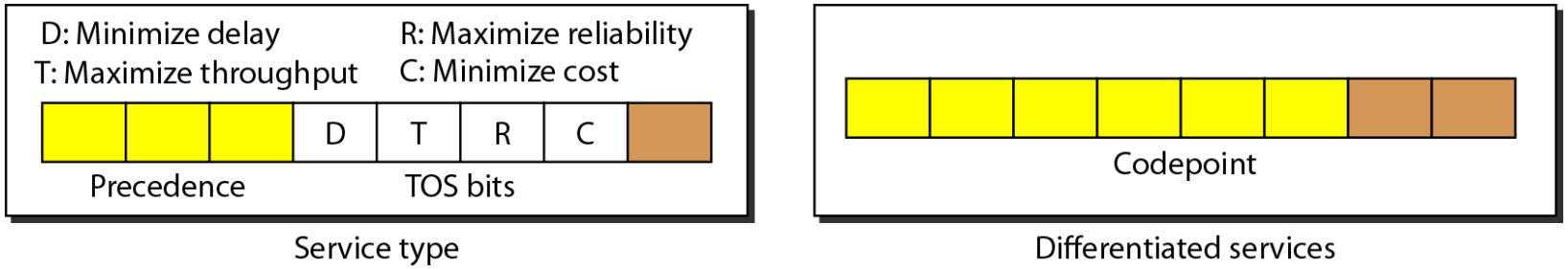
Start of header				
Ident= x			1	Offset= 64
Rest of header				
512 data bytes				

Start of header				
Ident= x			0	Offset= 128
Rest of header				
376 data bytes				

Fragment loss

- A fragment may be lost/dropped in transfer
- What happens to original datagram?
  - Destination drops entire original datagram
- How does destination identify lost fragment?
  - Sets timer with each fragment
  - If timer expires before all fragments arrive, fragment assumed lost
  - Datagram dropped
- Source is assumed to retransmit

Service types



TOS Bits	*Description*
0000	Normal (default)
0001	Minimize cost
0010	Maximize reliability
0100	Maximize throughput
1000	Minimize delay

* Figure is courtesy of B. Forouzan

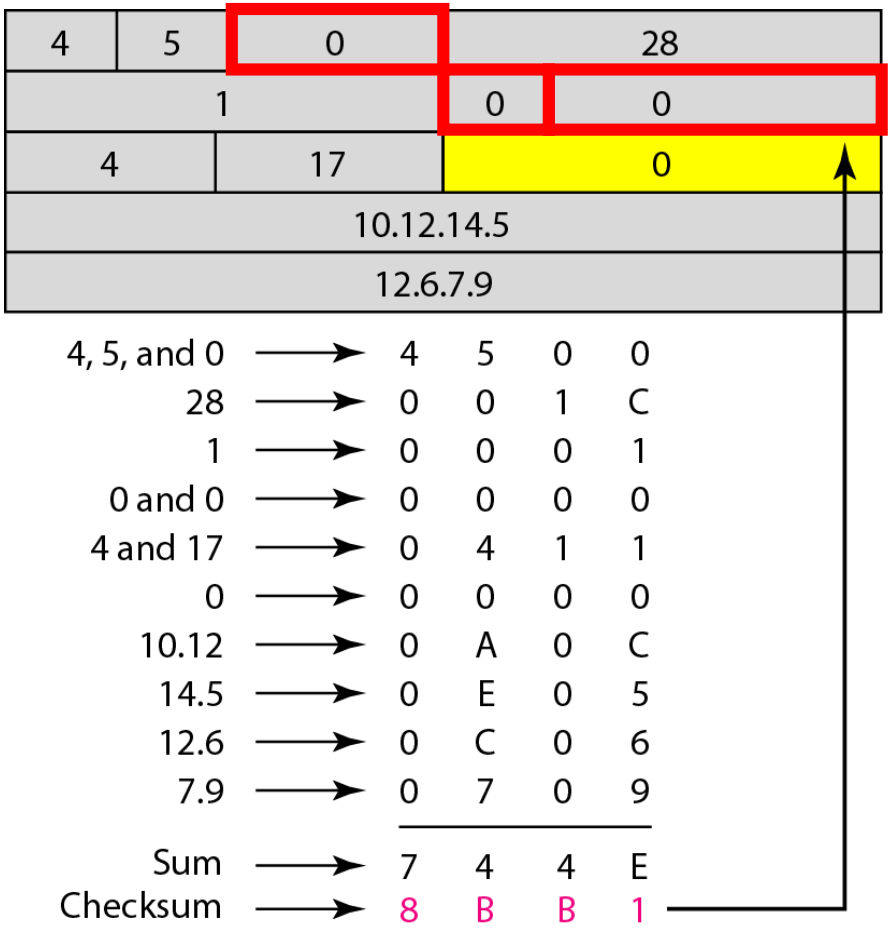
Protocols & TOS Bits

Protocol	*TOS Bits*	*Description*
ICMP	0000	Normal
BOOTP	0000	Normal
NNTP	0001	Minimize cost
IGP	0010	Maximize reliability
SNMP	0010	Maximize reliability
TELNET	1000	Minimize delay
FTP (data)	0100	Maximize throughput
FTP (control)	1000	Minimize delay
TFTP	1000	Minimize delay
SMTP (command)	1000	Minimize delay
SMTP (data)	0100	Maximize throughput
DNS (UDP query)	1000	Minimize delay
DNS (TCP query)	0000	Normal
DNS (zone)	0100	Maximize throughput

* Figure is courtesy of B. Forouzan

Checksum in IPv4 Header

- Calculation omits:
  - Service field
  - Fragment fields and offset
  - Checksum field

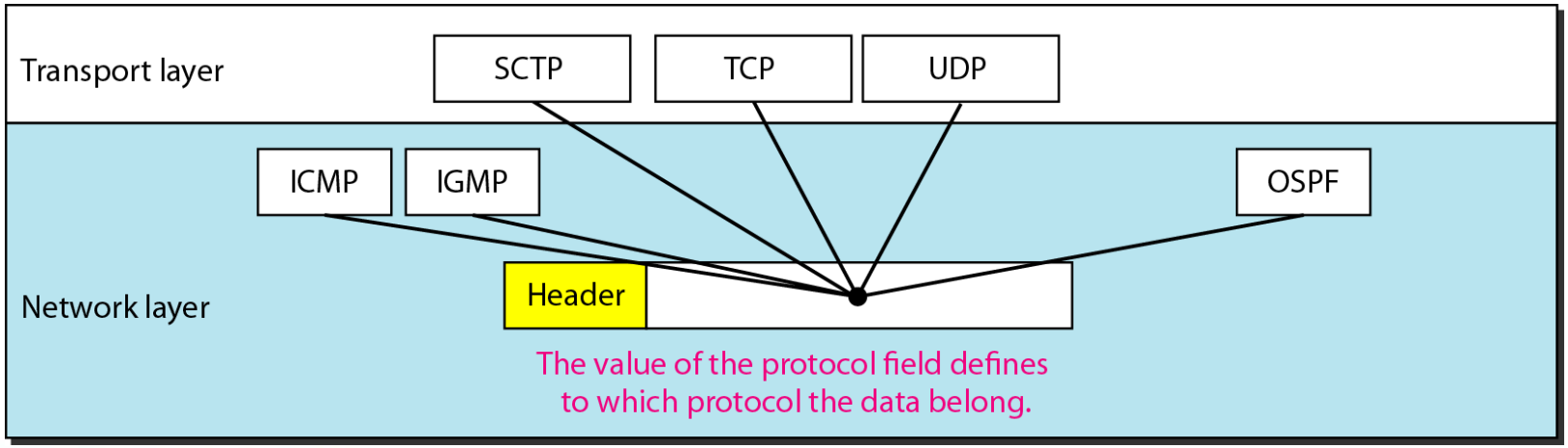


* Figure is courtesy of B. Forouzan

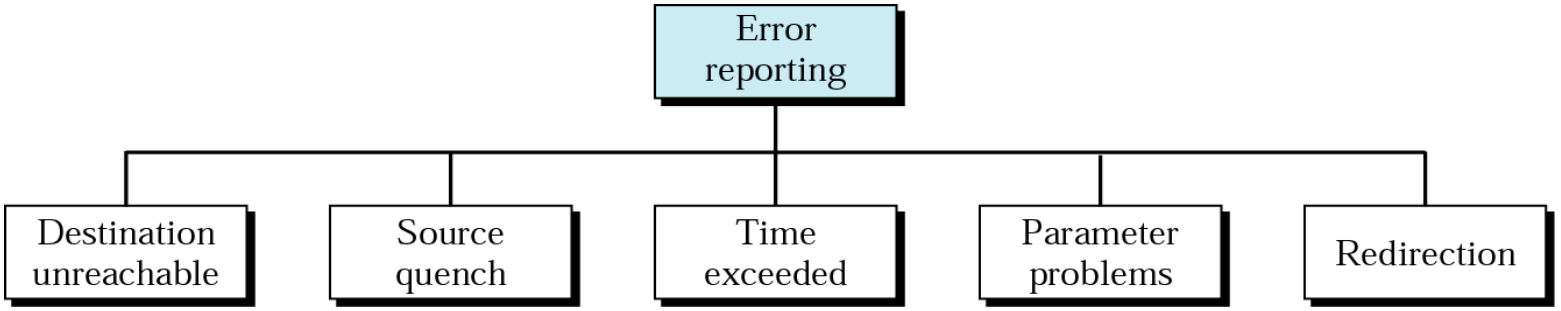
Protocol Field

- Specifies next protocol in stack

Value	*Protocol*
1	ICMP
2	IGMP
6	TCP
17	UDP
89	OSPF

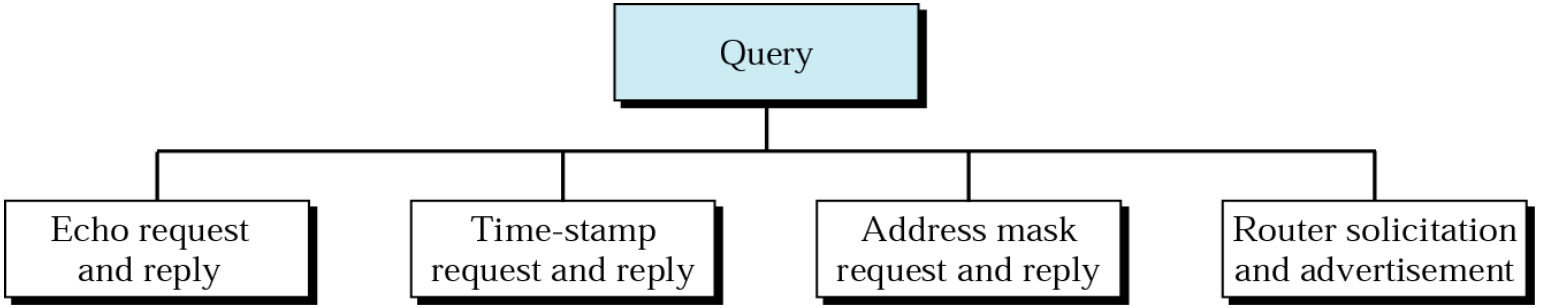


Internet Control Message Protocol



- Destination unreachable
- Source quench
  - Mechanism for destination and intermediate nodes to limit traffic from source
- Time exceeded
  - Send when datagram discarded due TTL value of 0
- Parameter problems
  - Send when datagram discarded due to parameter ambiguity
- Redirection
  - Send from router to update routing table of source

