Circuit Switching and Packet Switching

Dr Steve Gordon

ICT, SIIT

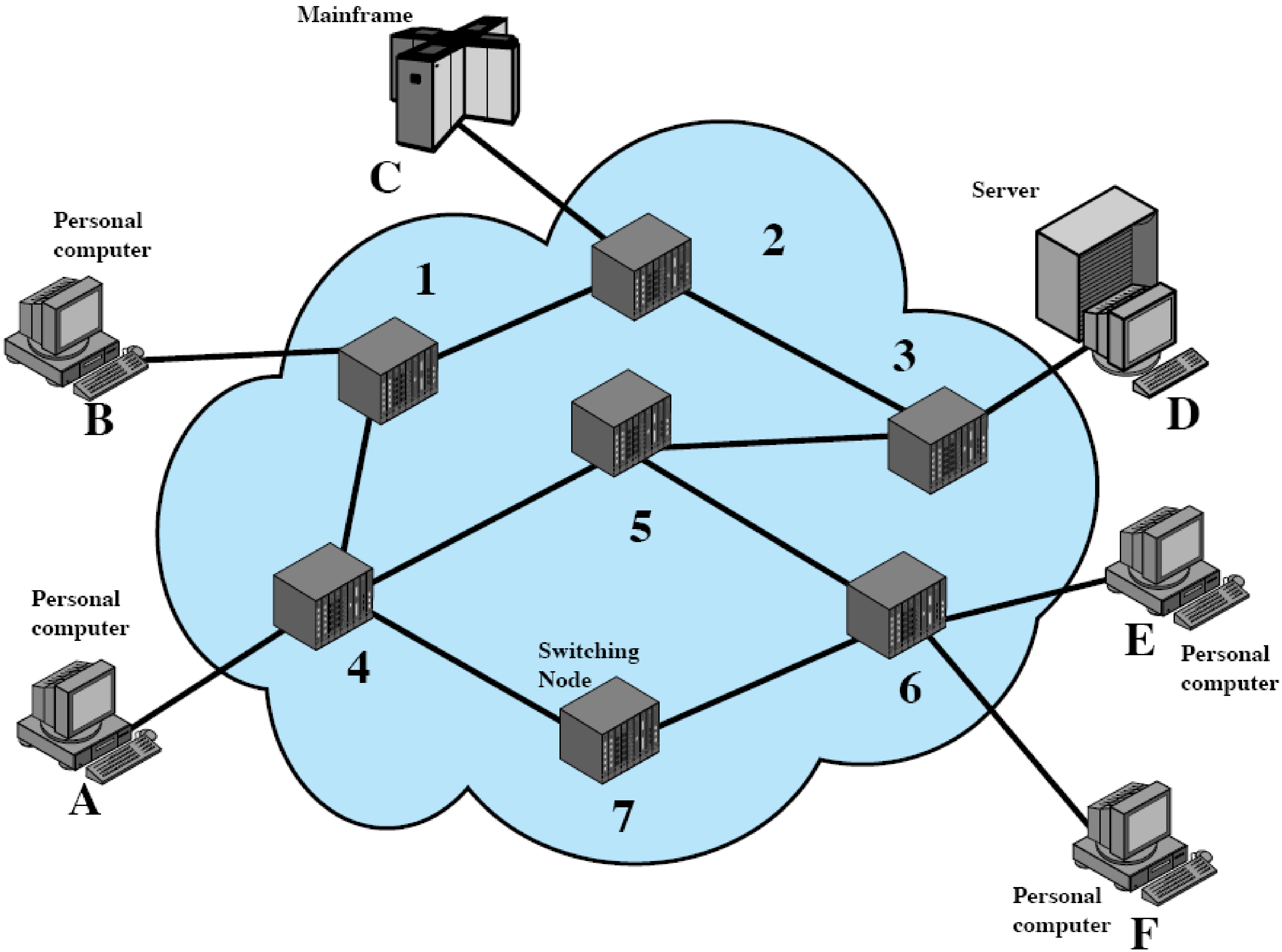
Contents

- Switched Networks
- Circuit Switching (CS)
- Packet Switching (PS)
  - Datagram Packet Switching (DPS)
  - Virtual Circuit Packet Switching (VCPS)
- Packet Switching versus Circuit Switching
  - Timing (Total Delay)
  - Queuing Delay in Packet Switching

Background

- We have focussed on technologies for transmission over communication links
  - Links can be wired or wireless; connect two (point-to-point) or more (broadcast) devices together
- What about data communications with someone not directly linked to you?
- Communication networks have multiple communication links
- In networks, switching is often used:
  - Data moves from one node (or switch) to another, until it reaches the destination
- Since the invention of telephone in late 1800's, circuit switching has been dominant technology for voice communications, and during computer era, also for data
- In 1970's, packet switching was developed: now the most effective means for long-distance data communications, e.g. the Internet

A Switched Network



Switched Networks

- Source transmits data through a network of switching nodes and finally to destination
  – Switching nodes do not care about content; main purpose is to forward the data until it reaches destination
- Devices attached to the network are called stations; often end-user devices
  – Computers, terminals, telephones, …
- Switching devices are called nodes or switches
  – Nodes connected to one another in some topology by transmission links
  – Stations attach to nodes
- A collection of nodes and connections is called a *communications network*
- Network is usually partially connected
  – Some redundant connections are desirable
  – Some nodes only connect to nodes (that is, not connect to stations); usually the links use multiplexing (TDM, FDM)
- Two different switching technologies
  – Circuit switching
  – Packet switching

Circuit Switching

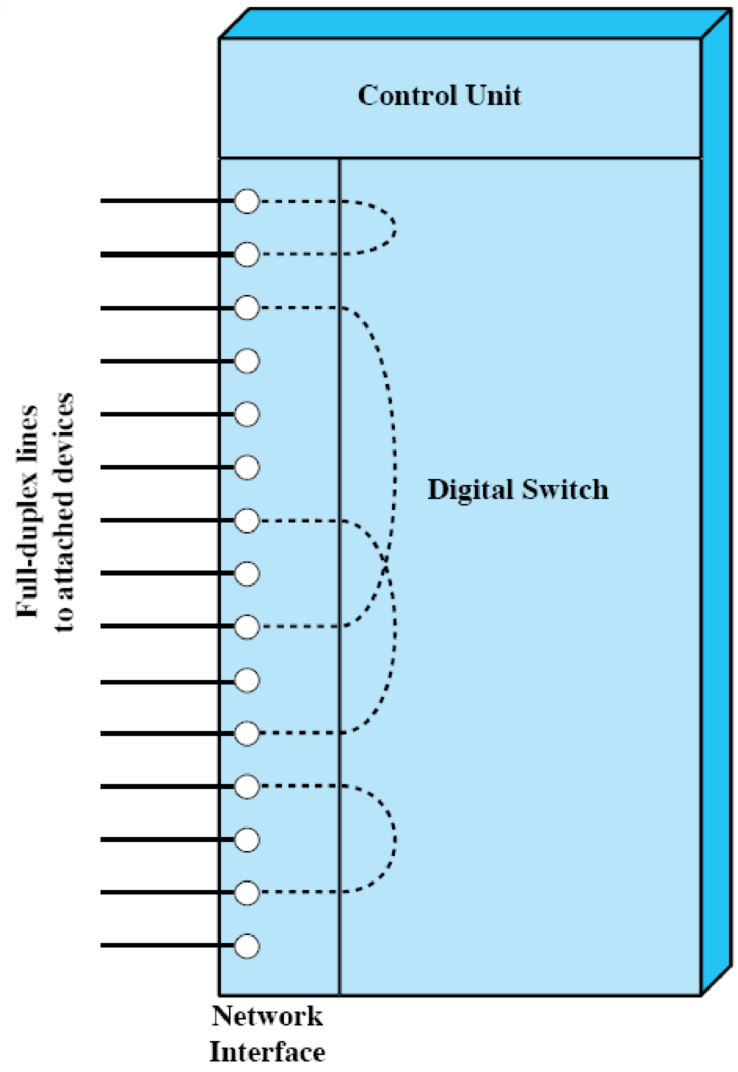

By pressing keys. . .

Circuit Switching

- Circuit switching uses a dedicated, physical path between two stations
  - Used in combination with multiplexing (FDM, TDM)
  - Result: the two stations have dedicated link between them
  - Circuit switching is performed at physical layer
    - Allows (analog or digital) signals to pass between stations
- Three phases in circuit switching:
  - Circuit establishment: before any data is sent, must create a circuit (including allocating resources for the circuit)
  - Data transfer: once the circuit is created, can transfer (analog or digital) data across circuit
  - Circuit disconnect: after certain time, circuit is terminated, and resources are freed
- Applications:
  - Developed to handle voice traffic, but also used for data
  - Public telephone network; Private telephone networks; private data networks

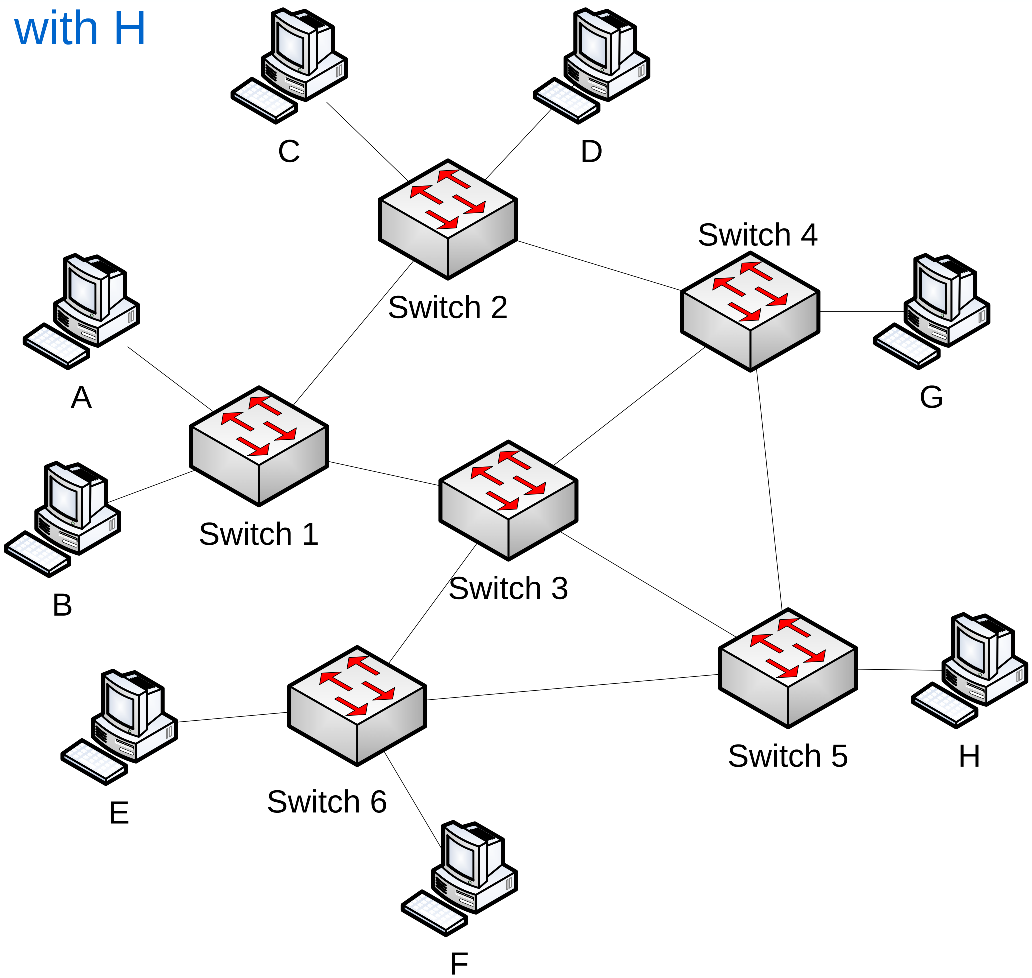
A Circuit Switching Node

- Aims to provide transparent signal path between any pair of attached devices
  - Usually full-duplex
- Three components in a node (switch)
  - Network Interface allows different network devices to connect (e.g. TDM E1 or E3 signals)
  - Digital Switch determines the connectivity of lines
    - Space division switching
    - Time division switching
    - And combinations …
  - Control Unit manages the connection
    - Involved in circuit (path) setup, teardown
    - Tells the Digital Switch which lines to connect together