ICMP Queries



Ping

uses the ICMP protocol's mandatory ECHO_REQUEST datagram to elicit an ICMP ECHO_RESPONSE from a host or gateway. ECHO_REQUEST datagrams have an IP and ICMP header, followed by a struct timeval and then an arbitrary number of bytes used to fill out the packet.

```
$ ping www.tcd.ie
PING dux6.tcd.ie (134.226.1.61): 56 data bytes
64 bytes from 134.226.1.61: icmp_seq=0 ttl=64 time=0.781 ms
64 bytes from 134.226.1.61: icmp_seq=1 ttl=64 time=0.466 ms
64 bytes from 134.226.1.61: icmp_seq=2 ttl=64 time=0.490 ms
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.457/0.548/0.781/0.135 ms
```

Summary: Internet Protocol

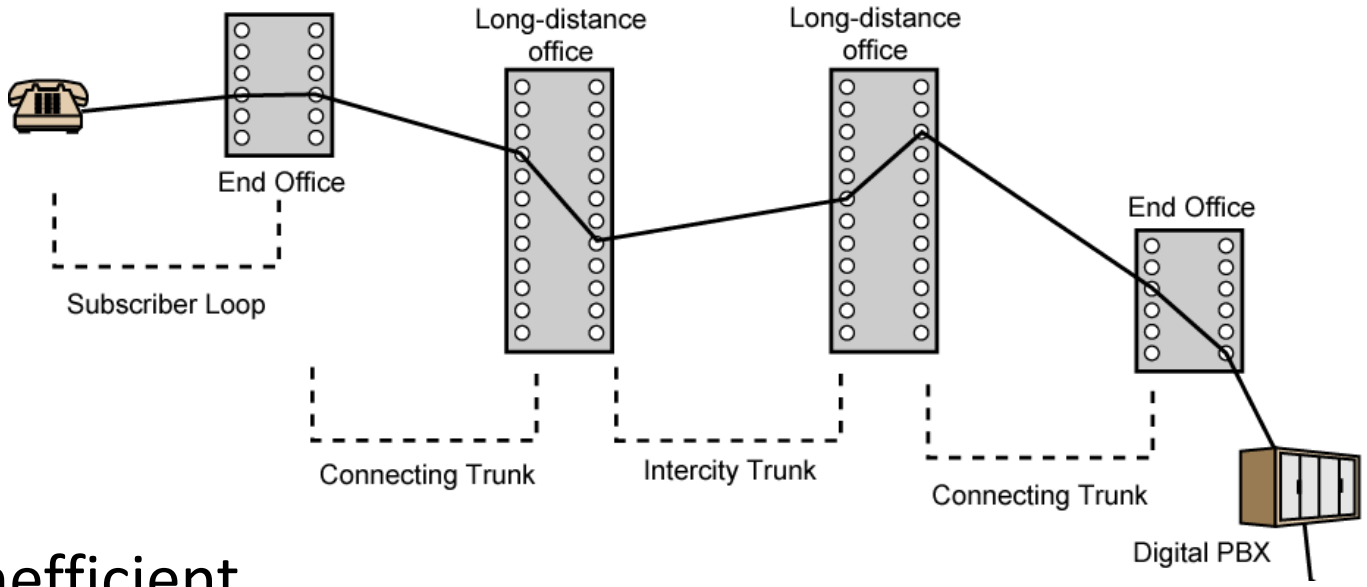
- IP Service Model
  - Connection-less, no order guaranteed
- IP Header
  - 20 bytes + options
- Fragmentation
  - Datagrams split into fragments to fit MTUs
  - Only re-assembled at destination
- Internet Control Message Protocol (ICMP)
  - Error Reporting e.g. source quench
  - Querying e.g. Ping


That's all folks
Telecomms II

CS2031
Telecommunications II

Circuit Switching & Packet Switching

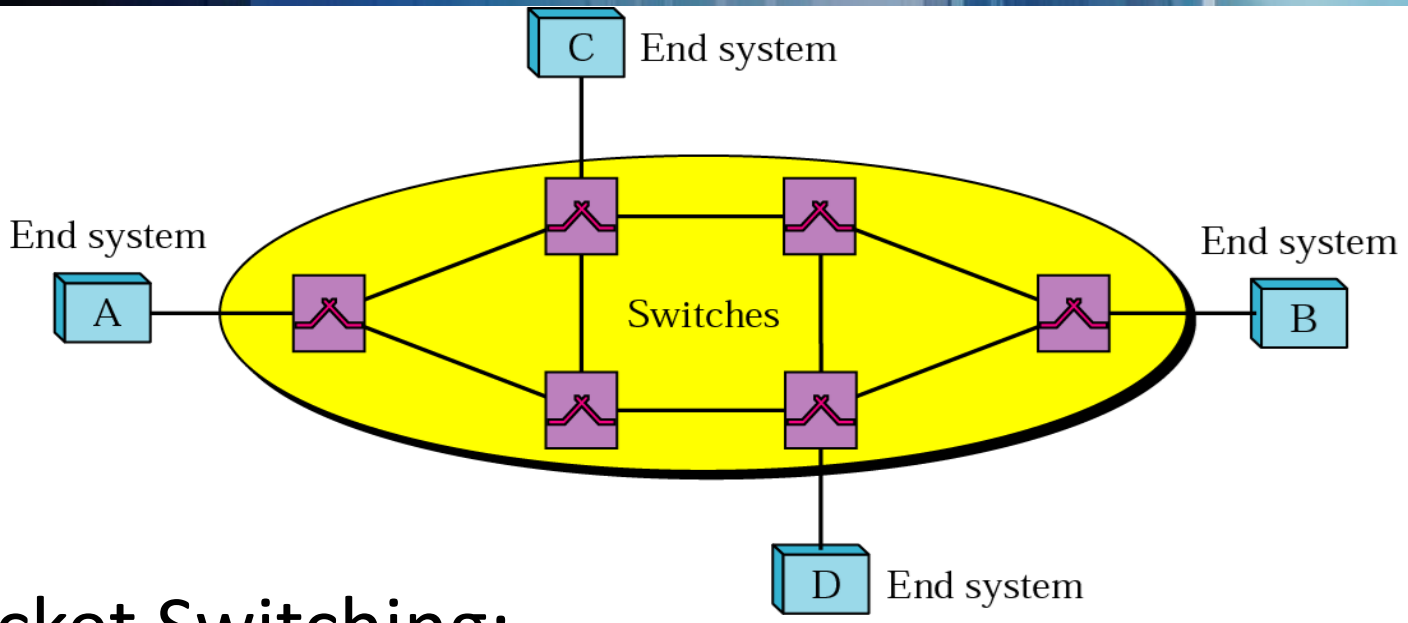
Public Circuit Switched Network



- Inefficient
  - Channel capacity dedicated for duration of connection
  - If no data, capacity wasted
- Set up of connection takes time
- Once connected, transfer is transparent

* Figure is courtesy of A. Tanenbaum

Switched Networks

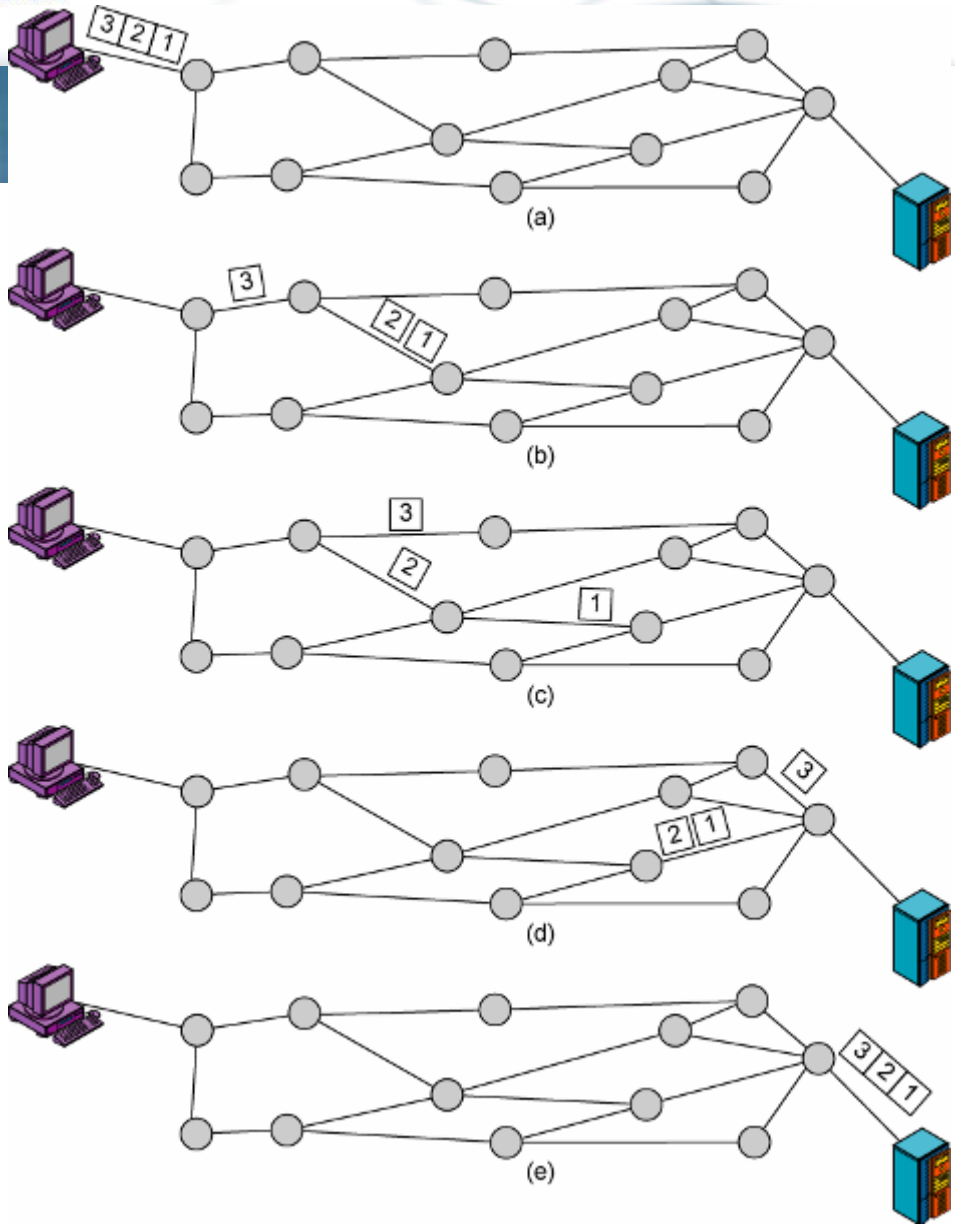


- Packet Switching:
  - Switching decisions are made on individual packets
- Virtual Circuit Switching:
  - A circuit is setup explicitly for individual connections

* Figure is courtesy of B. Forouzan

Packet switching

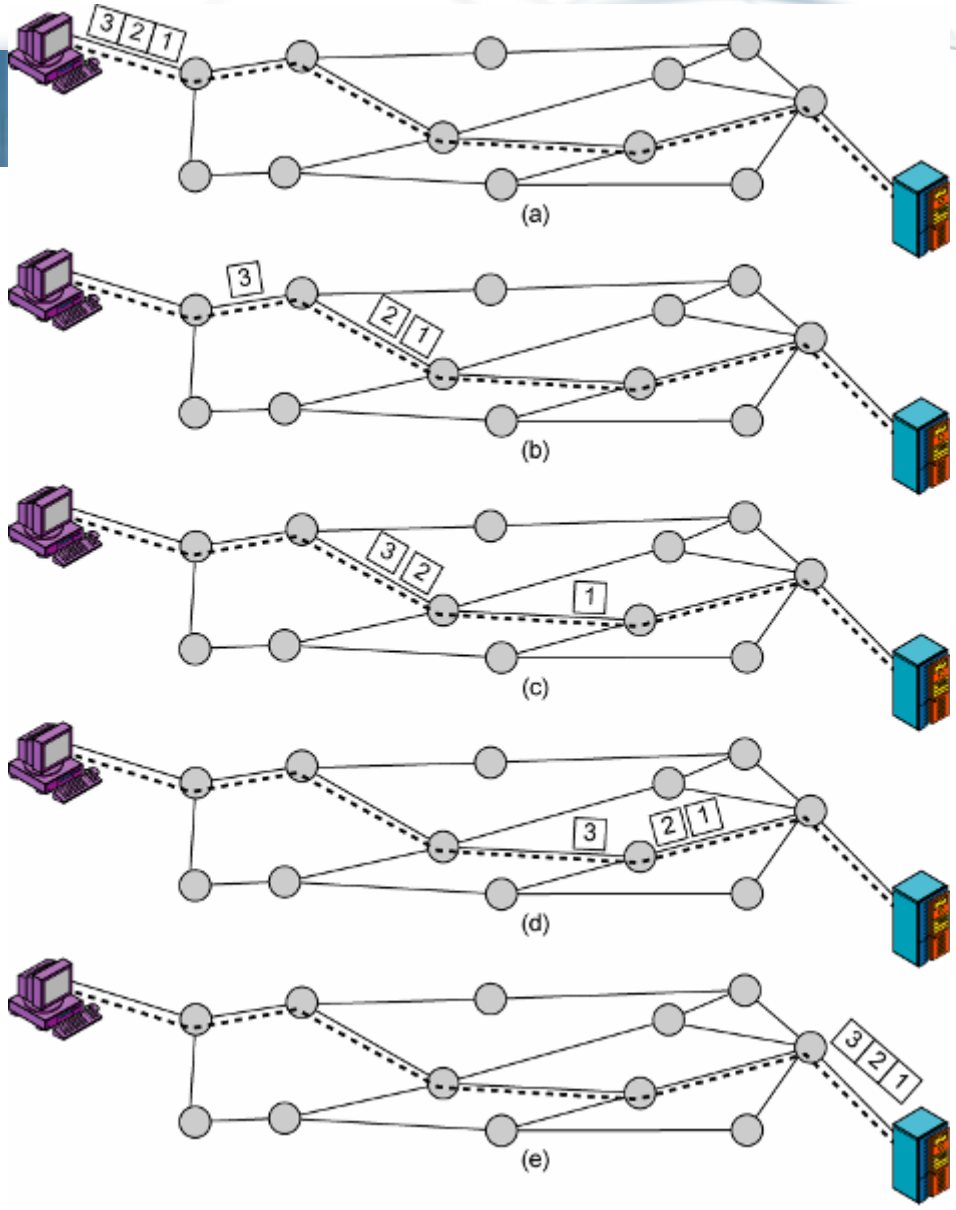
- Frames can be transferred over different paths in the network
- Reliability is generally delegated to higher layers
- Order is not necessarily maintained



* Figure is courtesy of B. Forouzan

Virtual Circuits

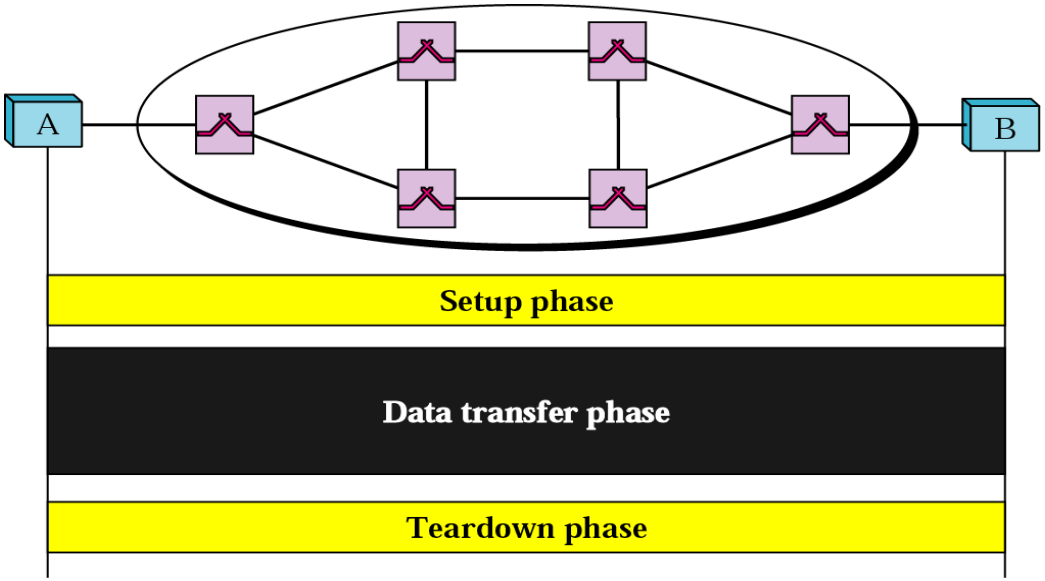
- Connection-oriented communication
- Connection is established before communication
- The network maintains order



* Figure is courtesy of B. Forouzan

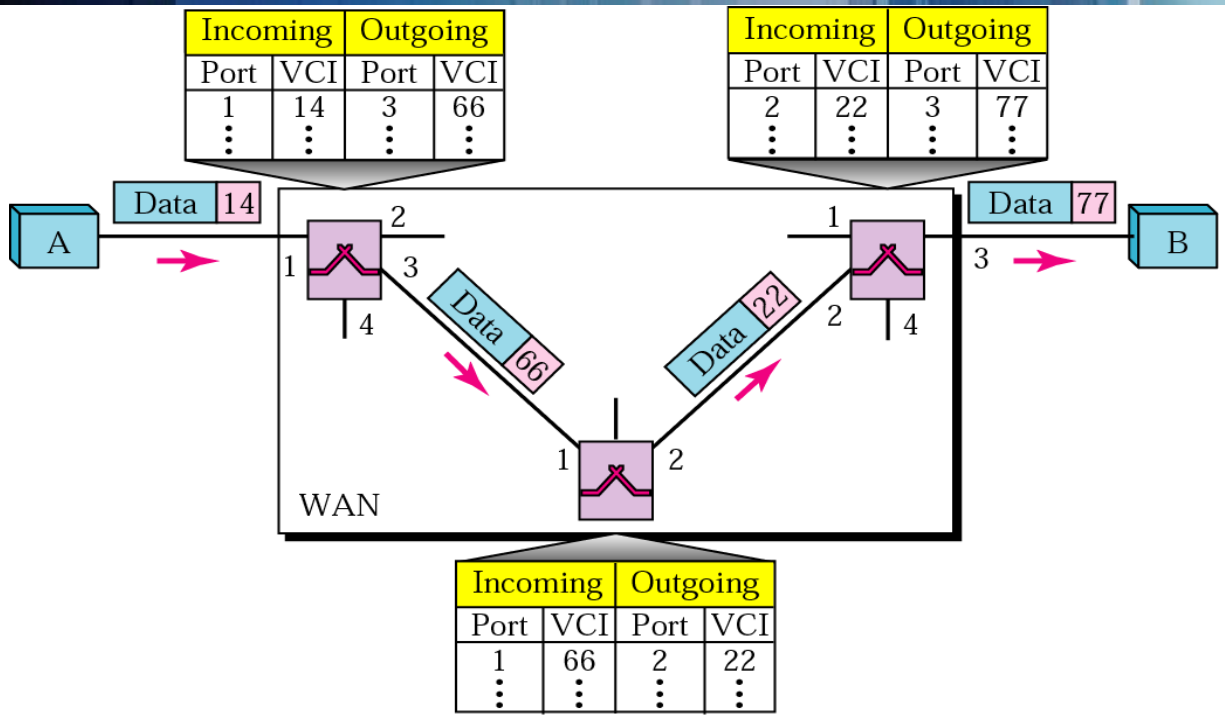
Phases for Virtual Circuit Communication

- Three phases
  - Connection setup
  - Data Transfer
  - Connection termination



* Figure is courtesy of B. Forouzan

Virtual Circuit Switching

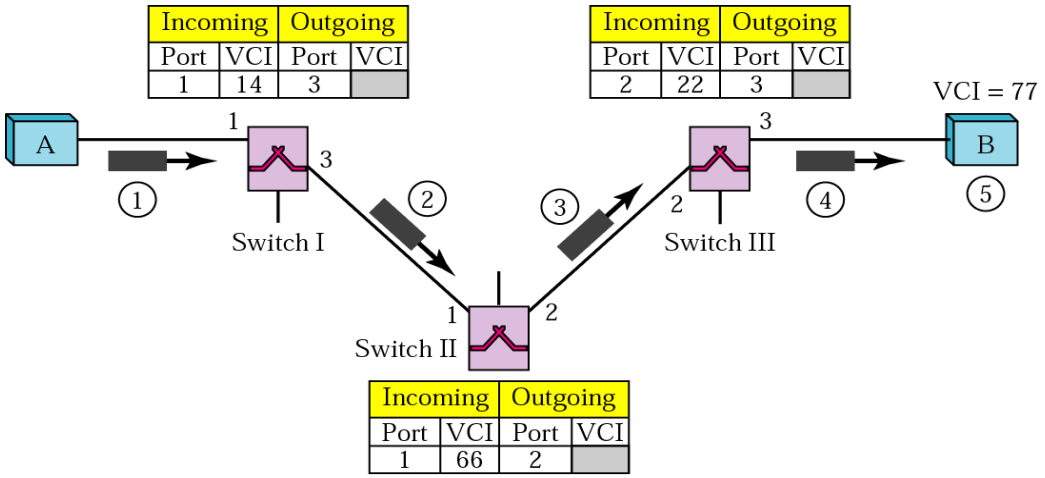


- Every switch maintains a table
  - For duration of communication one entry for incoming and outgoing line
  - Incoming and outgoing line are identified by port number and virtual circuit identifier