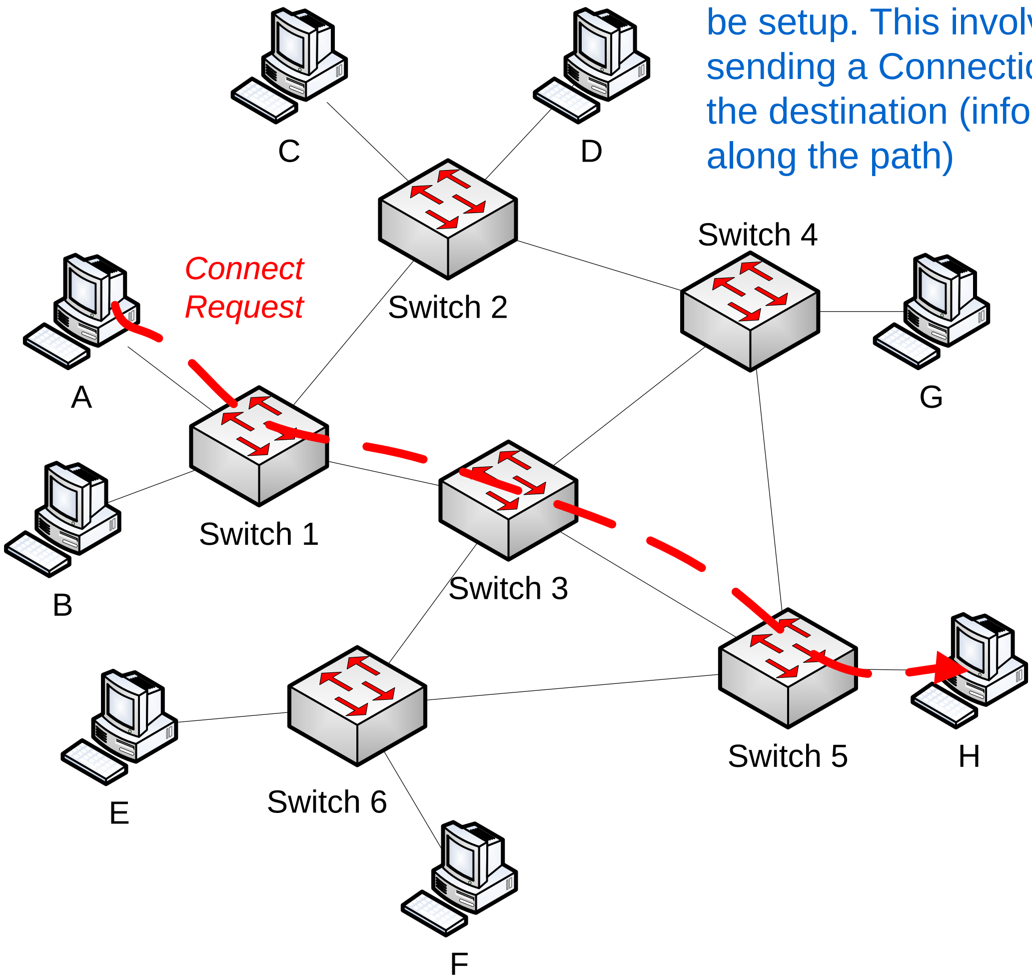
Circuit Switching

A communicating with H



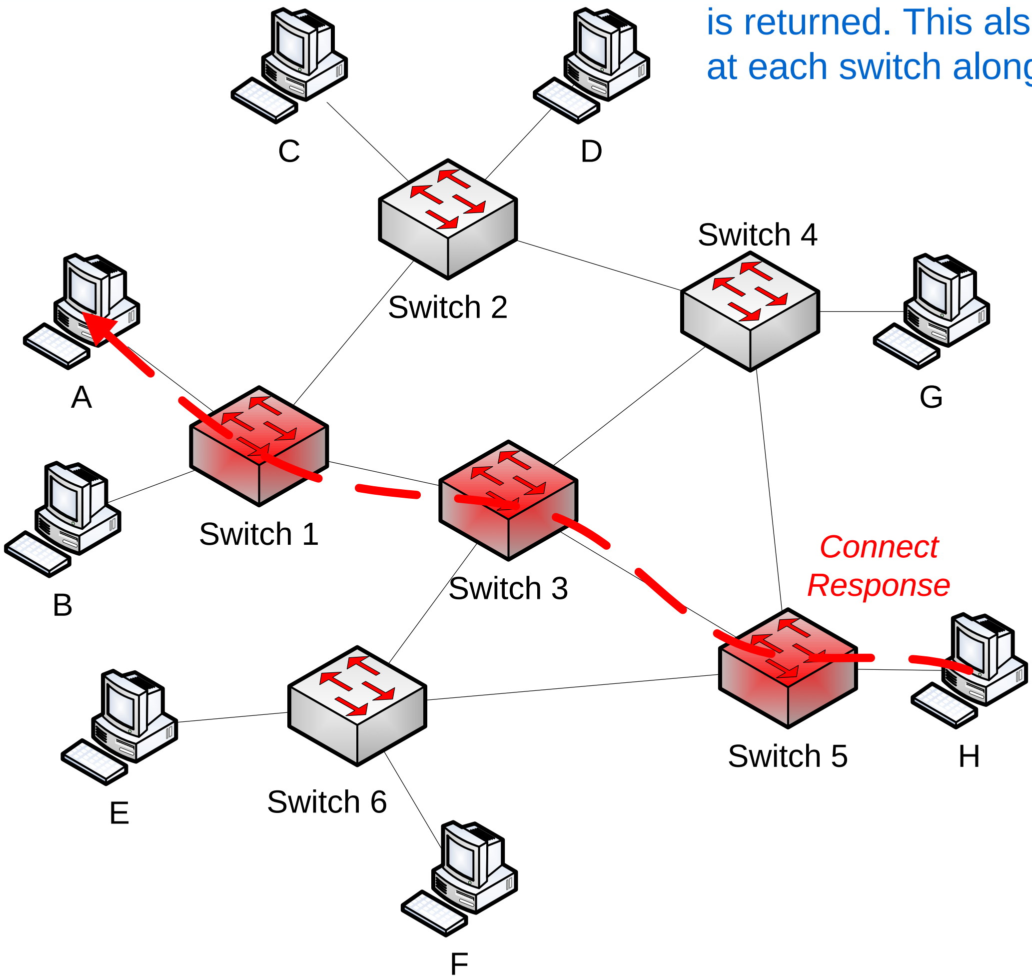
Circuit Switching

Before data is sent, a circuit (path) must be setup. This involves the source sending a Connection Request signal to the destination (informing the switches along the path)



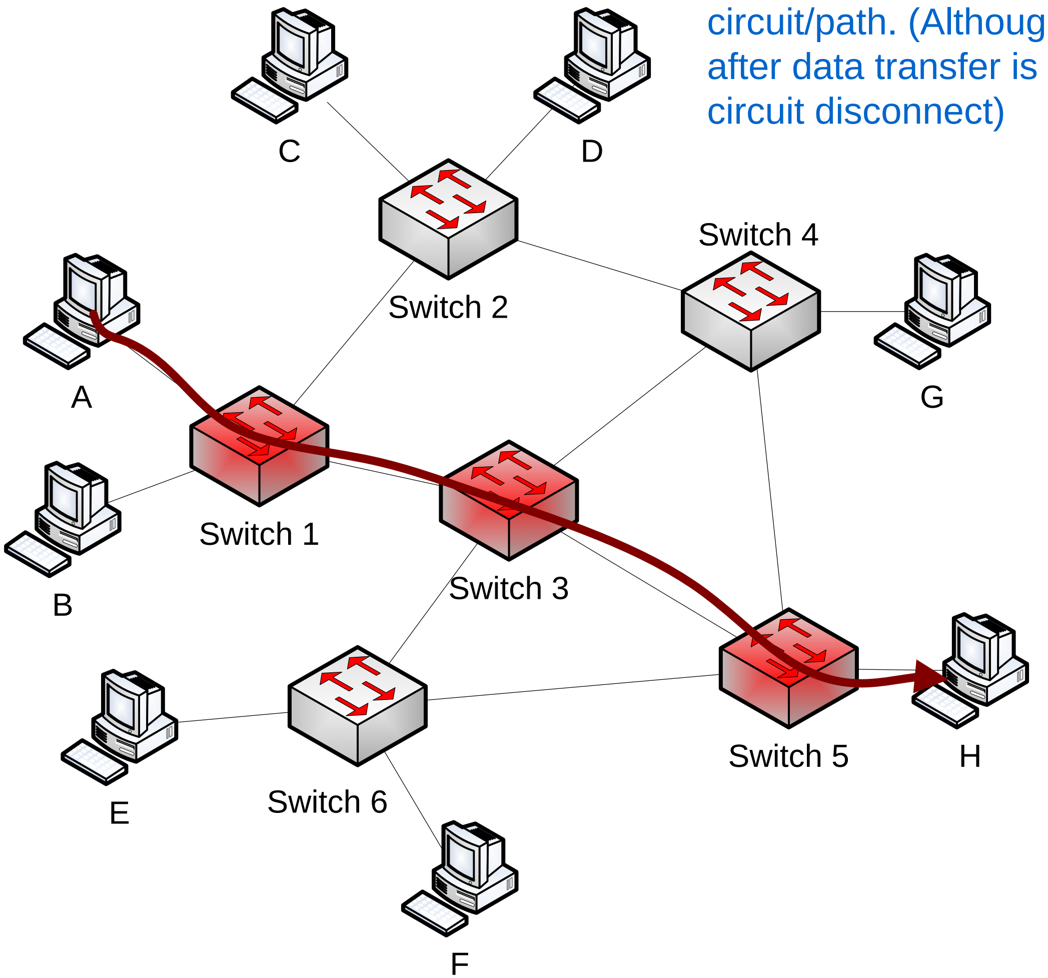
Circuit Switching

If connection is accepted, a response is returned. This also reserves the resources at each switch along the path.



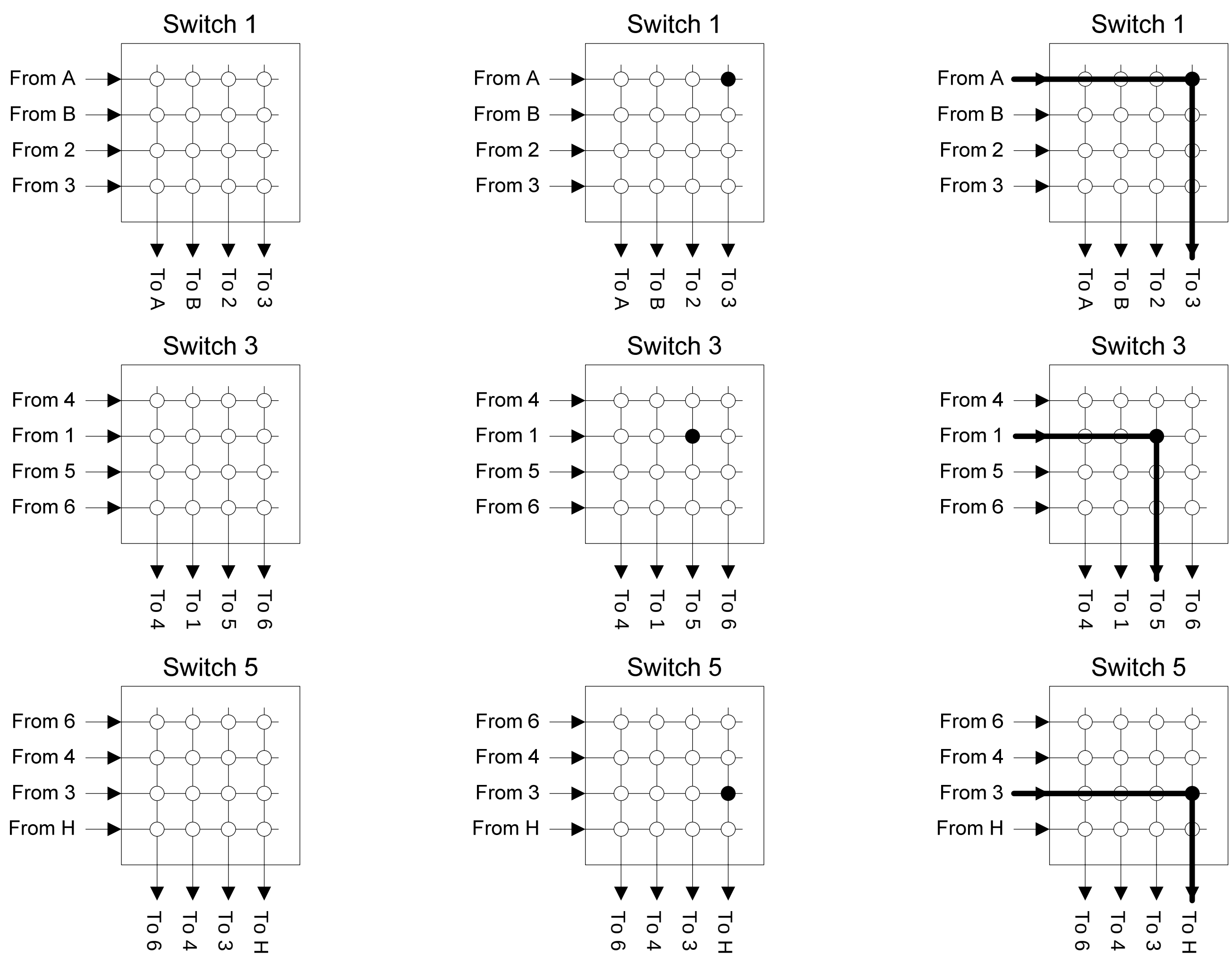
Circuit Switching

Now data can be transferred along the circuit/path. (Although not shown on slides, after data transfer is complete, there is a circuit disconnect)