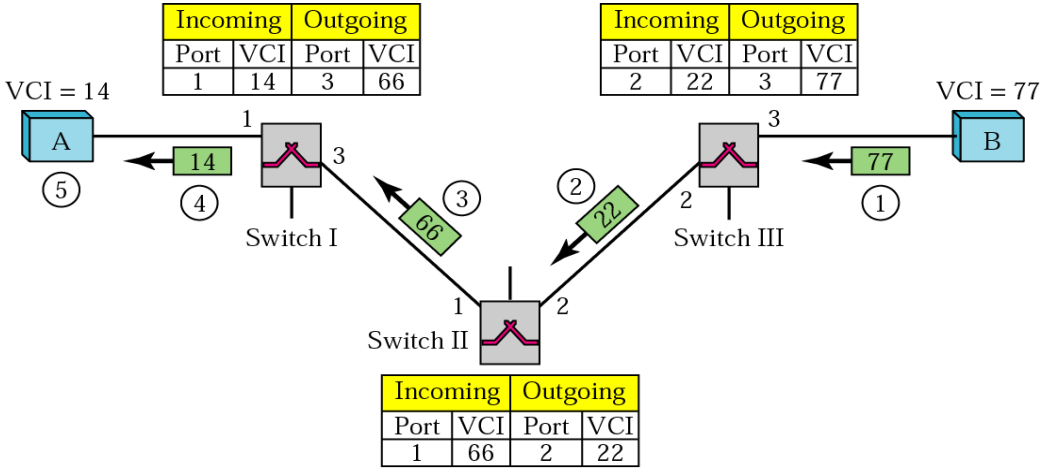
* Figure is courtesy of B. Forouzan

Setup Phase

- Setup request

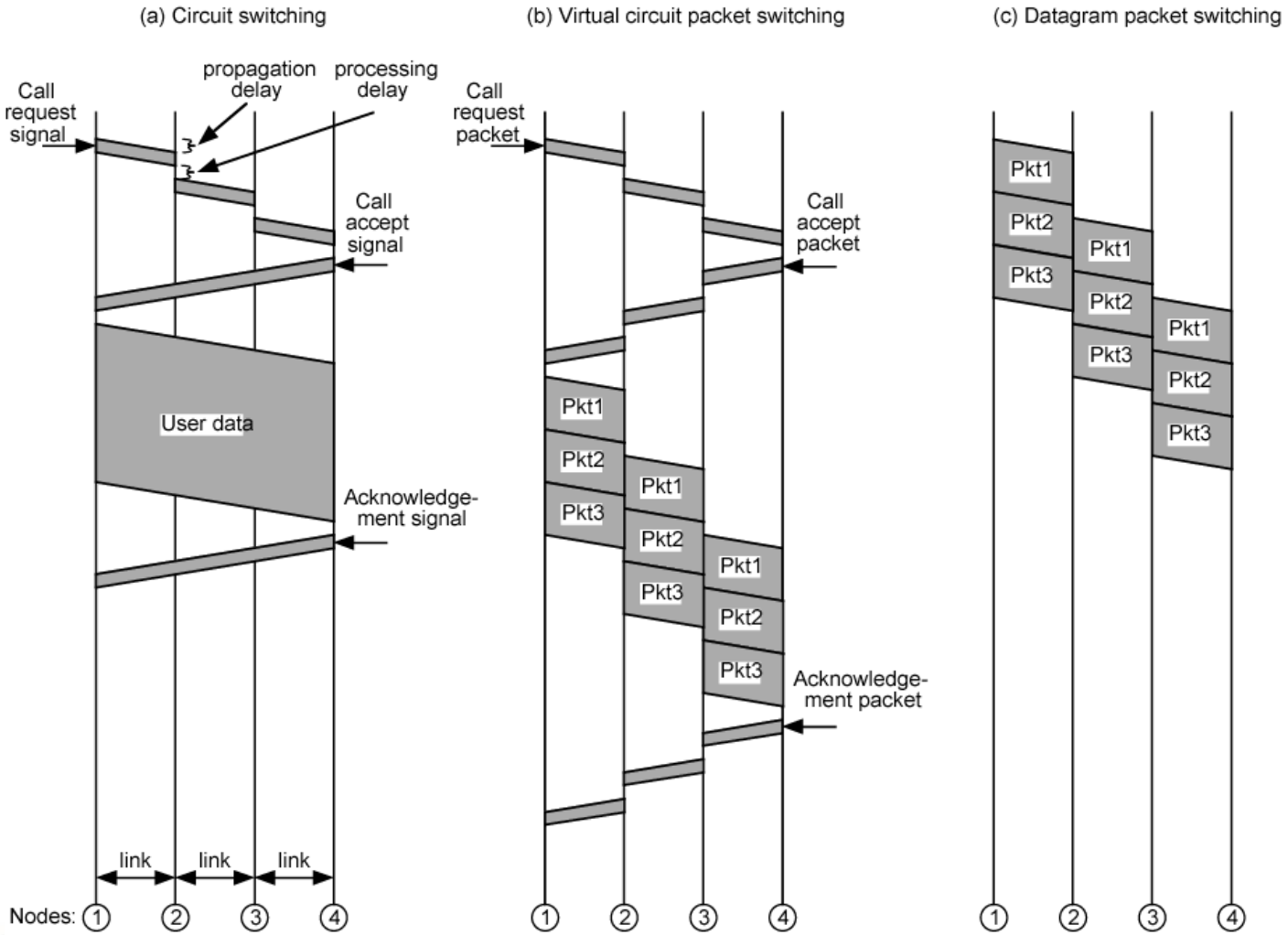


- Setup acknowl.



* Figure is courtesy of B. Forouzan

Event Timing

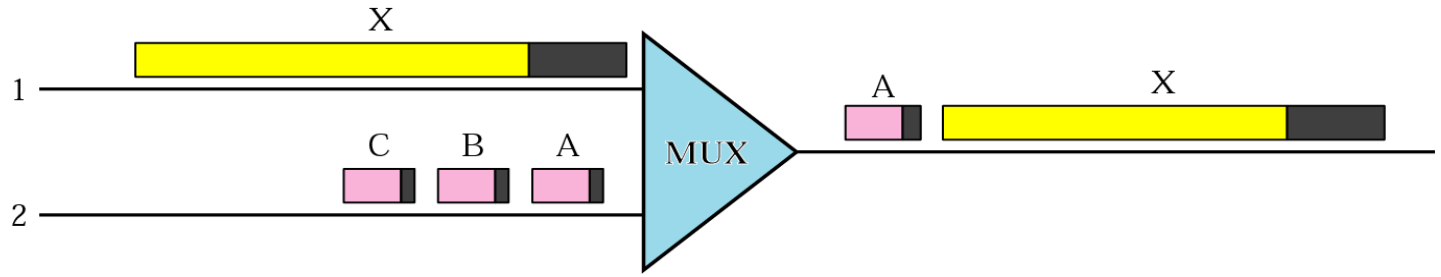


* Figure is courtesy of W. Stallings

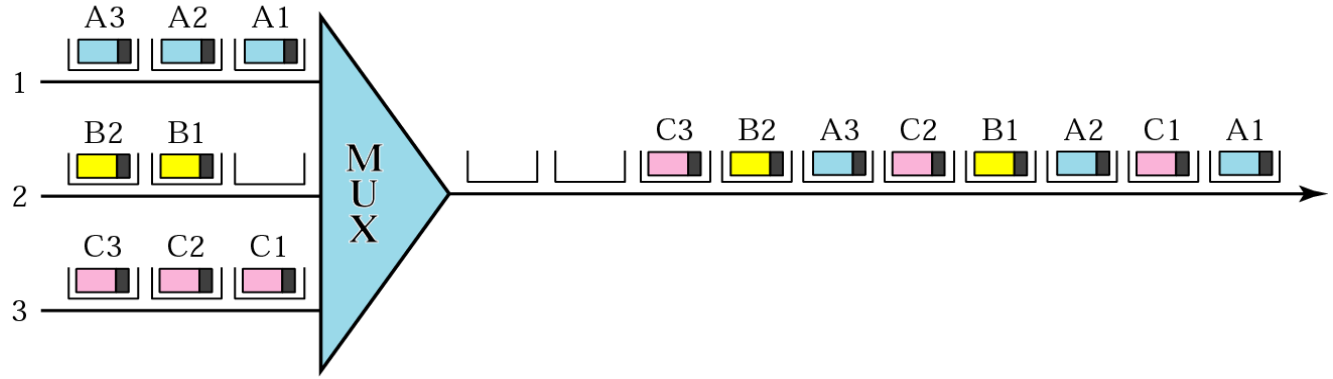
Asynchronous Transfer Mode (ATM)

- Example of virtual circuit switching
  - Cell-Switching
- Similarities between ATM and packet switching
  - Transfer of data in discrete chunks
  - Multiple logical connections over single physical interface
- In ATM flow on each logical connection is in fixed sized packets called cells
- Minimal error and flow control
  - Reduced overhead