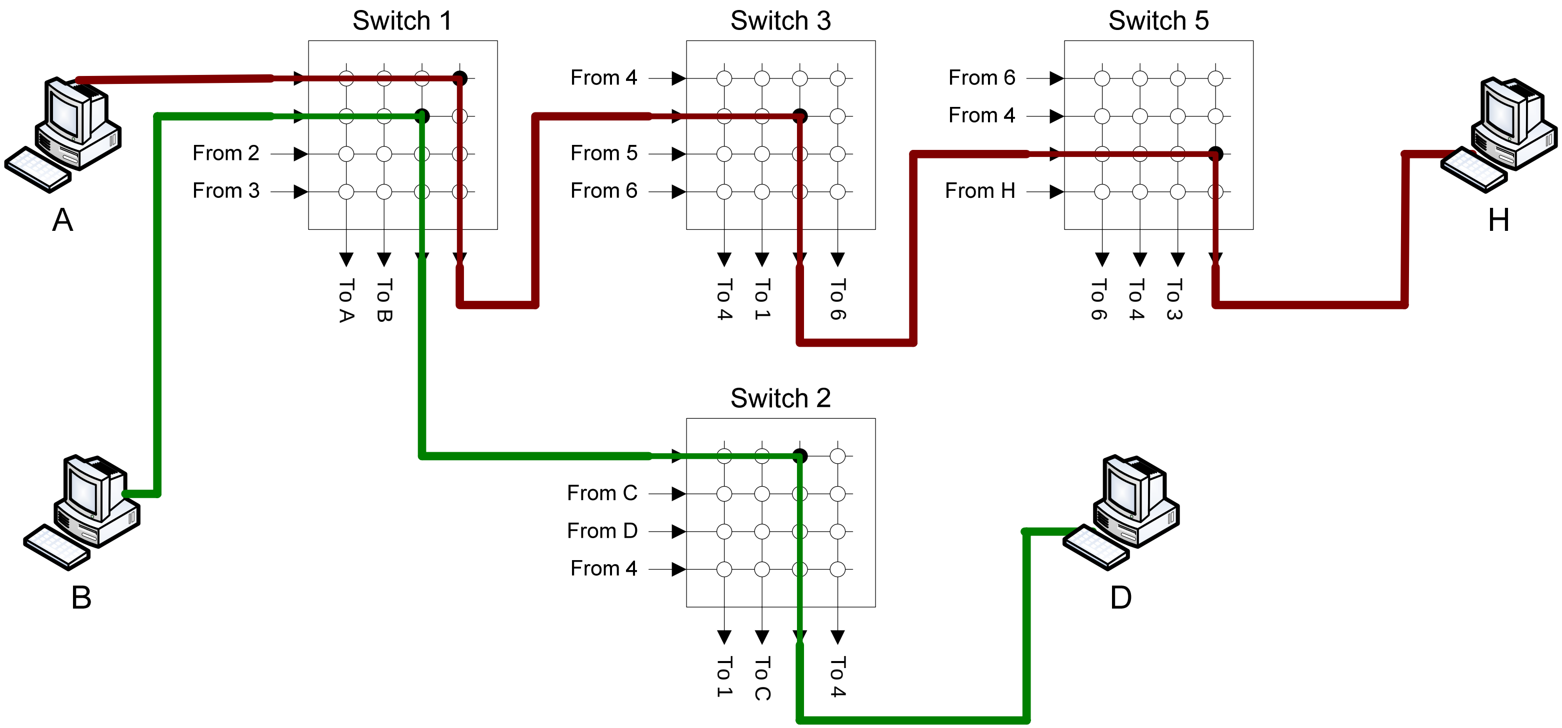


Space Division Switching

Before path setup *After path setup* *Data transfer*



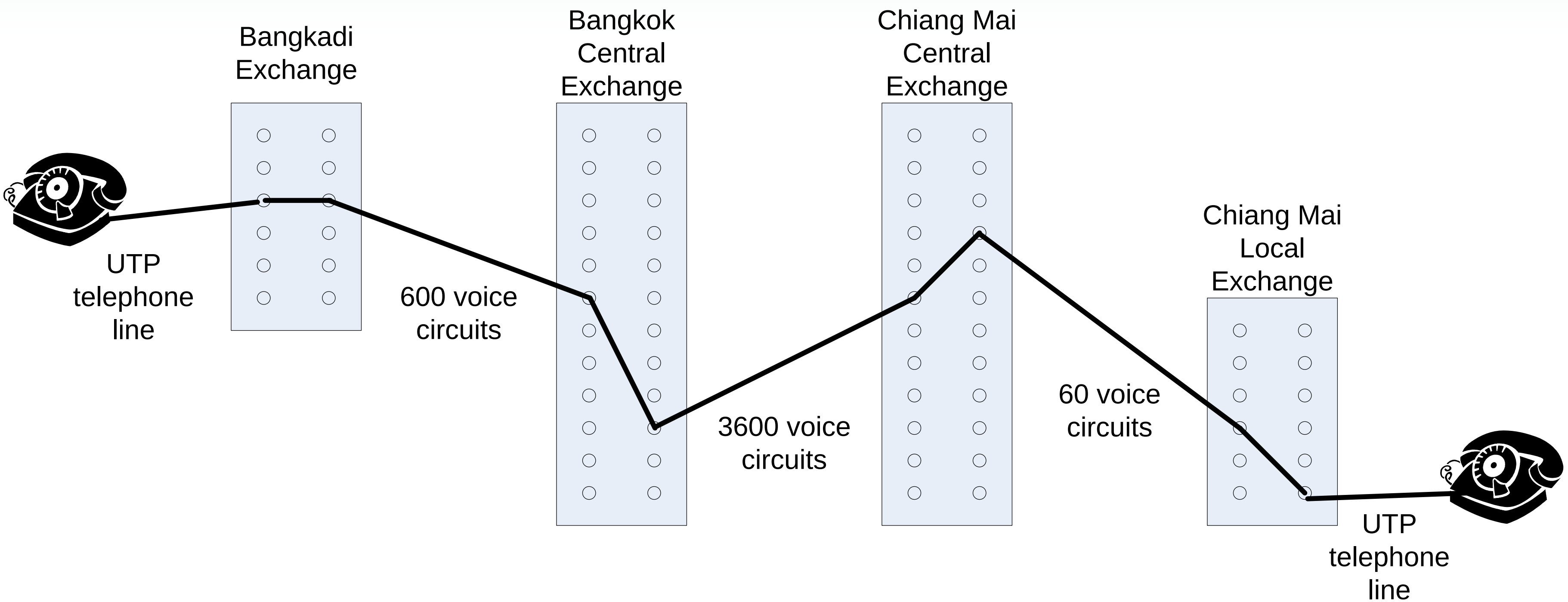
An Additional Circuit: B to D



Blocking and Switch Design

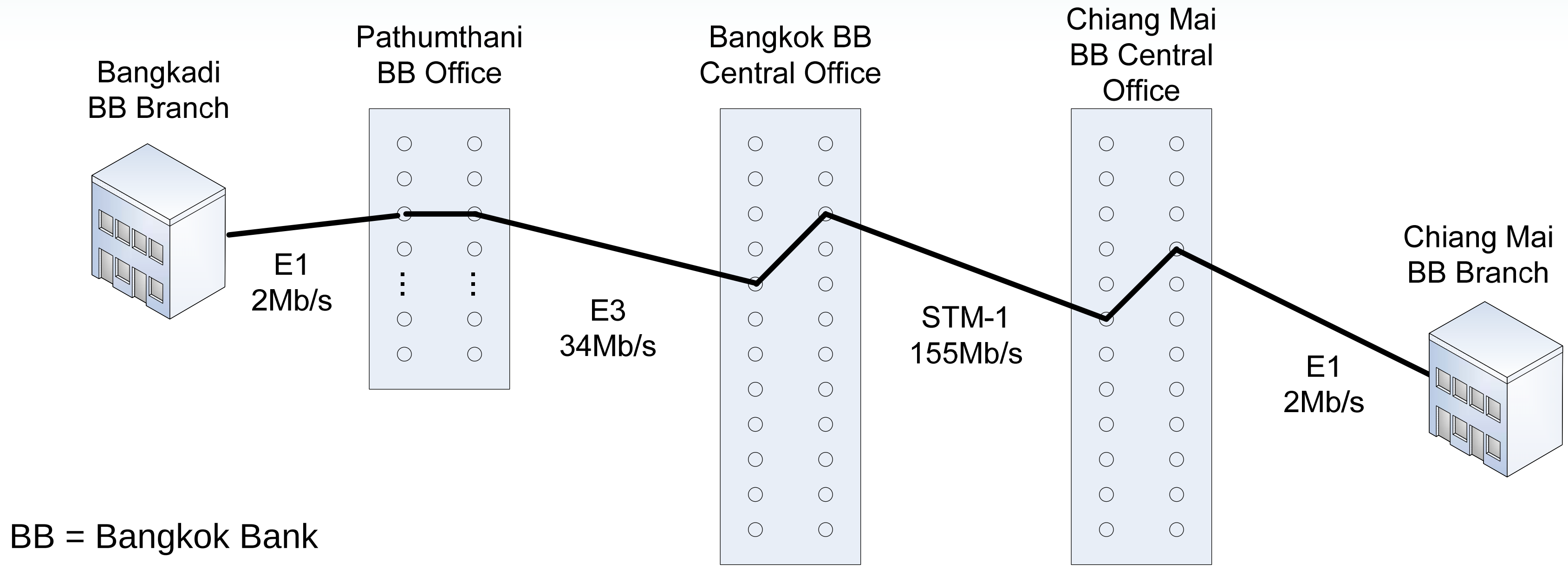
- Basic design aim of switches: reduce the number of cross-points, as this reduces the switch complexity/cost
  – With N input lines, allowed to connect to M output lines, NxM cross points needed
- Cross-points can be reduced using multiple stages
  – Allow N input lines to connect to M output lines, but not all at the same time
  – Leads to blocking (if cross-points are used
- Blocking
  – Call arrives at input line x to be connected to output line y, but the cross-points are in use
  – The call is *blocked*
- Reduce the number of cross-points can increase the probability of blocking
  – Telecommunication companies (and government regulators) define acceptable probabilities of block (e.g. 0.001% of calls are blocked)

Public Circuit Switched Telephone Network



Here we see an example combining circuit switching with FDM. There is a dedicated physical path between the telephones, even though on some links (e.g. Bangkok to Chiang Mai) the link is shared with other circuits. When the user in Bangkadi dials the number of the user in Chiang Mai and call is accepted, a path is setup across the network. Once setup, the two users can talk (transfer data). When a user ends the call, the path is disconnected.

Private Circuit Switched Data Network



Another example of circuit switching, this time using TDM. It illustrates a private data network of Bangkok Bank (BB), which connects branches together across Thailand. Assume Bangkadi BB branch has to transfer data (e.g. account and transaction information) to a BB branch in Chiang Mai. To do so, a path is setup across the network, and then the data is transferred.

Circuit Switching Phases

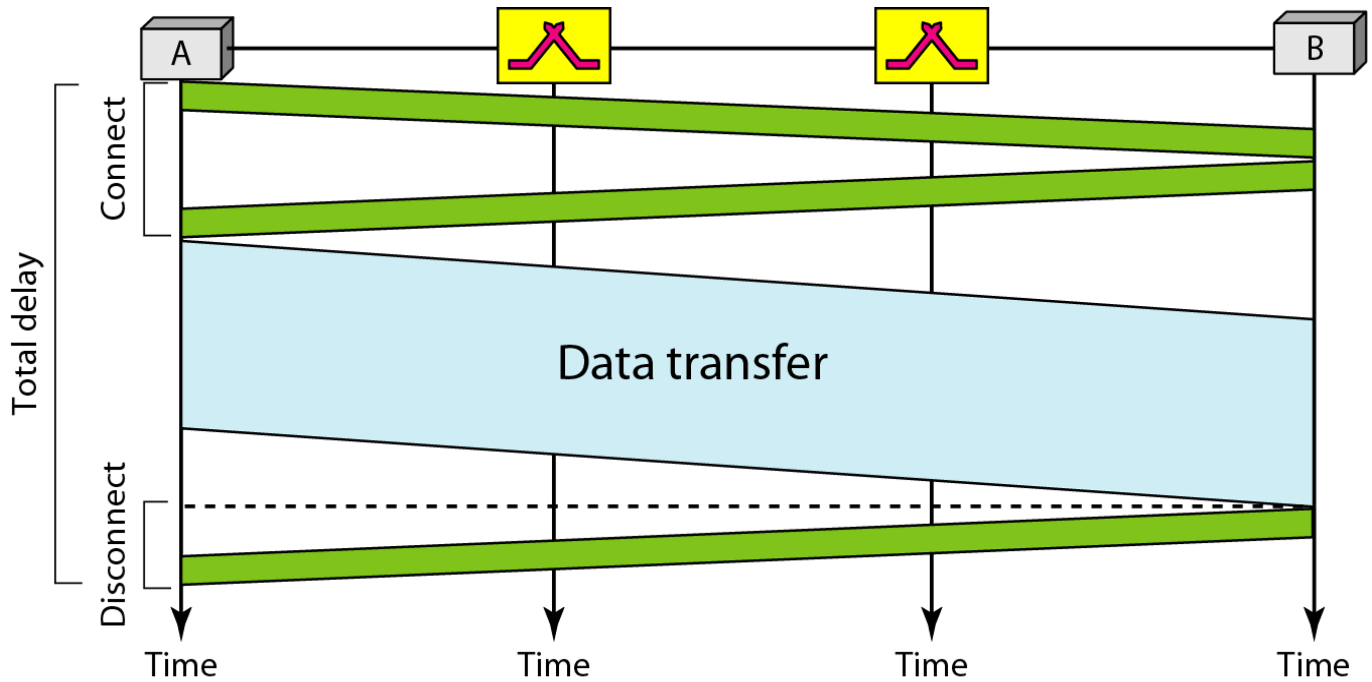
- Before sending data, the circuit must be established (setup, connected)
  - Involves selecting a path and reserving resources that will be used during data transfer. Resources include:
    - Bandwidth (FDM) or Time Slots (TDM); buffers in switches; switch input/output lines; switching processing time
    - These remain reserved (allocated for that circuit) until teardown, EVEN if not used
  - Needs an end-to-end addressing scheme (so user A can connect to user B)
    - Telephone numbers
- Data transfer is allowed once a circuit is setup
  - Data transfer is continuous through the circuit. Since there is a physical path from A to B, the analog and digital signal can travel from A to B without delay at switches
  - Data is NOT sent as packets – no headers are needed to identify the destination
  - Note: even if no data is being transferred, the resources for the circuit are still reserved; they cannot be used by anyone else (other circuits)
- Once one of the users needs to end the communication, the circuit is disconnected (teardown)
  - All resources are released

Characteristics of Circuit Switching

- Efficiency
  - Since the resources are reserved for the entire duration of connection, circuit switching will be inefficient if data is not transferred during connection (hence good for voice calls, not good for Internet traffic)
- Quality
  - The quality of service is guaranteed during the connection
  - If the connection is 2Mb/s, then the two users full access to the 2Mb/s circuit for the duration of the circuit; No-one will take away resources during your connection
- Link Speeds
  - End devices (stations) must be the same “speed”
    - E.g. Telephone lines both support 4kHz bandwidth
    - E.g. Data stations must both be 2Mb/s
  - This limits connectivity between devices of different technologies

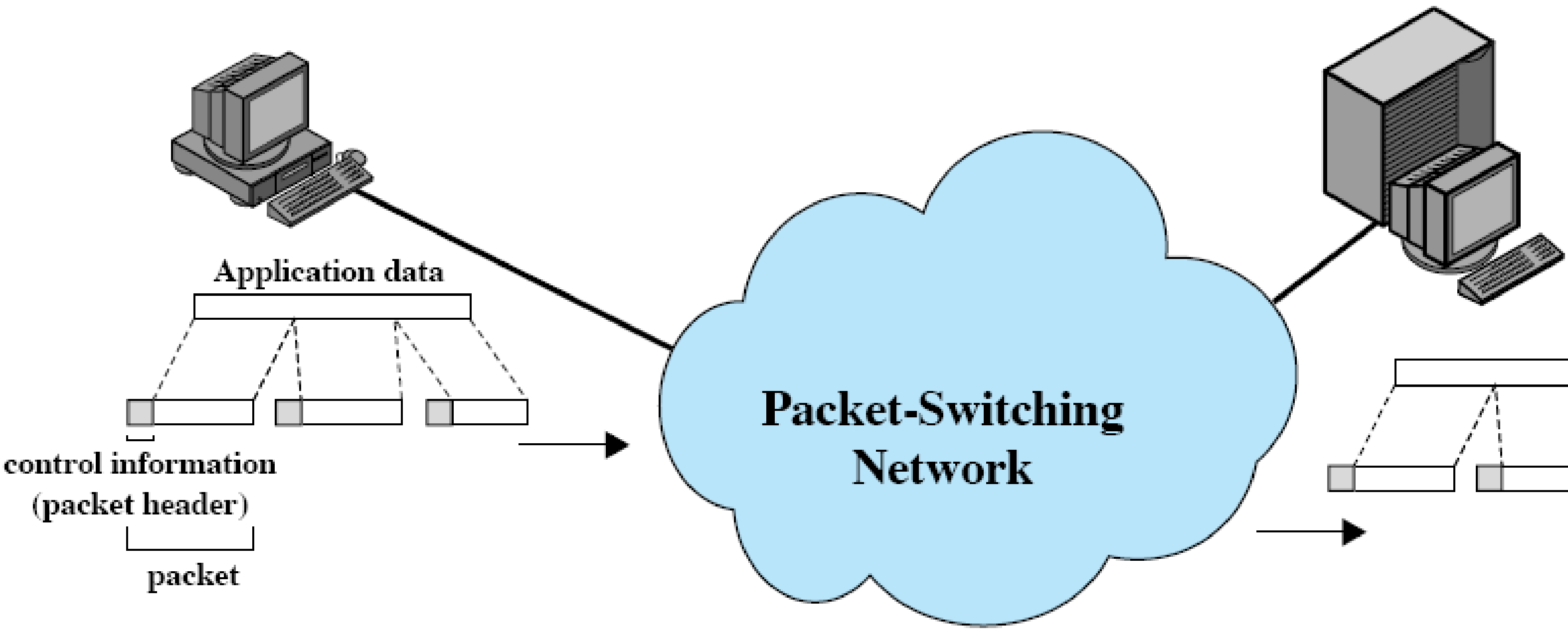
Characteristics of Circuit Switching

- Delay
  - Delay in connection setup, before data transfer can start
    - This is bad if only want to transfer small amount of data, or need almost instantaneous response time
  - Once data transfer starts, there are no queuing delays at switches
    - Transmission and propagation delay are the main contributors to total delay



Packet Switching

Packet Switching

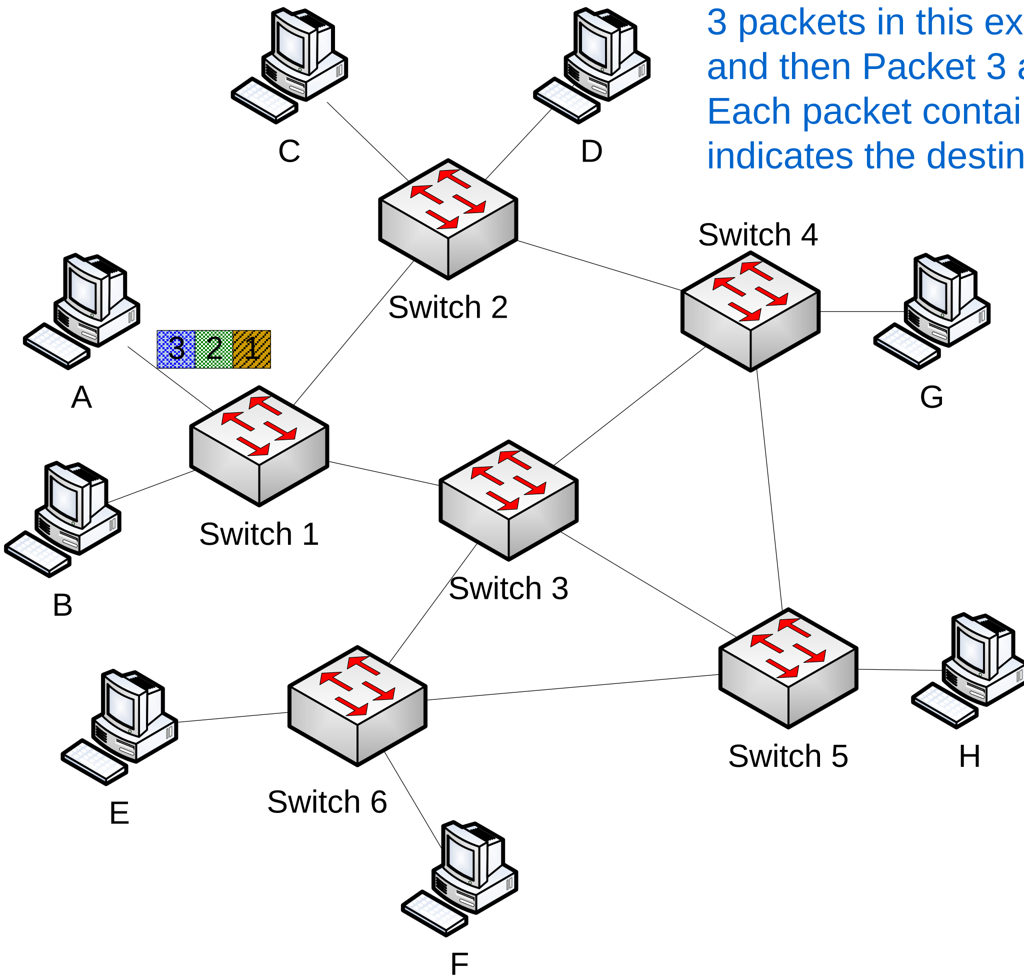


Packet Switching

- Packet switching breaks the data into packets, and sends the packets one at a time from source to destination across the network
  - Destination combines the received packets to get original data
- Two types of packet switching:
  - Datagram Packet Switching
    - Each packet is treated independently of all others
    - Packets belonging to the same message may:
      - Take different paths across the network
      - Arrive at destination out of order and may be lost
    - Packets need headers so switches know where to send them (destination)
  - Virtual Circuit Packet Switching
    - Virtual circuit setup and teardown (as in circuit switching)
    - Once setup, data is transferred as individual packets
      - Take the same path across the network
      - Arrive in-order at the destination, but may be lost
    - Packets need headers so switches know what is the next switch it must be sent to

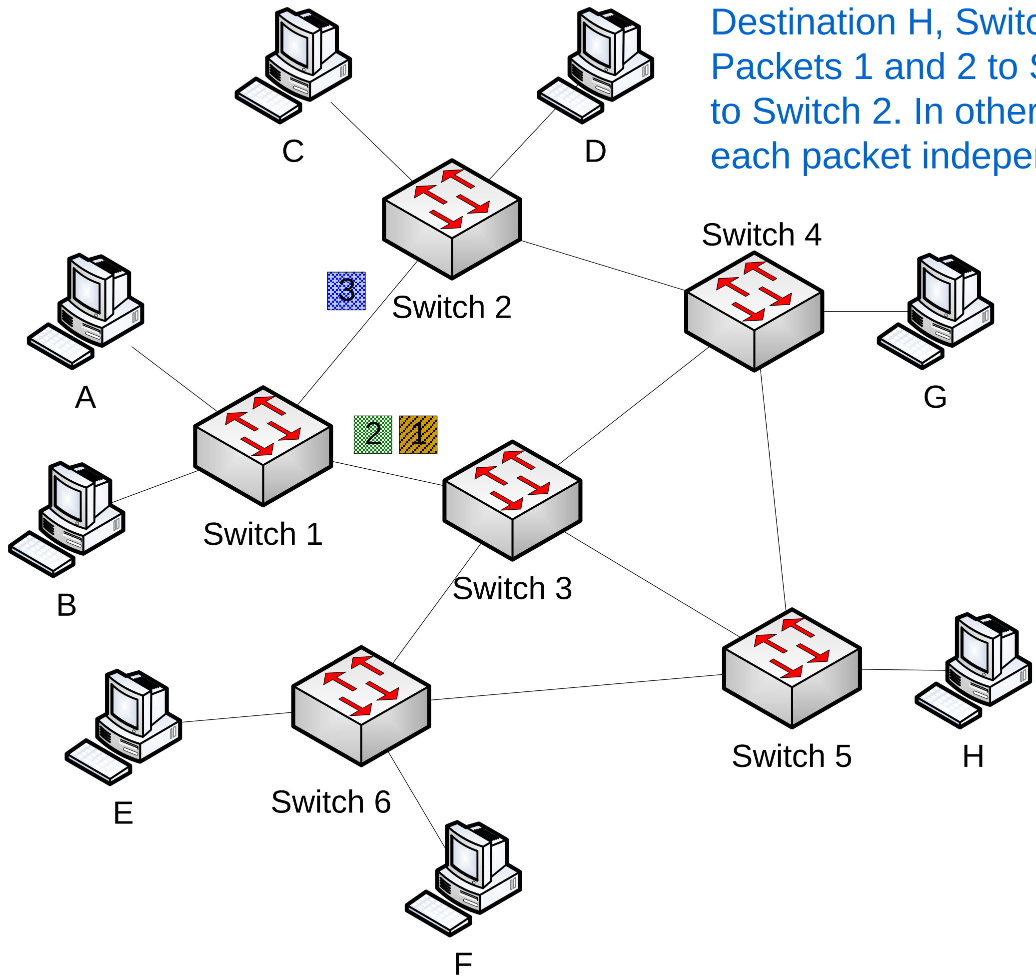
Datagram Packet Switching

Source A sends the original message as 3 packets in this example. Packet 1, Packet 2 and then Packet 3 are sent to Switch 1. Each packet contains a header which indicates the destination H.

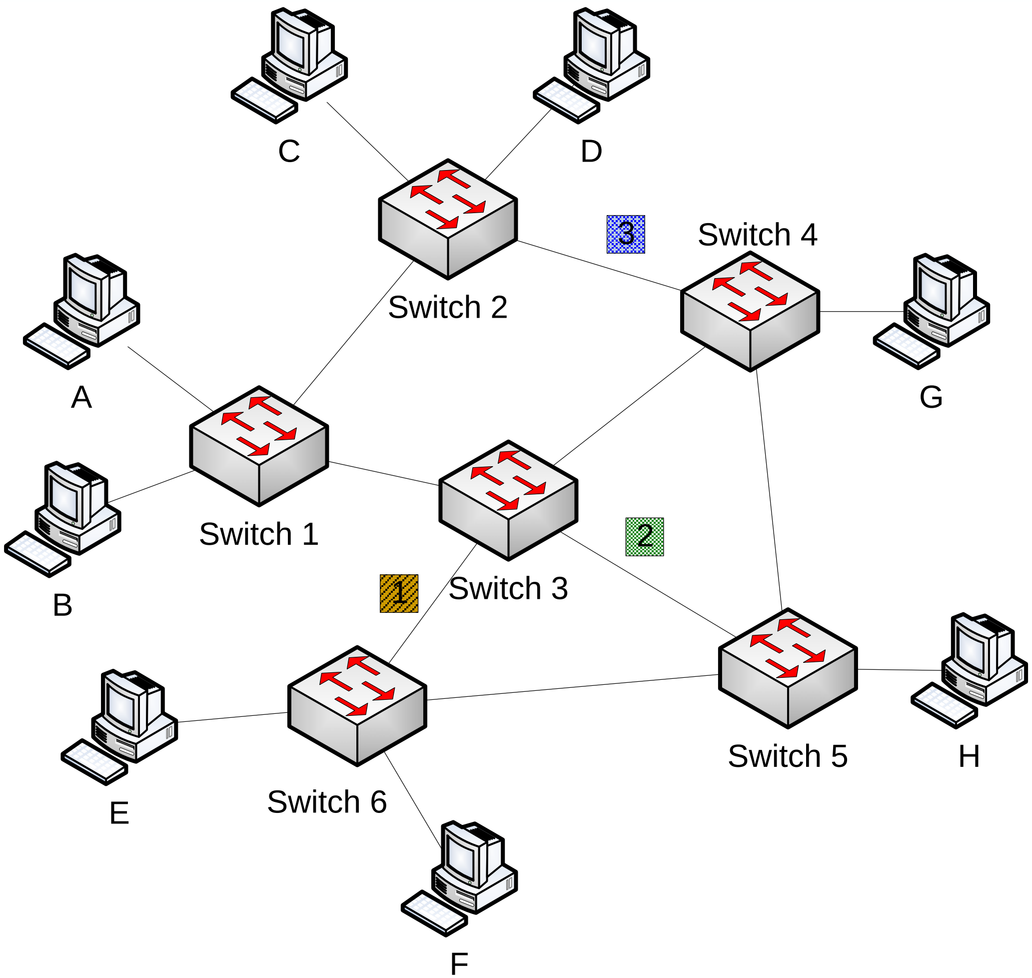


Datagram Packet Switching

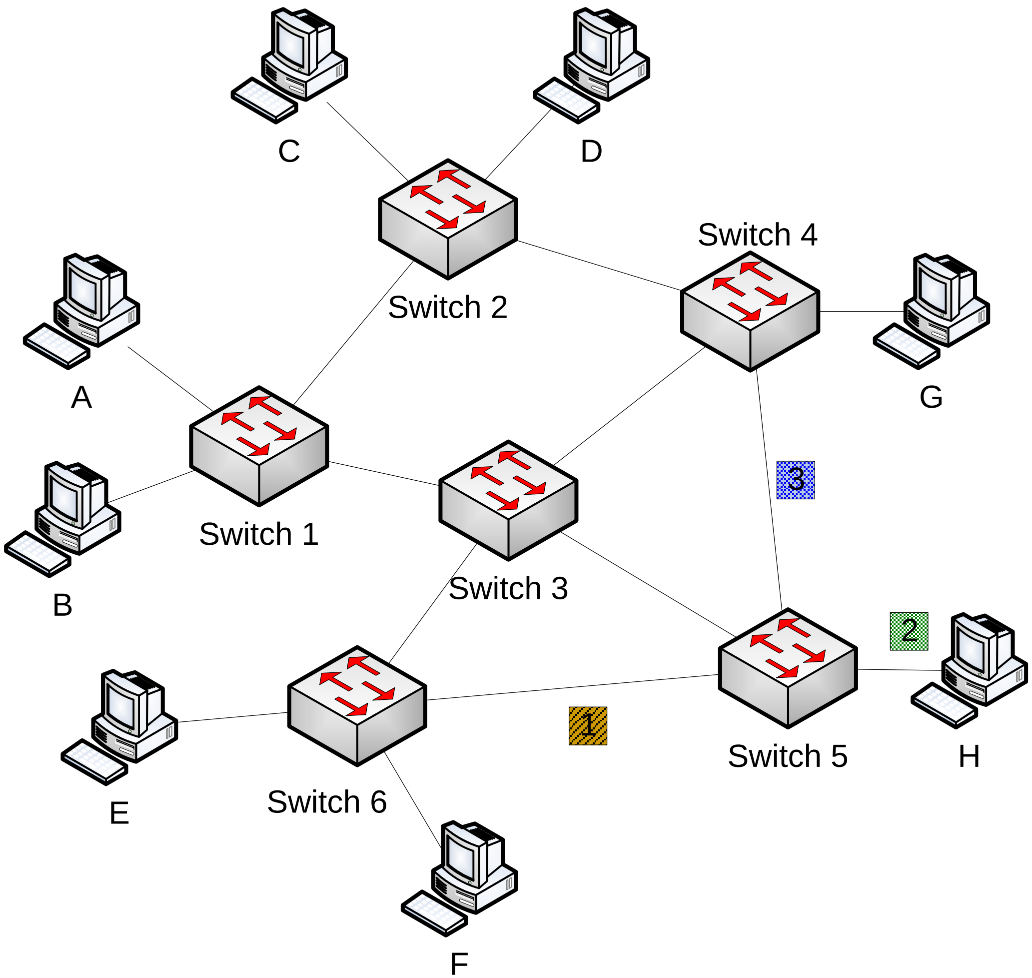
Assuming all switches know paths to the Destination H, Switch 1 decides to send Packets 1 and 2 to Switch 3, and Packet 3 to Switch 2. In other words, a switch treats each packet independently.