Motivation for ATM



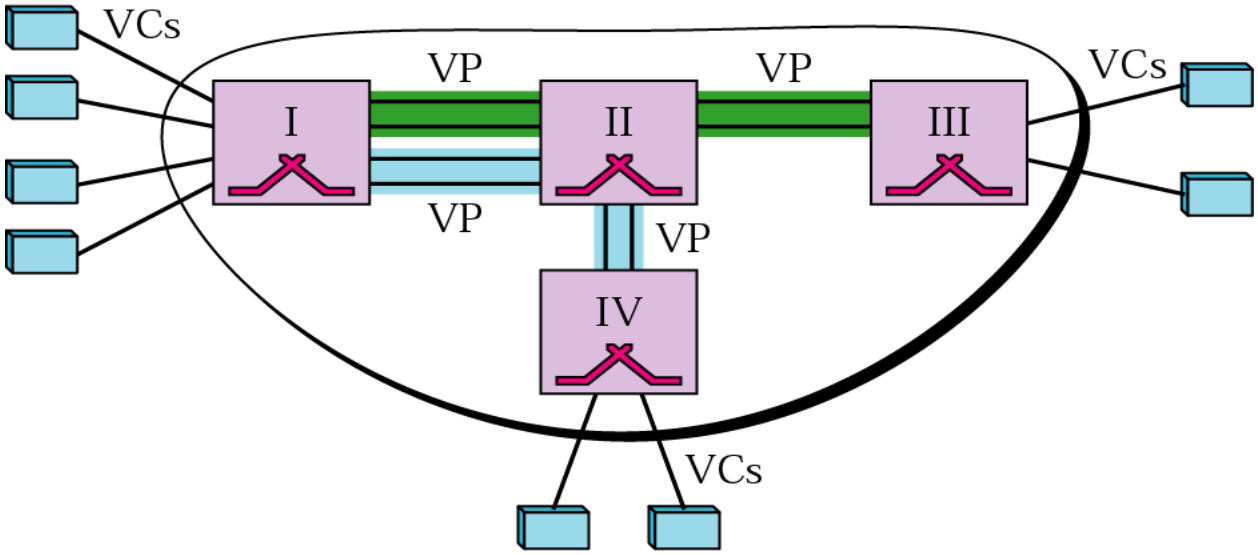
- Frames at a switch may be handled in any order and occupy switch for underspecified time



- Small, fixed-size frames allow simple, fast switches

* Figure is courtesy of B. Forouzan

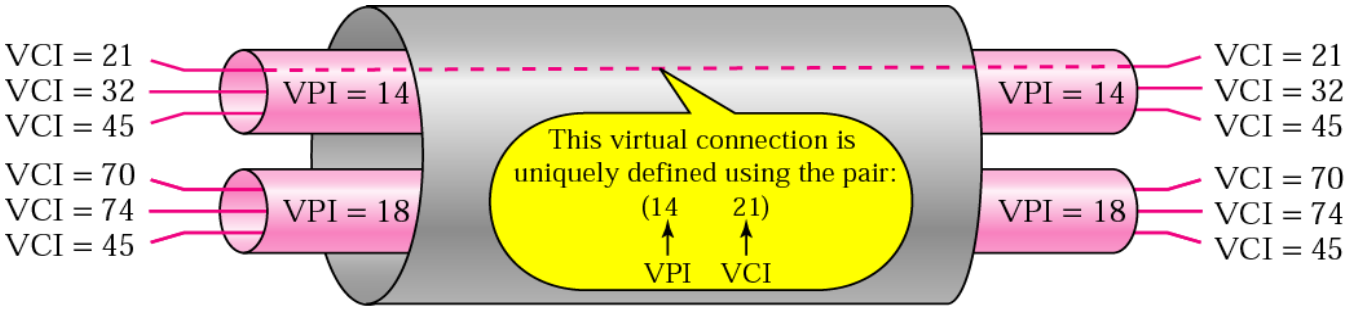
Virtual Circuits / Virtual Paths



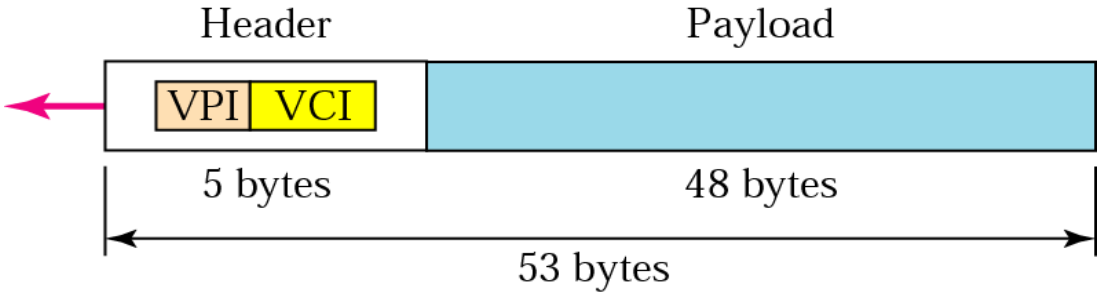
- Virtual circuits are collected into virtual paths

* Figure is courtesy of B. Forouzan

ATM Packet



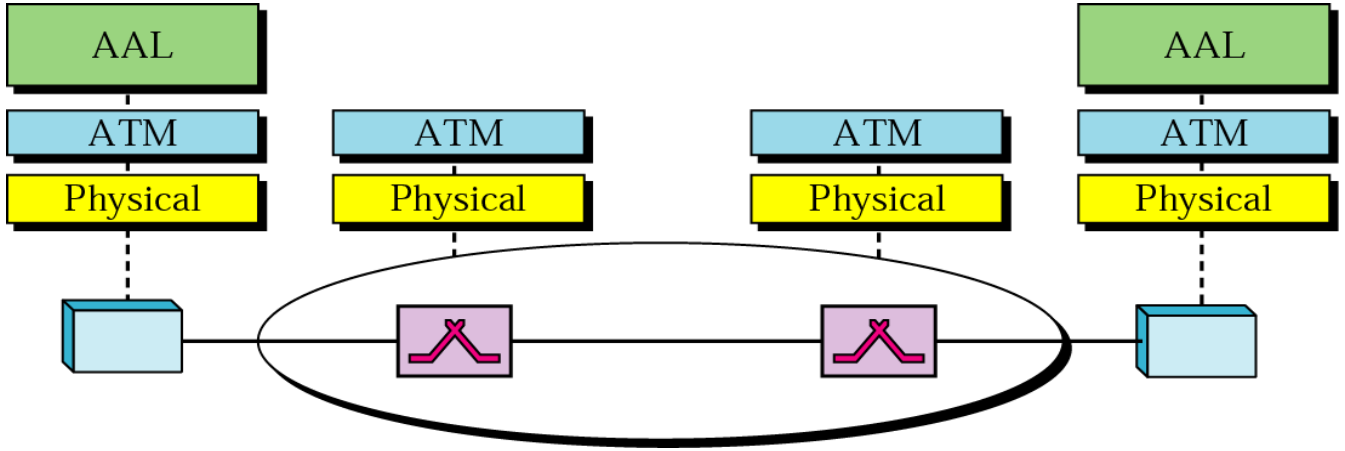
- Connection is specified by combination of Virtual Path ID and Virtual Circuit ID



- Every frame is exactly 53 bytes

* Figure is courtesy of B. Forouzan

Application Adaptation Layer (AAL)



- ATM defined a number of AALs for various purposes (each has its own header format):
  - AAL1: Constant bit rate e.g. multimedia
  - AAL2: Variable-data-rate
  - AAL3/4: Connection-oriented data services
    - Sequencing and Error Control
  - AAL5: Simple and efficient adaptation layer (SEAL)

* Figure is courtesy of B. Forouzan

ATM – It Didn't Happen

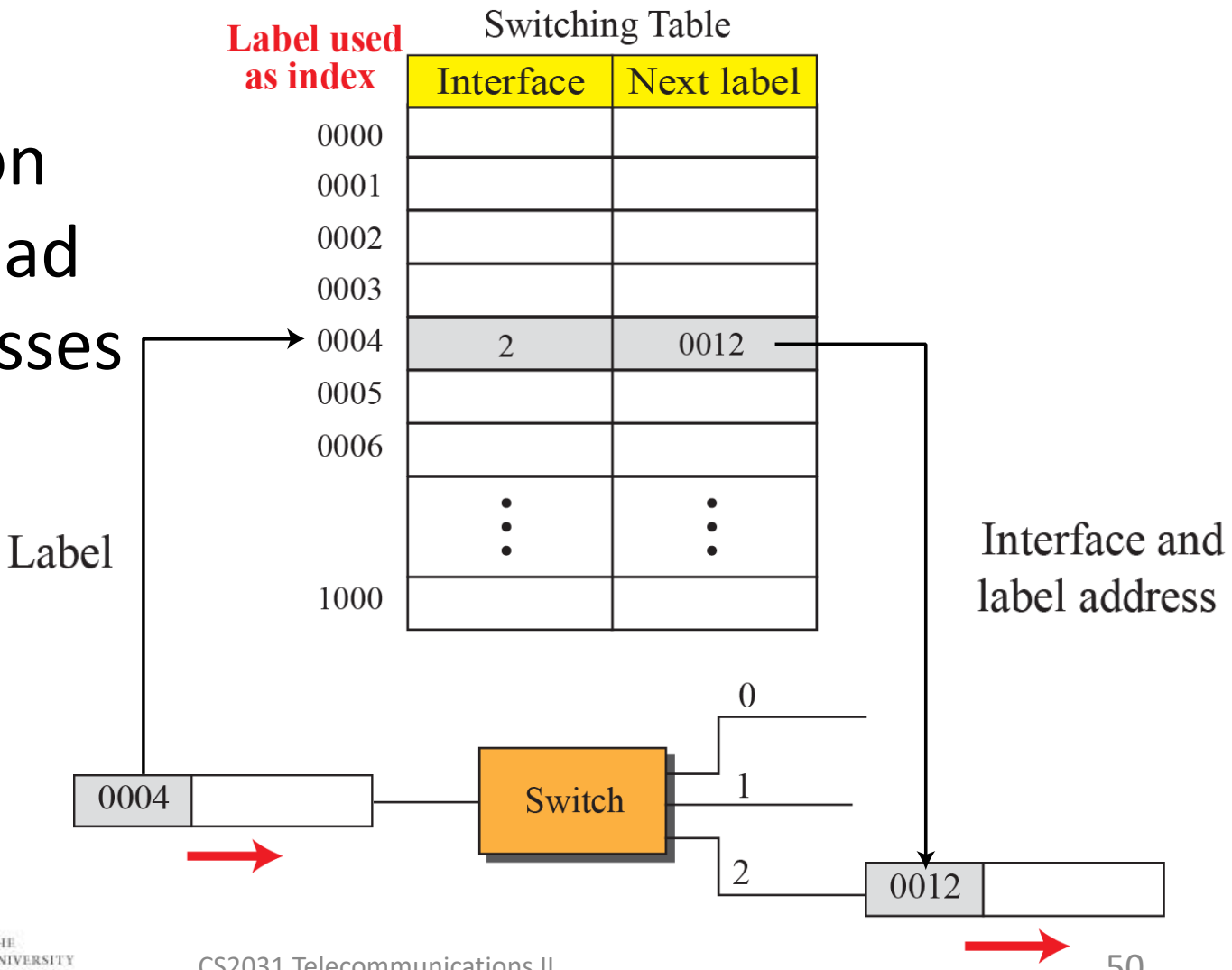
- From Tanenbaum:

  "ATM was going to solve all the world's networking and telecommunications problems by merging voice, data, cable television, telex, telegraph, carrier pigeons, …"