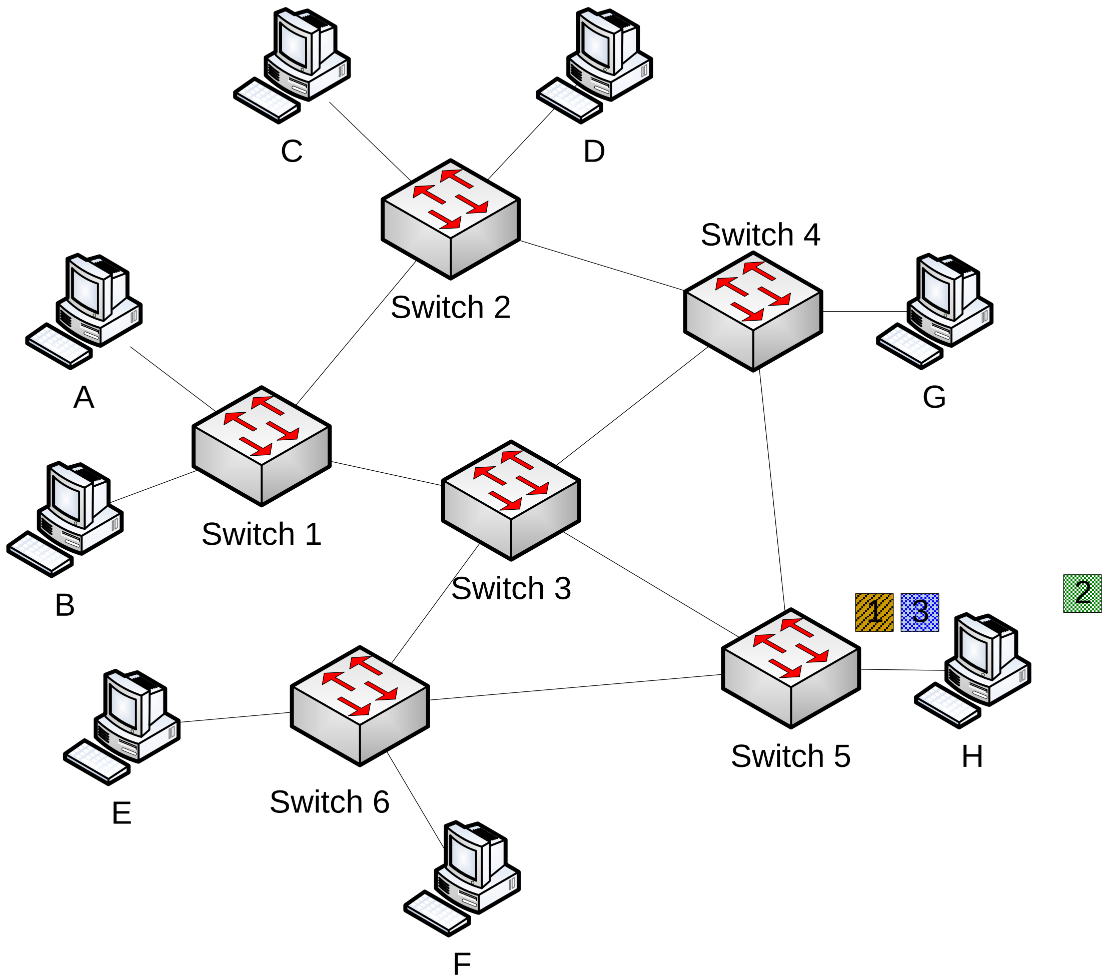
Datagram Packet Switching



Datagram Packet Switching

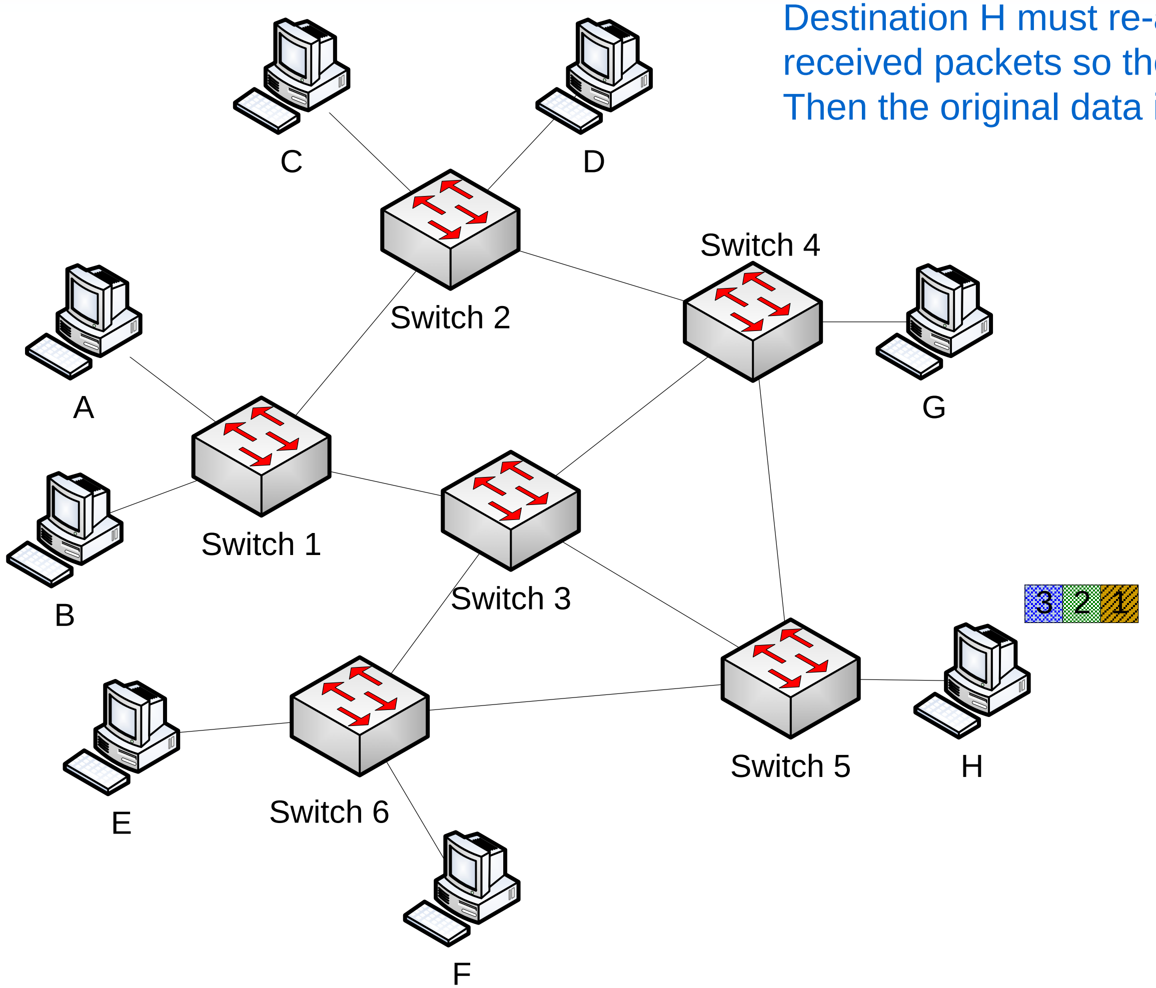


Datagram Packet Switching



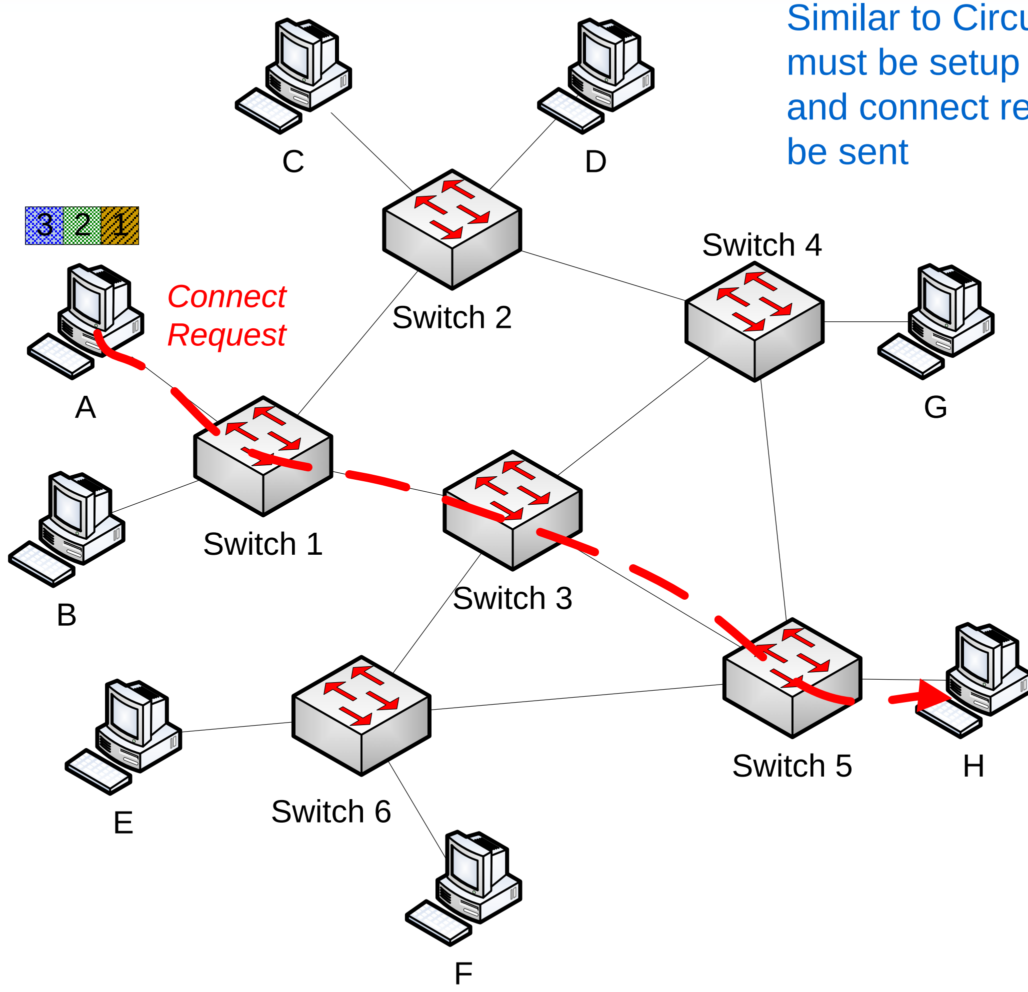
Datagram Packet Switching

Destination H must re-arrange the received packets so they are in-order. Then the original data is obtained.

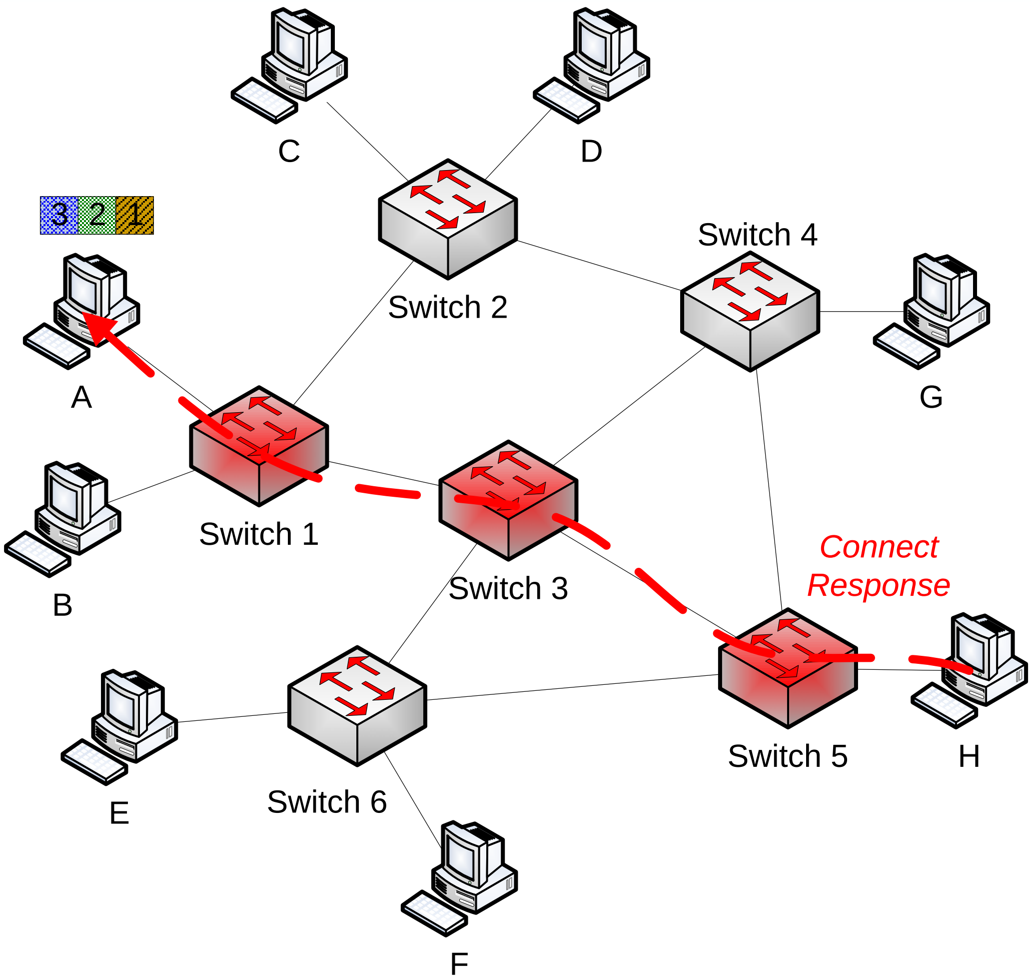


Virtual Circuit Packet Switching

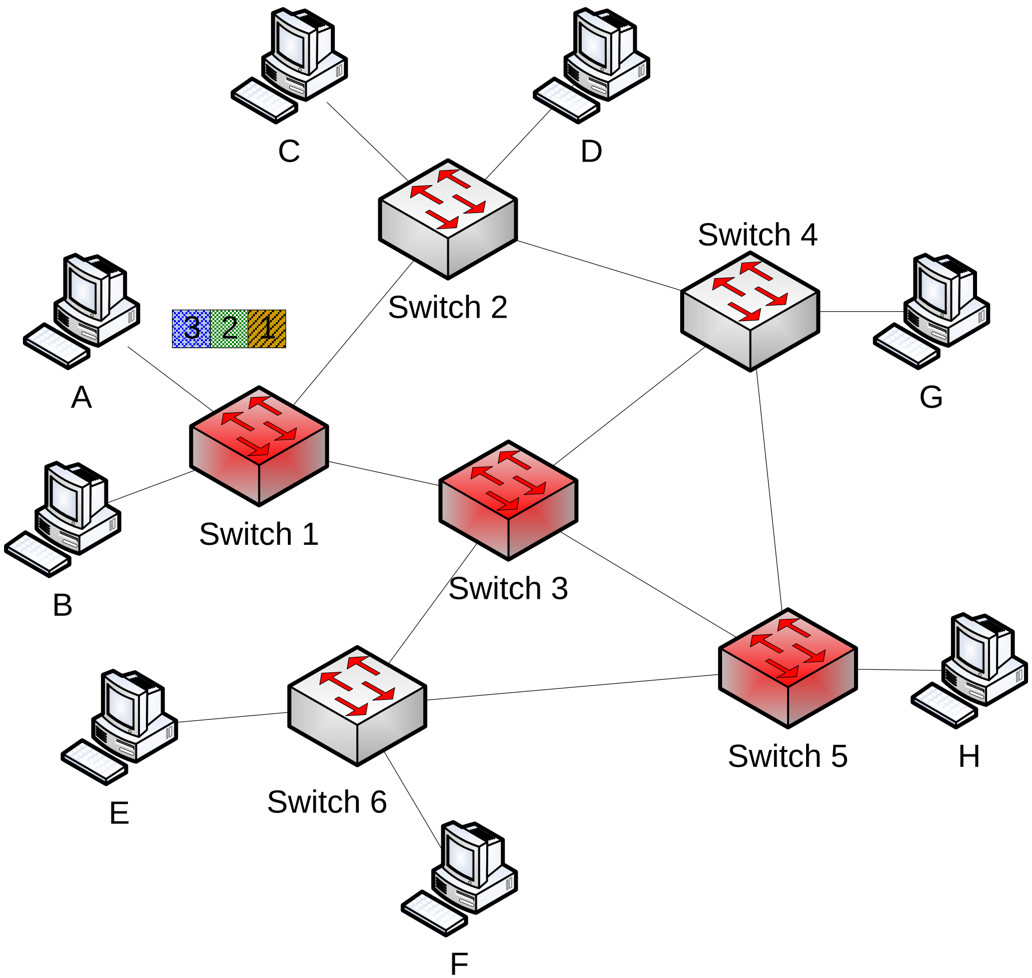
Similar to Circuit Switching, a connection must be setup (using connect request and connect response) before data can be sent



Virtual Circuit Packet Switching

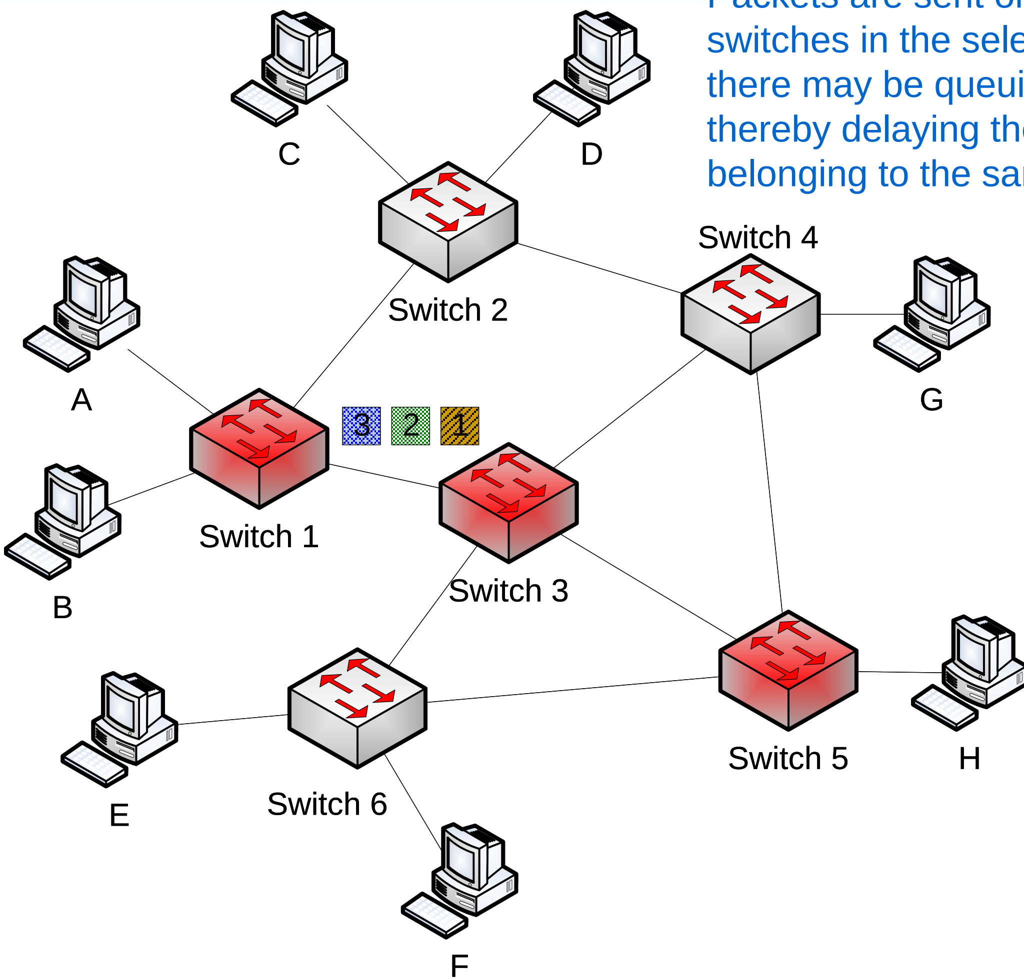


Virtual Circuit Packet Switching

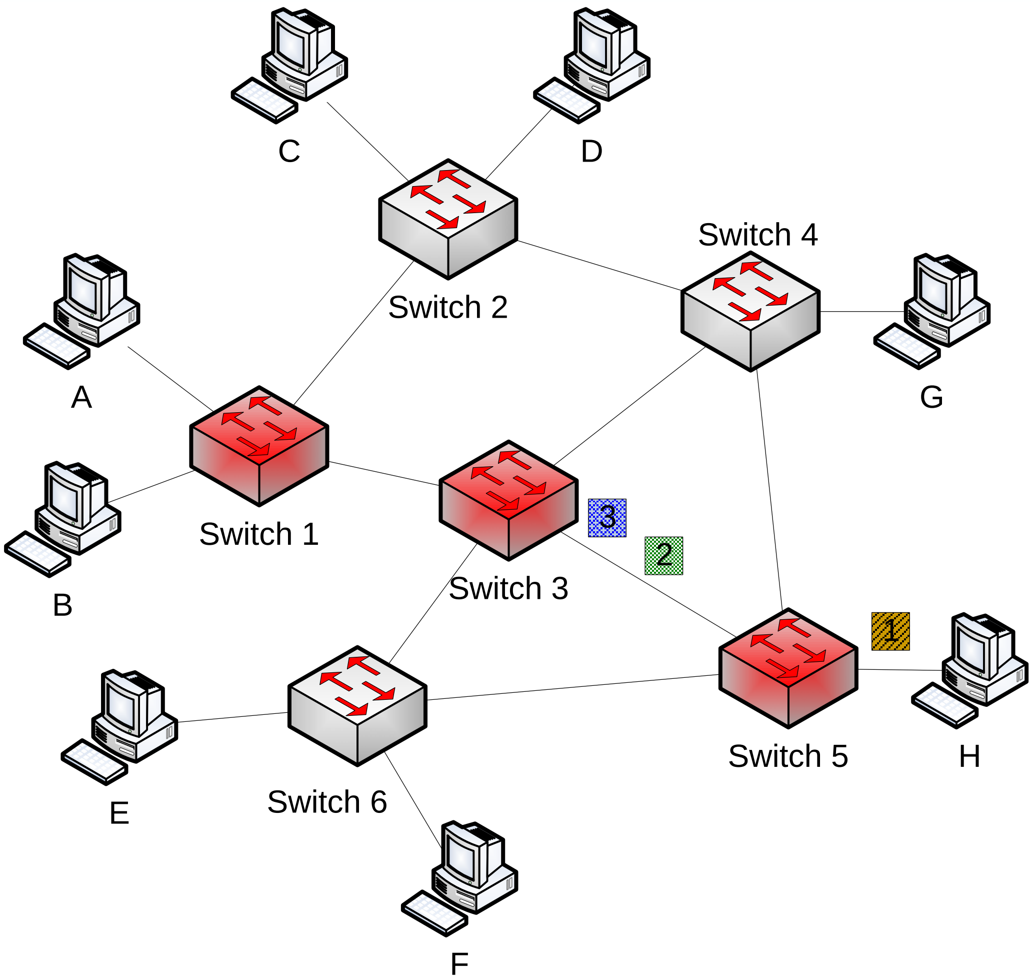


Virtual Circuit Packet Switching

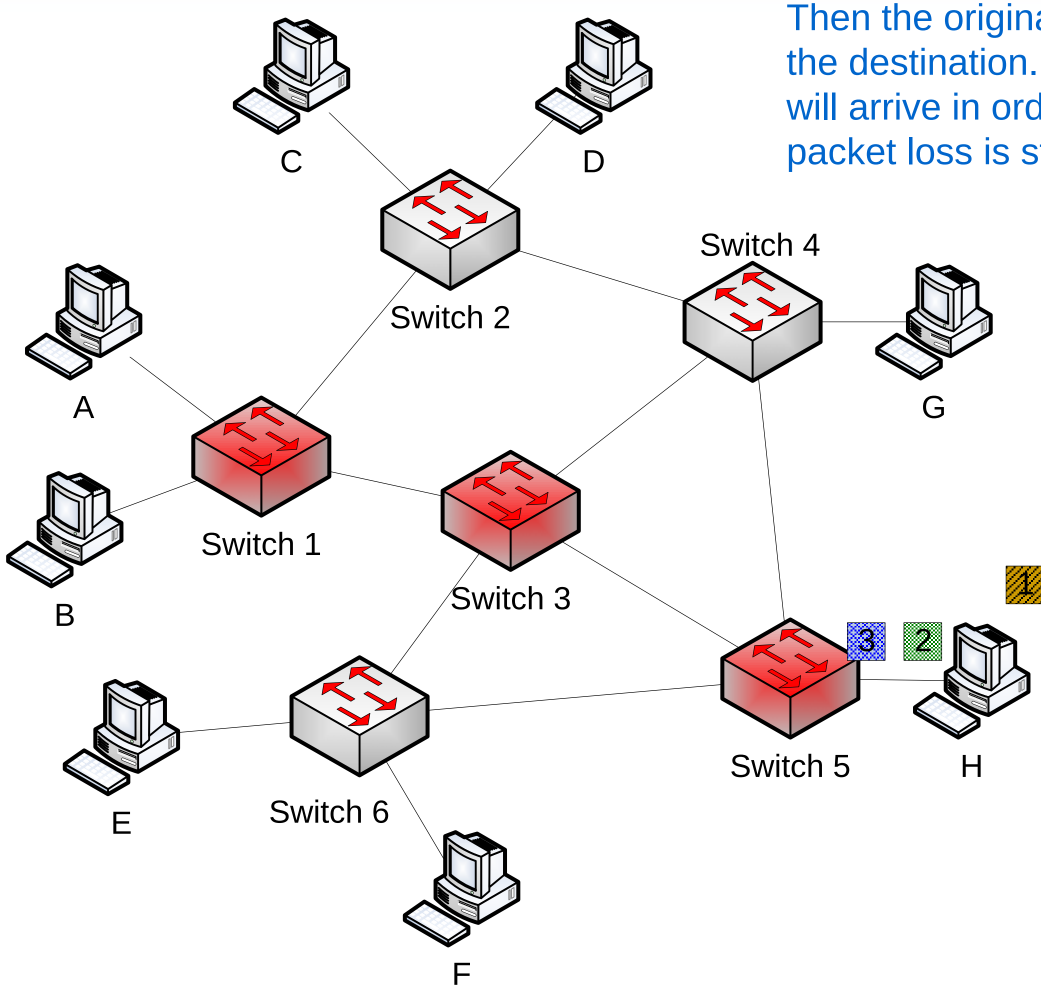
Packets are sent one at a time to the switches in the selected path. Note that there may be queuing delays in switches, thereby delaying the sending of Packets belonging to the same original data.



Virtual Circuit Packet Switching



Virtual Circuit Packet Switching



Then the original data is obtained at the destination. Note that the packets will arrive in order, although packet loss is still possible.