- It didn't happen:
  - Bad Timing
  - Technology
  - Implementation
  - Politics

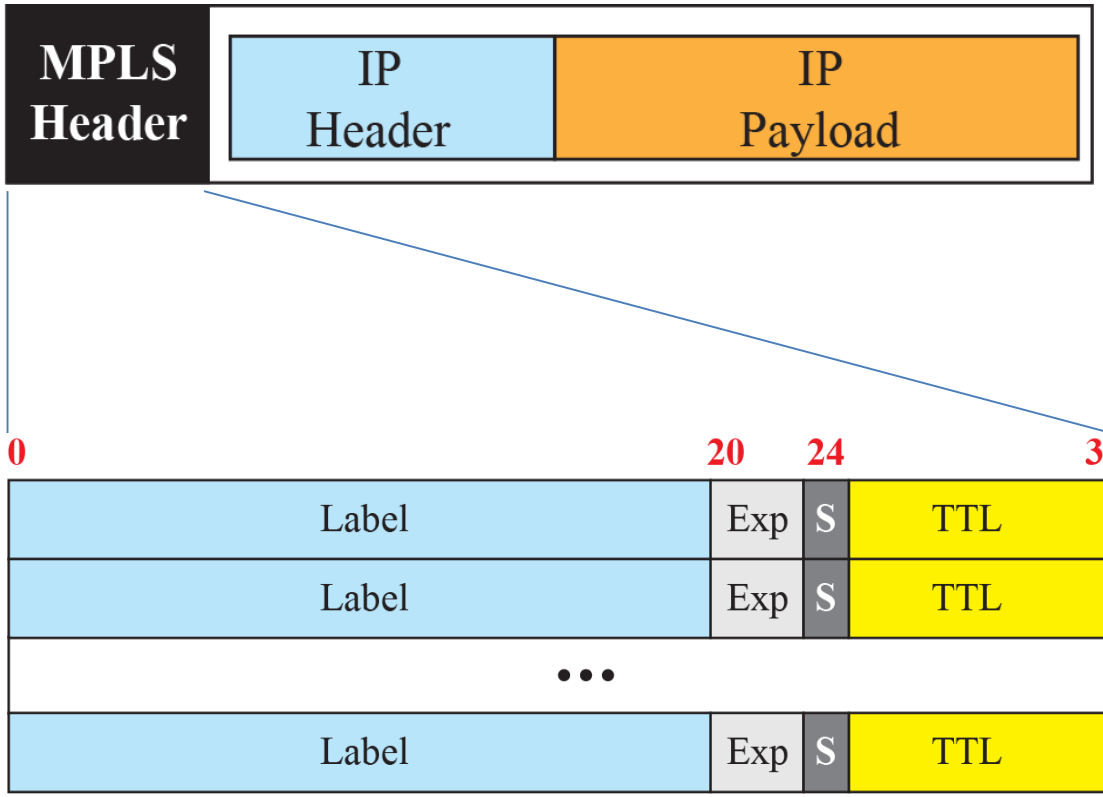
Multiprotocol Label Switching (MPLS)

- Enables switching on labels instead of IP addresses

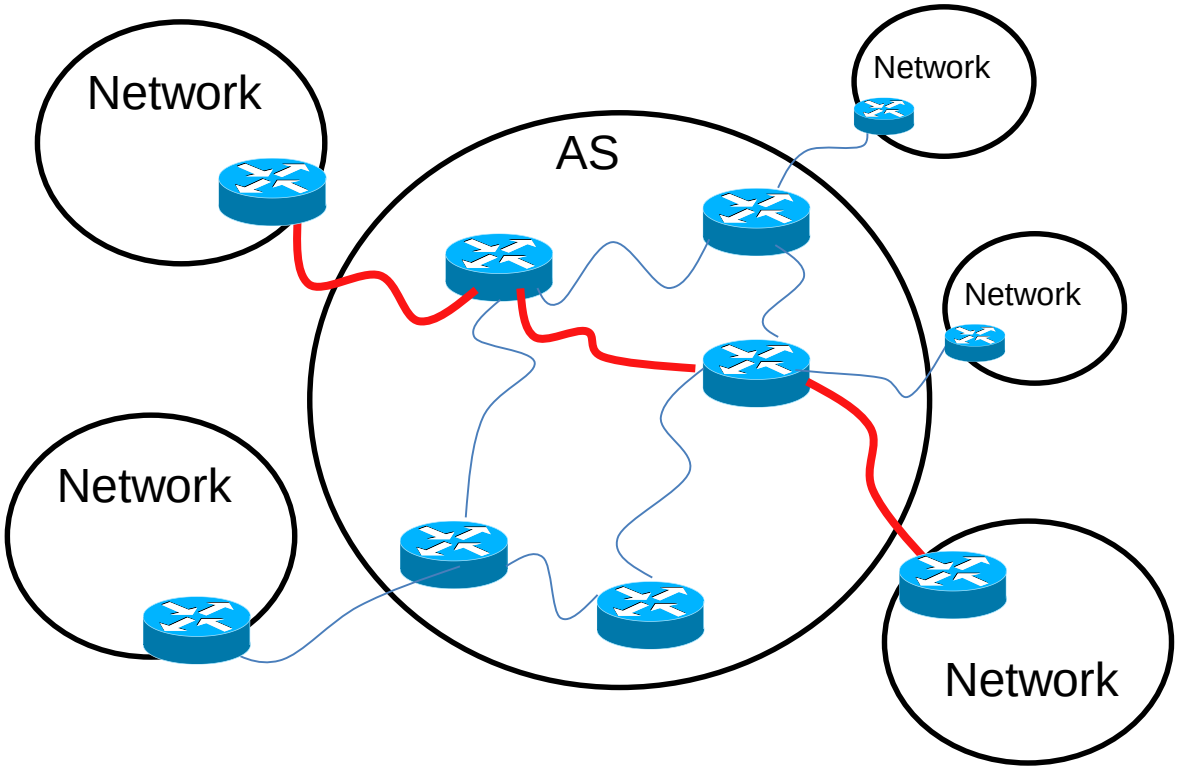


MPLS Header

- MPLS header as stack of labels

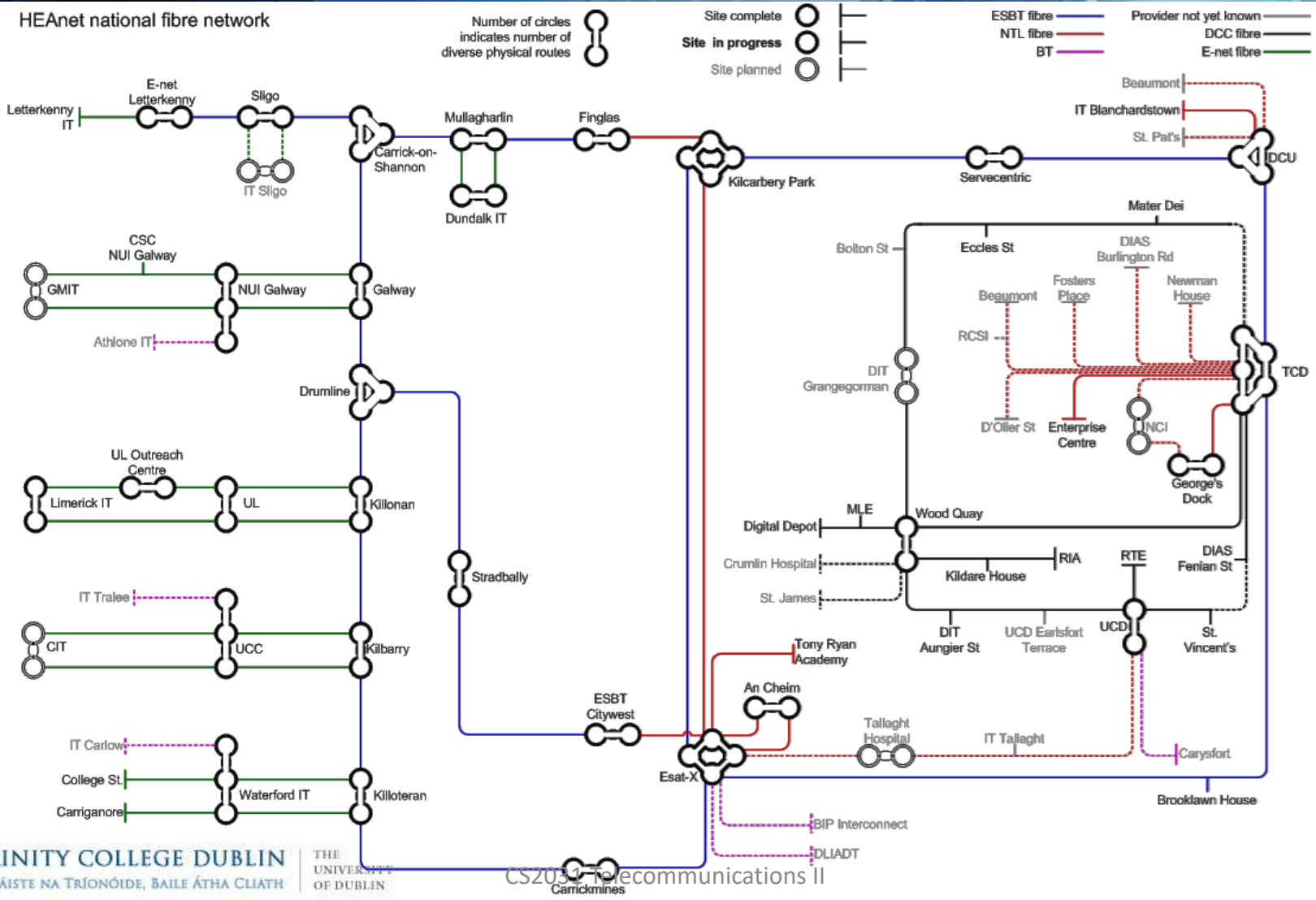


MPLS Use Case



- Creating a virtual network

HEAnet Fibre Network



Summary:Virtual Circuit Switching – ATM

- Virtual Circuit Switching
  - Preplanned route established before any frames sent
  - Call request and call accept frames establish connection (handshake)
  - Each frame contains a virtual circuit identifier instead of destination address
  - No routing decisions required for each frame
  - Clear request to drop circuit
  - Not a dedicated path
- Asynchronous Transfer Mode (ATM)
  - Example for virtual circuit switching
  - Cells consist of 5-byte header and 48-byte payload
  - Circuits identified by virtual circuit ID and virtual path ID
  - Application adaptation layer (AAL) for specific application areas