Comparing the Switching Techniques

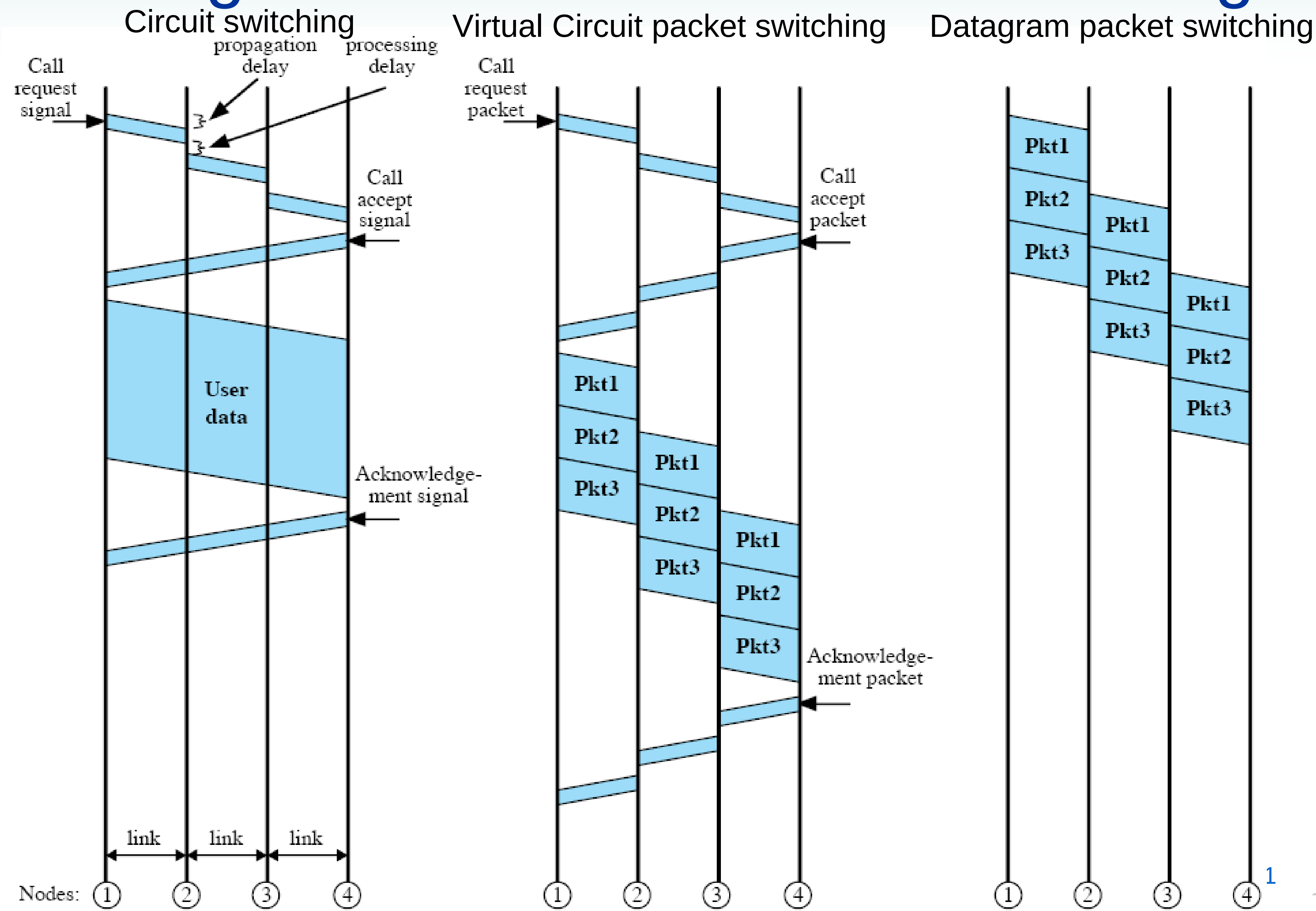
Advantages of Circuit Switching

- Guaranteed Quality of Service
  - With Circuit Switching, resources are allocated to circuits for the duration of connection
    - A voice call needs 4kHz of bandwidth – upon connection setup, 4kHz of bandwidth is allocated from source to destination
    - A video transfer may need maximum data rate of 1Mb/s – upon connection setup, 1Mb/s is allocated from source to destination
  - Data will not be dropped or delayed during the connection
  - (With Virtual Circuit Packet Switching, similar benefits apply, however, delays may not be guaranteed)
- Reduced complexity at switches
  - After a connection is setup, switch simply forwards the signal
  - No need to process headers to determine where to send data

Advantages of Packet Switching

- Line Efficiency (when amount of traffic varies)
  - A single switch-to-switch link can be shared by many packets over time
    - Packets are queued up and sent as fast as possible over link
  - In circuit switching, TDM is inefficient when connections are idle
- Data rate conversion
  - Two stations with different data rates can communicate
- Non-blocking
  - In circuit-switched, if no circuits, new connections are blocked
    - Example: with telephone calls over circuit switched network, if network supports 100 calls, then the 101st call will not be allowed
  - In packet-switched, packets are still accepted, but delay increases
    - Example: with telephone calls over packet switched network (e.g. VoIP), the 101st call (and subsequent calls) may be allowed, but the quality of every call will be reduced (extra delay)
- Priorities
  - Packets can be given priorities, and those with higher priorities transmitted first (giving them less delay)
    - Give high priority to Internet traffic from staff, low priority to Internet traffic from students!

Timing in Circuit and Packet Switching



Timing in Circuit and Packet Switching

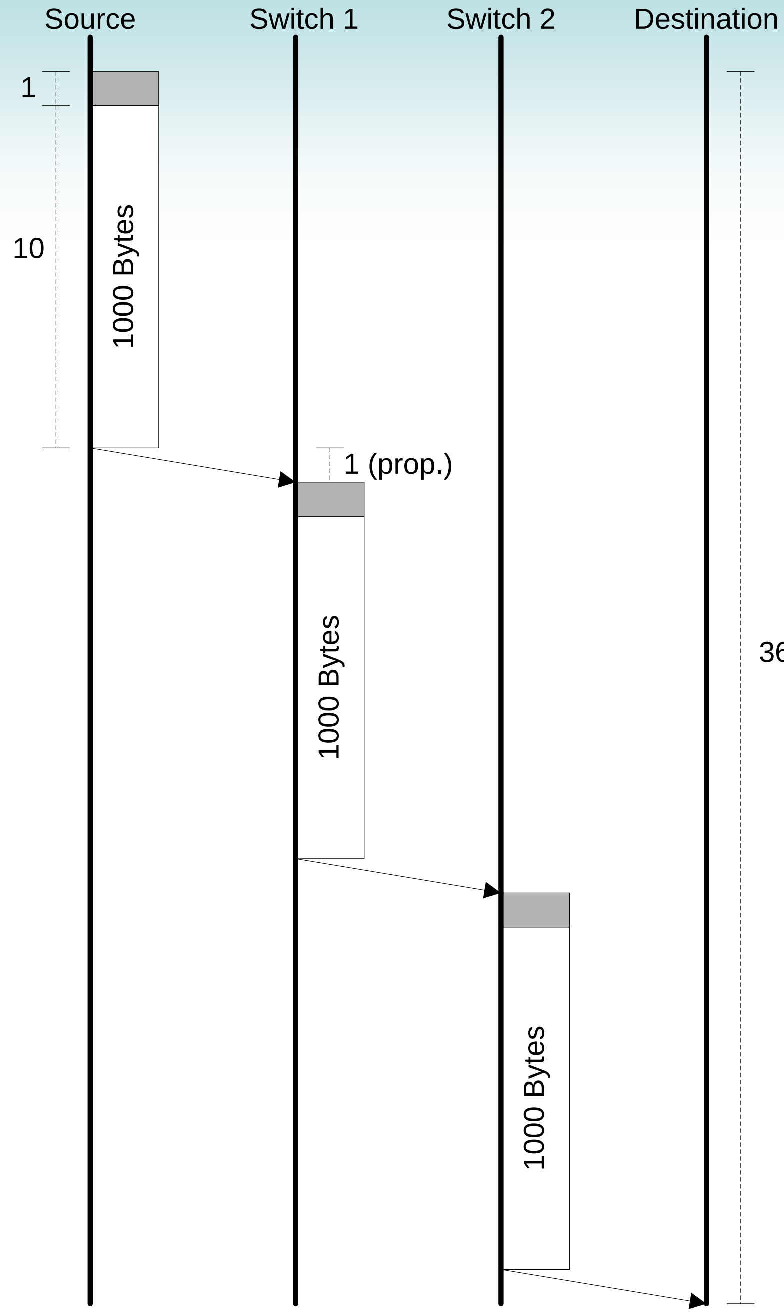
- Circuit Connection Setup
  - CS: special request signal sent and processed by all switches between source and destination. Switches setup circuit when receiving request. Call accept signal returned along circuit.
  - VCPS: special request packet sent and processed by all switches between source and destination. Accept packet returned along VC
- Data Transfer
  - CS: data is transmitted along circuit (effectively, there is a direct link between source and destination)
    - No headers; no queuing at switches; no (or almost 0) processing delay
  - PS: data is split into packets, and packets sent one by one over network
    - Packets include headers; small processing delay at switches; possible queuing delays at switches
    - Have to wait for entire packet to be received before sending on next link
- Circuit Disconnect
  - CS: special ACK signal sent when connection complete (either by source or destination)
  - VCPS: special ACK packet sent when connection complete (either by source or destination)

Timing in Circuit and Packet Switching

- Total delay depends on many factors
  - Number of links
  - Transmission rates and propagation delays for each link
  - Processing delays of switches
  - Size of data to send
  - For CS and VCPS only:
    - Size of special signals/packets
  - For PS only:
    - Size of packets, including header sizes
- General observations:
  - If the size of data is very large, then the time to setup a (virtual) circuit will be not have a large impact on total delay
    - Circuit Switching will therefore be faster than Packet Switching
  - If the size of data is very small, then Datagram Packet Switching will be fastest
    - E.g. to send 100 bytes of data, it is inefficient to send a connect request, then connect accept and then disconnect (each of which may be 100 bytes)
  - Packet sizes: depends on overhead plus number of links
    - See next slide

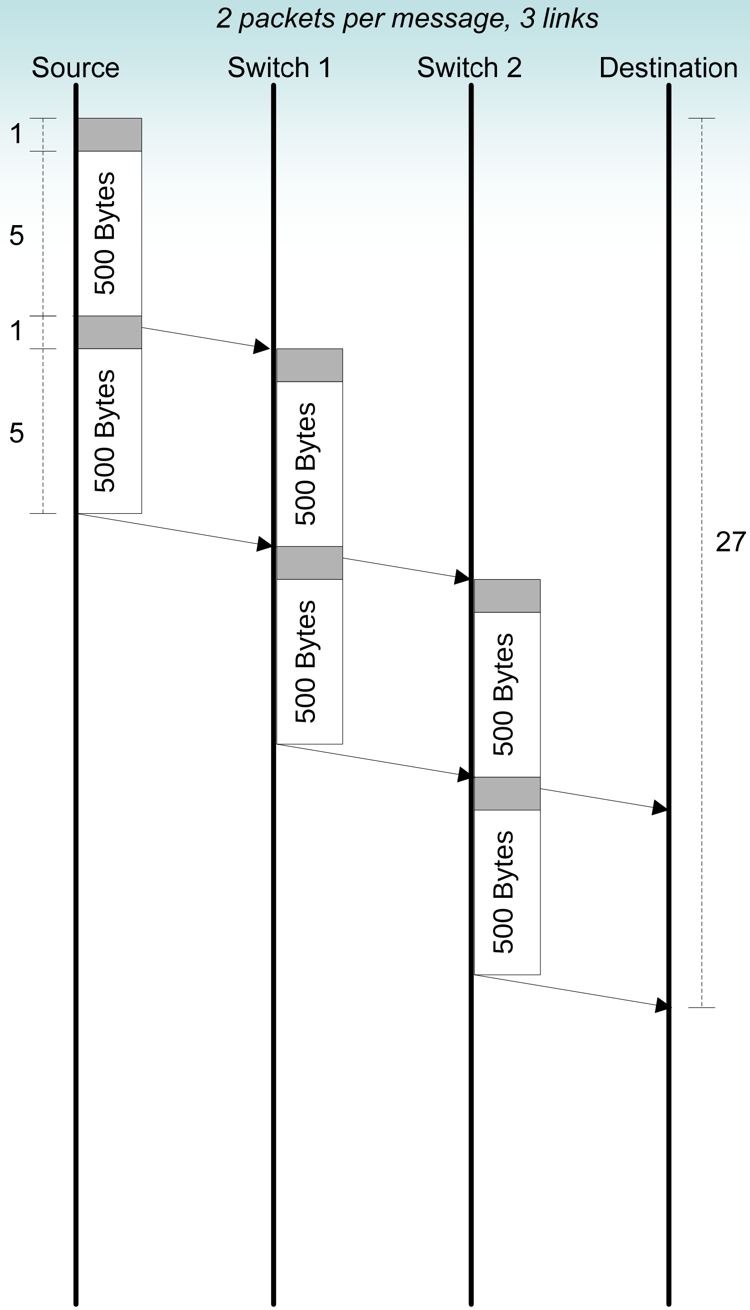
Lets assume:
Processing delay = 0
Queuing delay = 0
3 links, all the same
Data rate = 800kb/s
Propagation = 1ms
Message = 1000 Bytes
Header = 100 Bytes