That's all folks
Telecomms II

CS2031
Telecommunications II

Assignment 2

Openflow – Quick Intro

OpenFlow: Enabling Innovation in Campus Networks

Nick McKeown
Stanford University

Tom Anderson
University of Washington

Hari Balakrishnan
MIT

Guru Parulkar
Stanford University

Larry Peterson
Princeton University

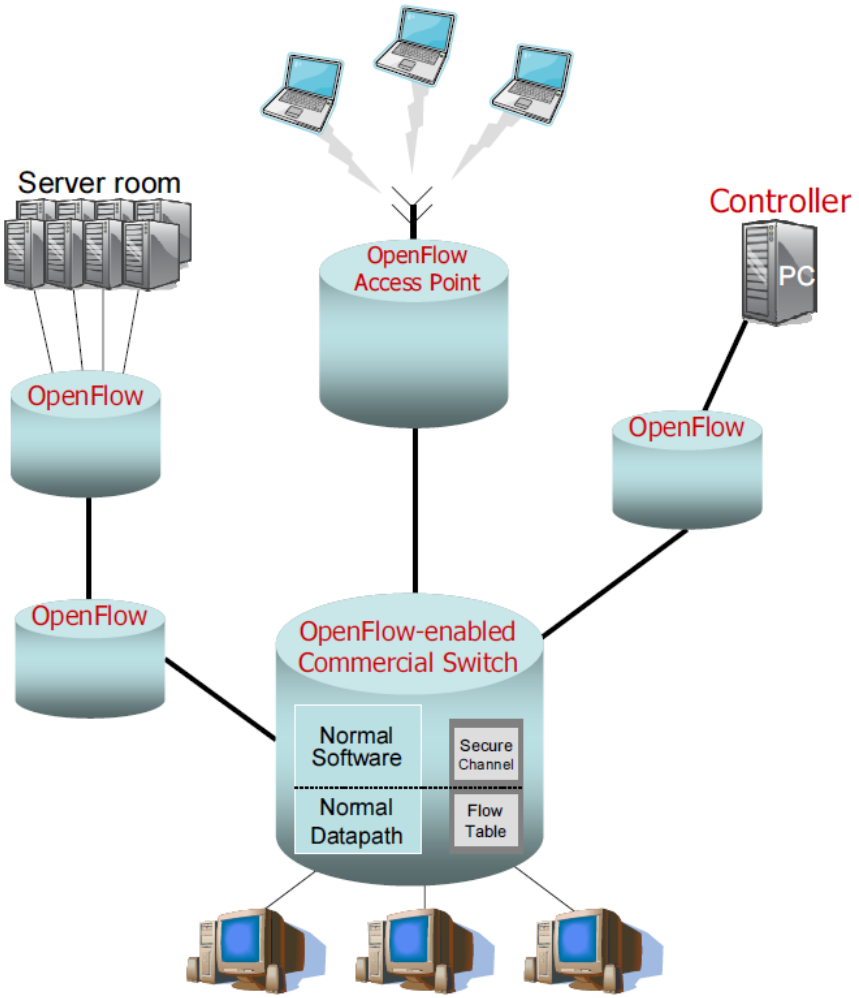
Jennifer Rexford
Princeton University

Scott Shenker
University of California, Berkeley

Jonathan Turner
Washington University in St. Louis

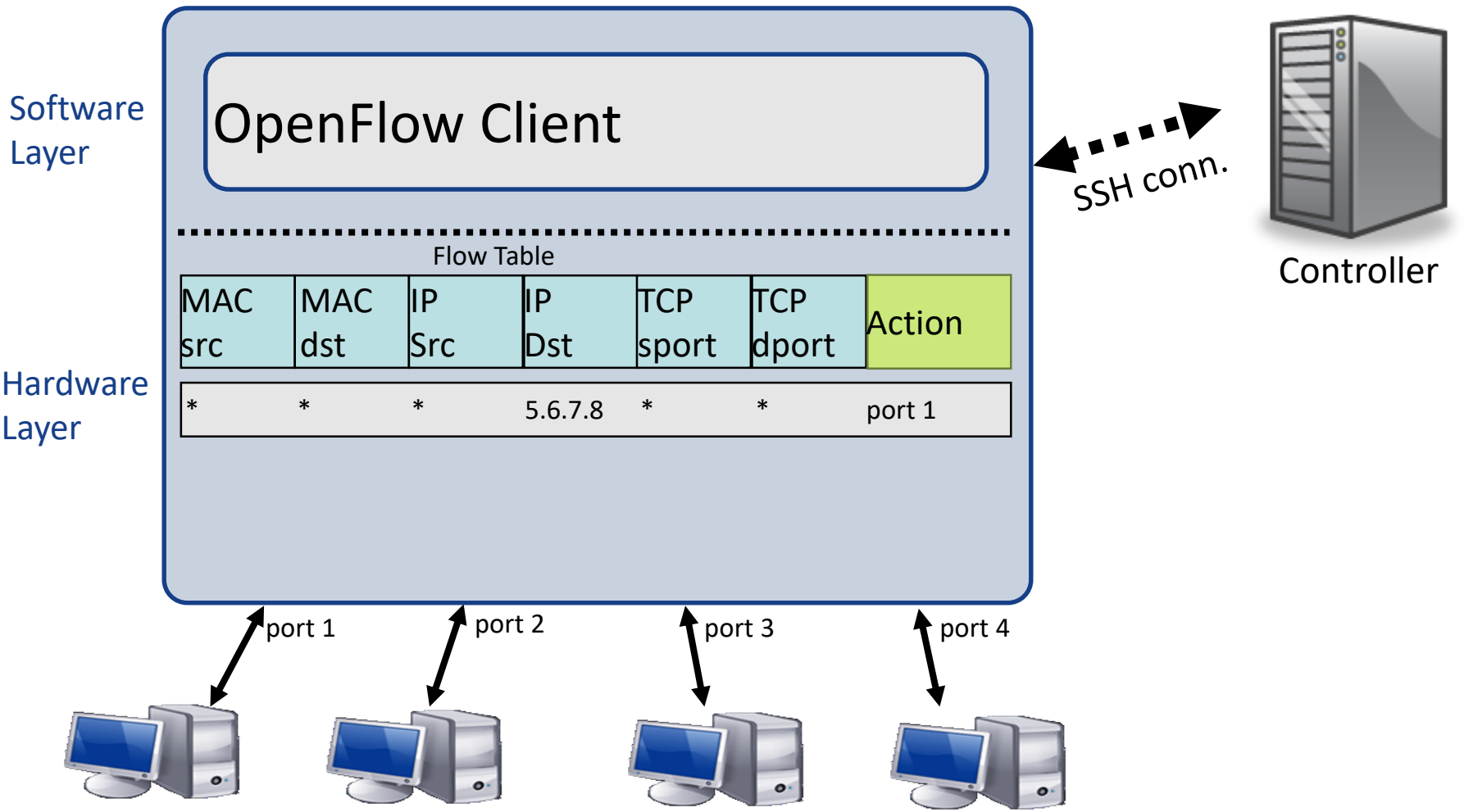
Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, and Jonathan Turner, OpenFlow: Enabling Innovation in Campus Networks. *SIGCOMM Computer Communications Review,* vol 38, issue 2, March 2008, 69-74.

From the Original Openflow Paper



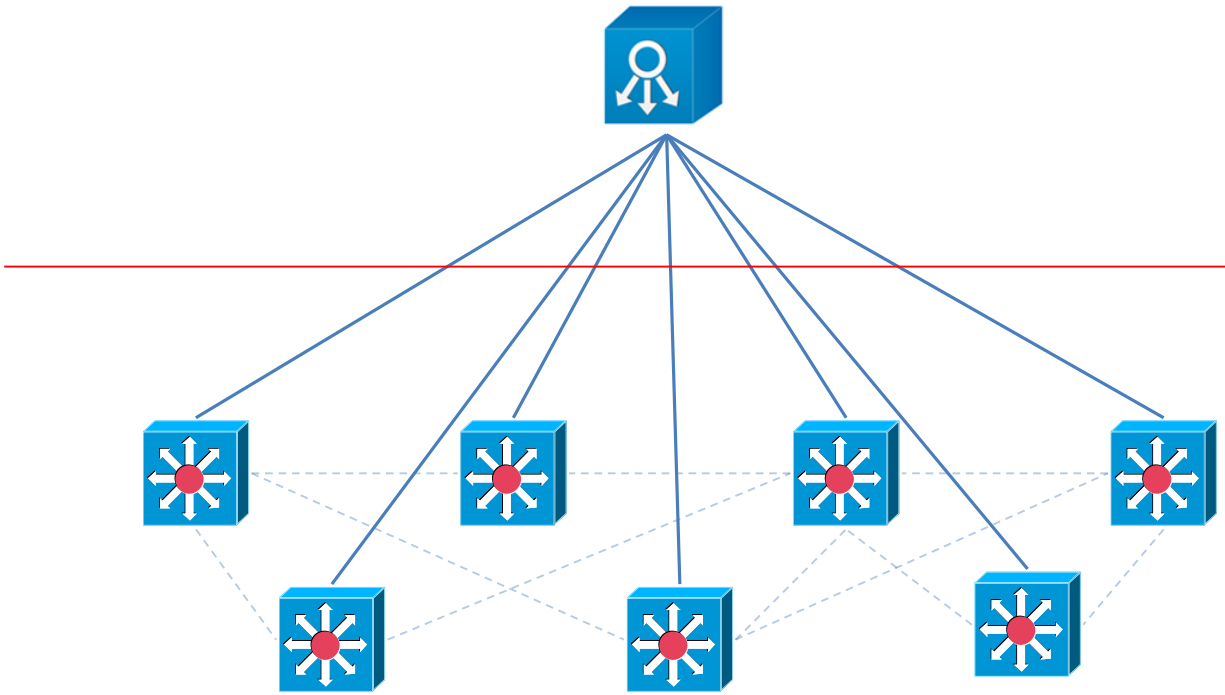
•Nick McKeown et al, SIGCOMM 2008

Openflow Switch



•Yiannis Yiakoumis, et al, HOT-Interconnects 2010

Control Plane vs Data Plane



Control Plane

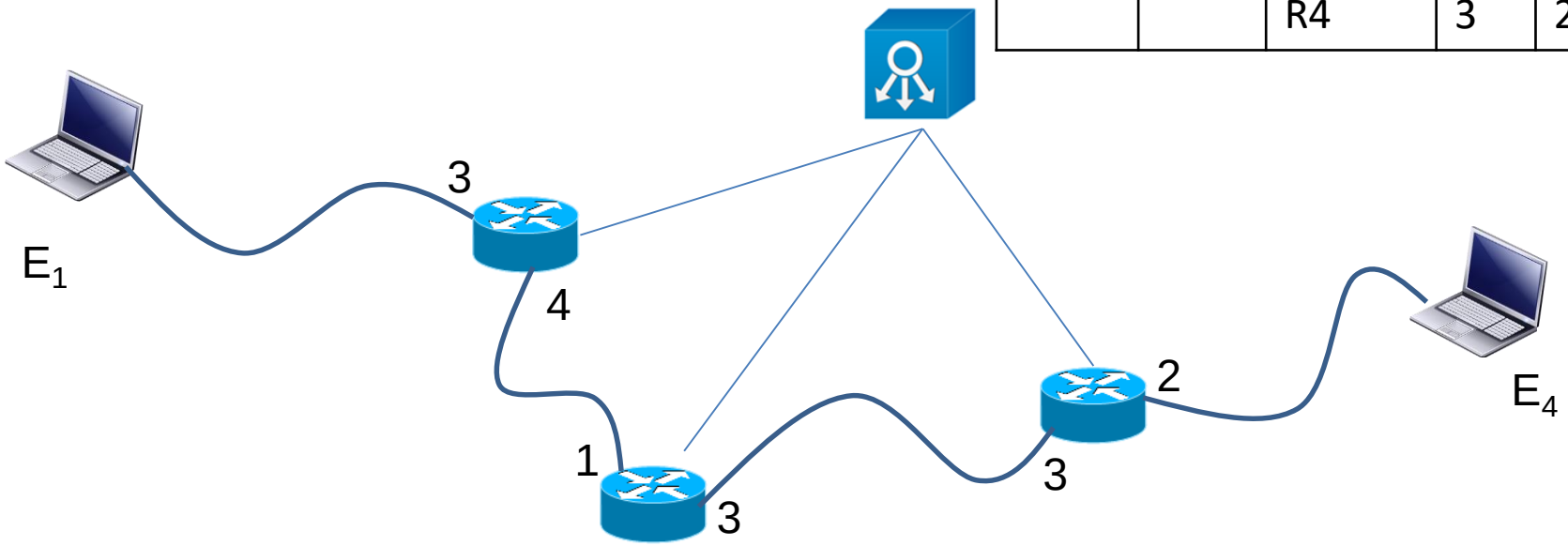
making routing decisions

Data Plane

forwarding data

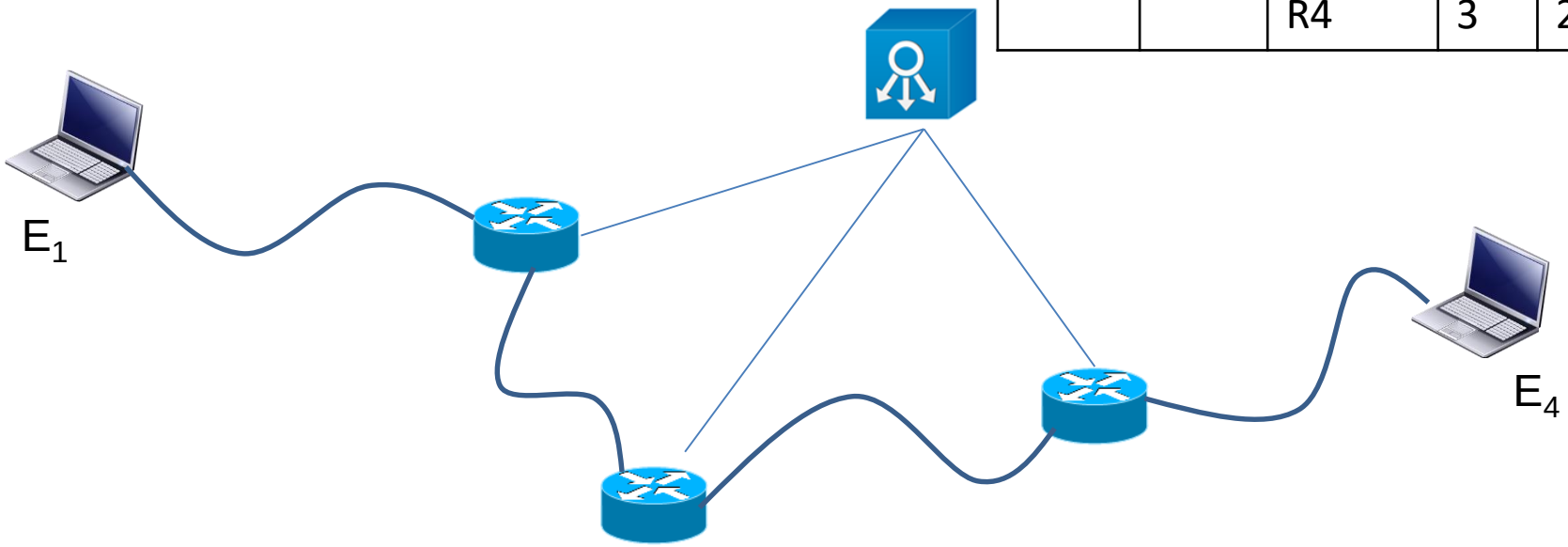
Assignment 2

Dest	Src	Router	In	Out
E4	E1	R1	3	4
		R2	1	3
		R4	3	2



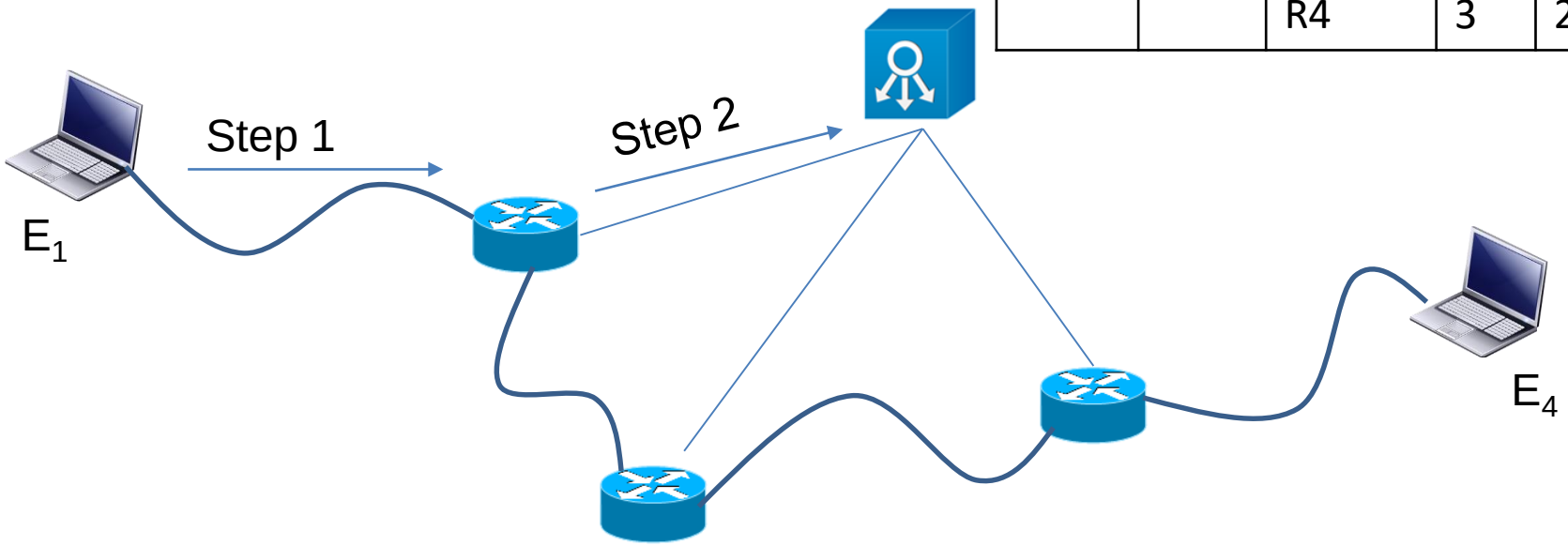
Assignment 2

Dest	Src	Router	In	Out
E4	E1	R1	3	4
		R2	1	3
		R4	3	2



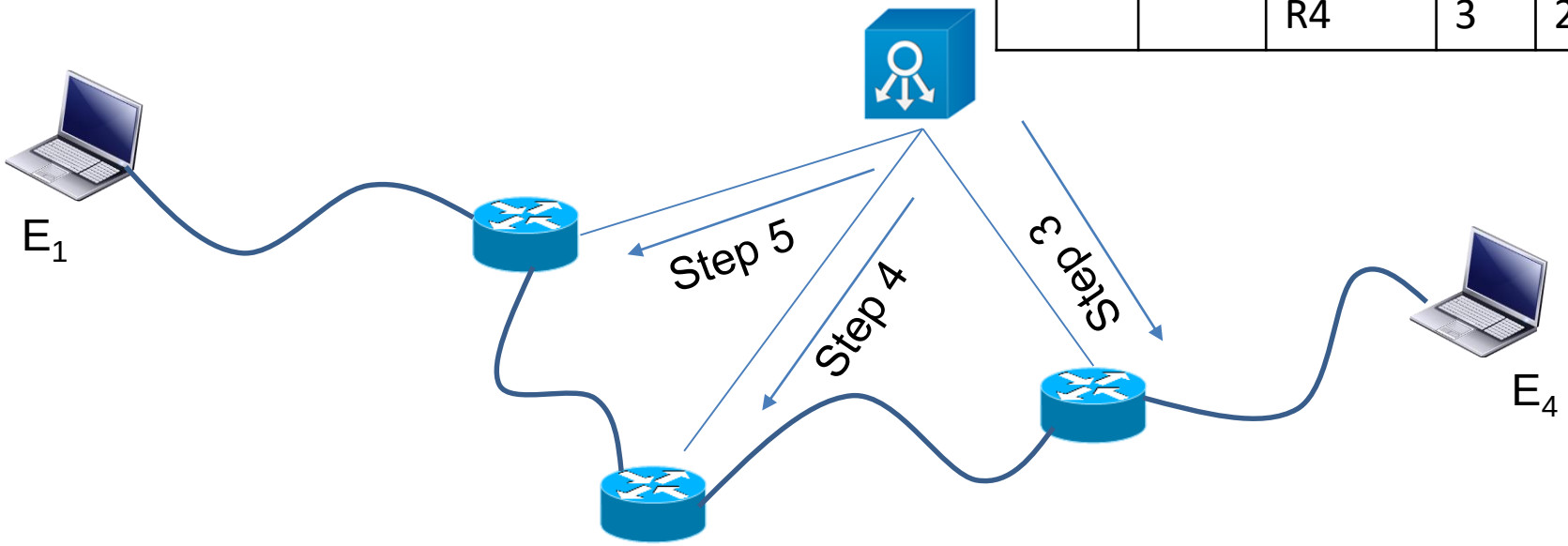
Assignment 2

Dest	Src	Router	In	Out
E4	E1	R1	3	4
		R2	1	3
		R4	3	2



Assignment 2

Dest	Src	Router	In	Out
E4	E1	R1	3	4
		R2	1	3
		R4	3	2



Assignment 2

Dest	Src	Router	In	Out
E4	E1	R1	3	4
		R2	1	3
		R4	3	2