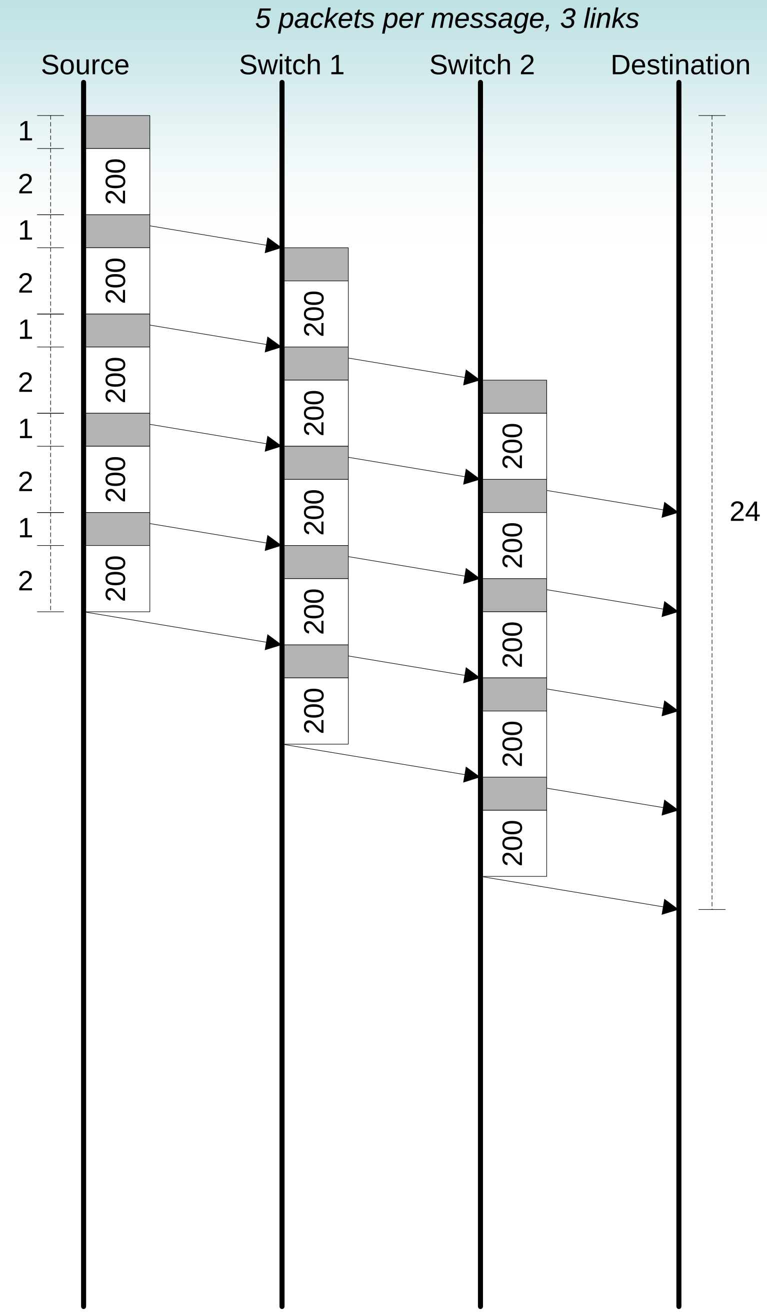
Therefore:
Header transmission = 1ms

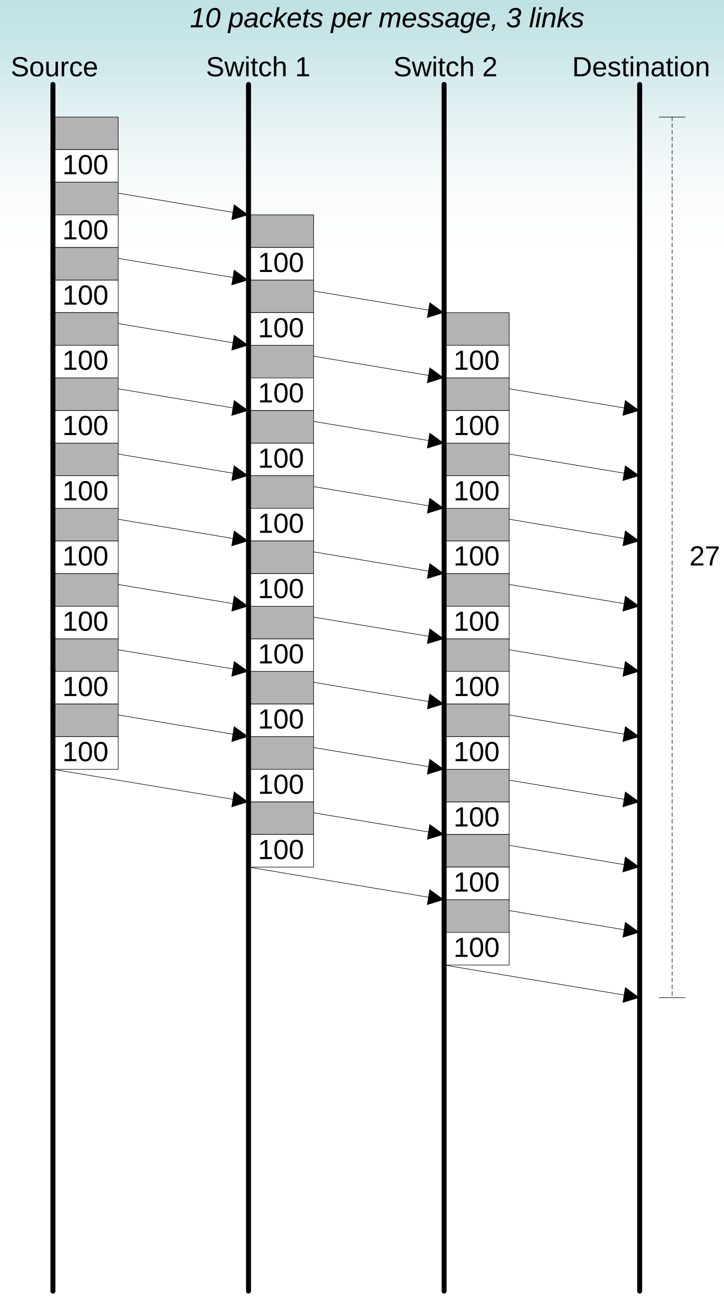


In this example, a packet has header plus 1000 bytes of data. That is, 1 packet carries entire message.

Total delay


2 packets per message, 3 links
Source
Switch 1
Switch 2
Destination
1
5
1
5
500 Bytes
500 Bytes
500 Bytes
500 Bytes
500 Bytes
500 Bytes
27


5 packets per message, 3 links
Source
Switch 1
Switch 2
Destination
1
2
1
2
1
2
1
2
1
2
200
24



- With a single link, a large packet size was more efficient (because overhead of header is less)
  - However, due to errors, buffers, and fairness, often we want small packets
- With a network (of multiple links), the efficiency is more complex!
  - With multiple links, it is more efficient if we make use of all links at the same time, e.g.
    - Source send packet 3 to Switch 1
    - Switch 1 send packet 2 to Switch 2
    - Switch 2 send packet 1 to Destination
    - ALL AT THE SAME TIME!
  - This depends on number of links, and number of packets
- Note: not a big issue in moderate to large networks, because there are generally packets from multiple sources over links

Queuing Delay

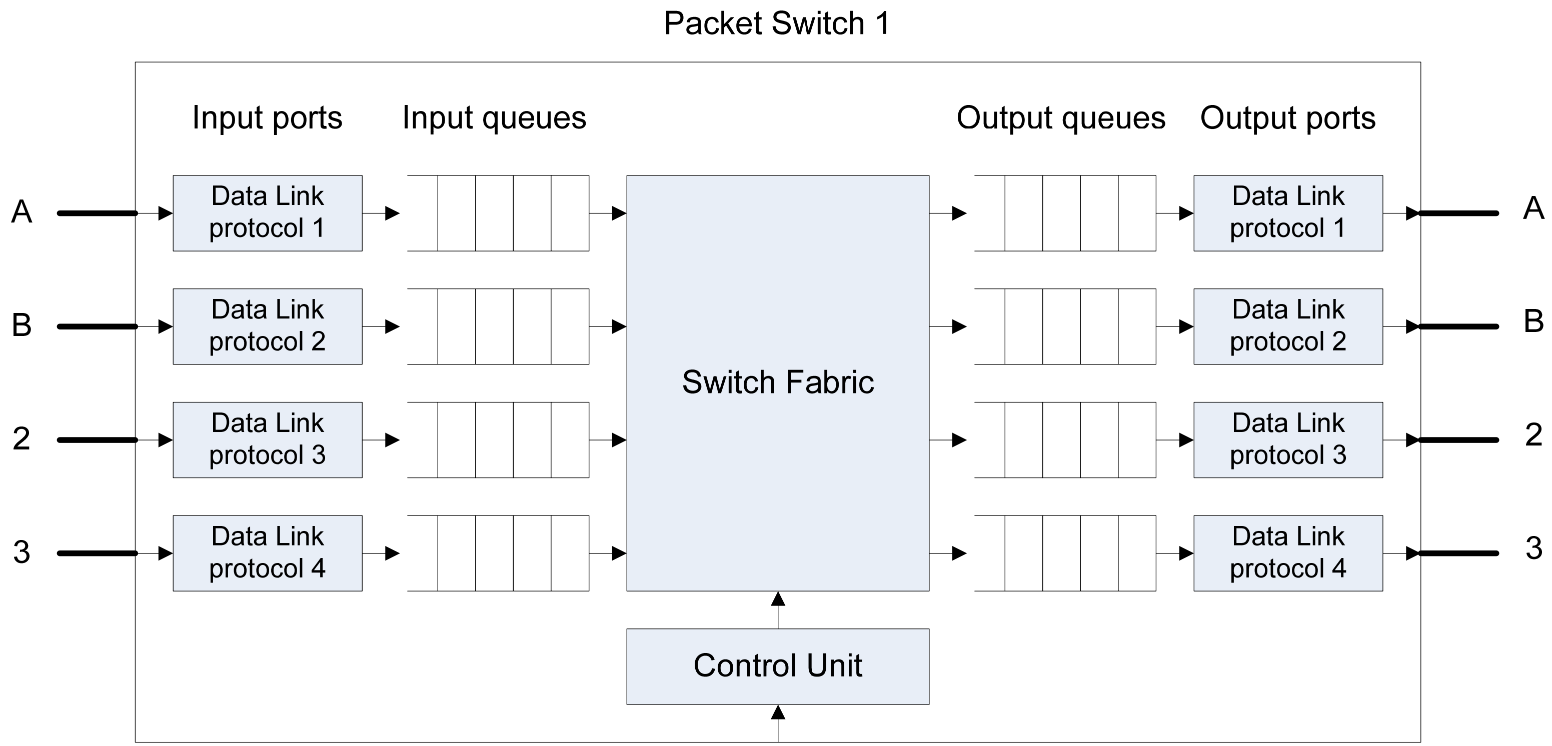
Packet Switching

Packet Switching Nodes

- The details of a packet switch:
  - Input physical and data link layer processing of received signal/frames
  - Input queue: stores the received packet until switched to output
  - Switch fabric: connects the input queue to output queue
    - In practice this has a speed limit: how many packets can it process at a time (or packets per second)
  - Output queue: stores the packet until ready to be sent
  - Output data link and physical layer processing to send frames/signals
  - Control Unit: tells the switch fabric how inputs should be connected to outputs (e.g. based on packet headers, virtual circuits)

- Next slide shows the blocks of a packet switch for Switch 1 in our example network
  - 4 inputs: A, B, Switch 2 and Switch 3
  - 4 outputs: A, B, Switch 2 and Switch 3

Packet Switching Nodes

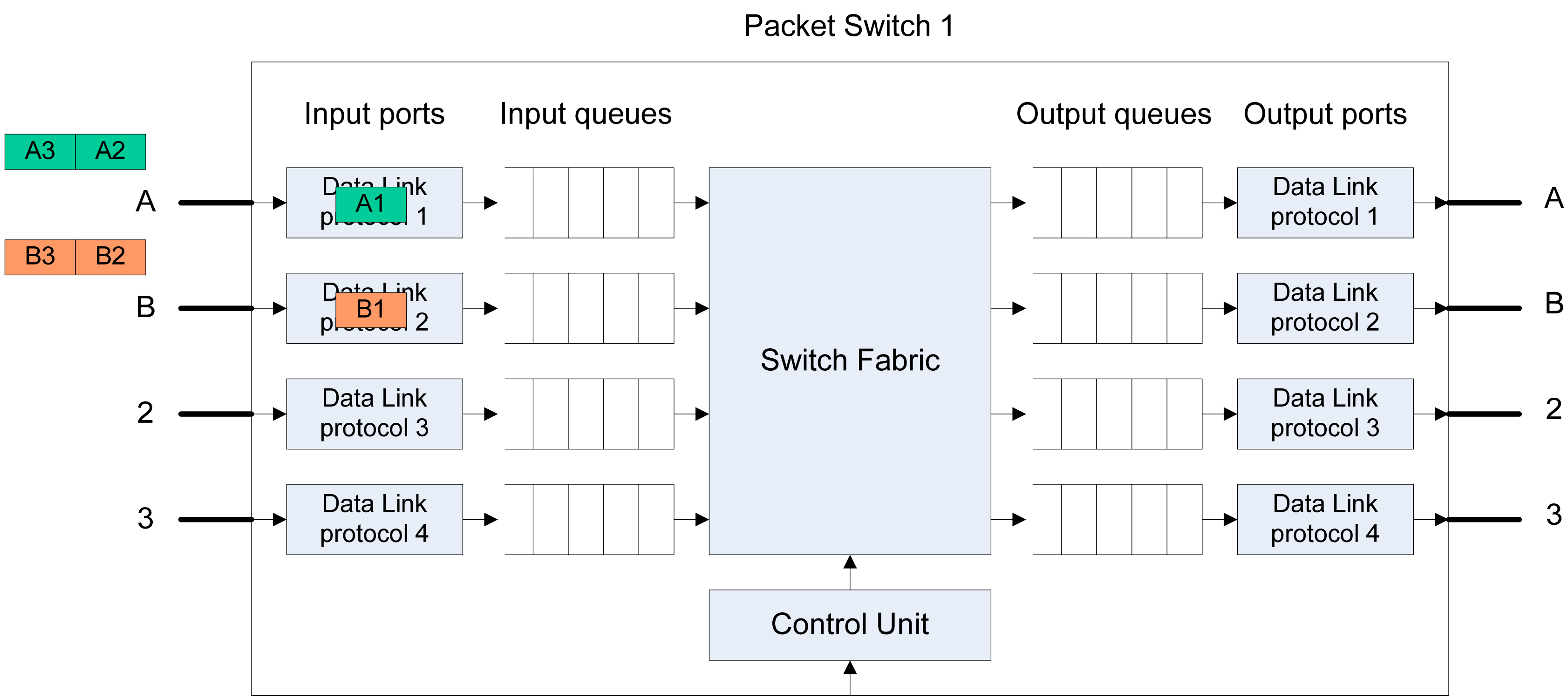


Queuing at Packet Switching Nodes

- Example at Switch 1. Lets assume:
  - A is sending to H (via switch 1, 3 and 5)
  - B is sending to D (via switch 1 and 2)
  - The data rates of input and output lines are each 1 packet per second
    - E.g. Arrive from A at 1 packet per second; can be sent to Switch 2 at 1 packet per second
  - Switch fabric can process up to 4 packets at a time
    - Delay incurred in switch fabric is 0.01sec
  - Processing time for Data Link and Physical layer at switch is 0.01sec
  - Processing time for adding packet in queue and removing packet from queue is 0

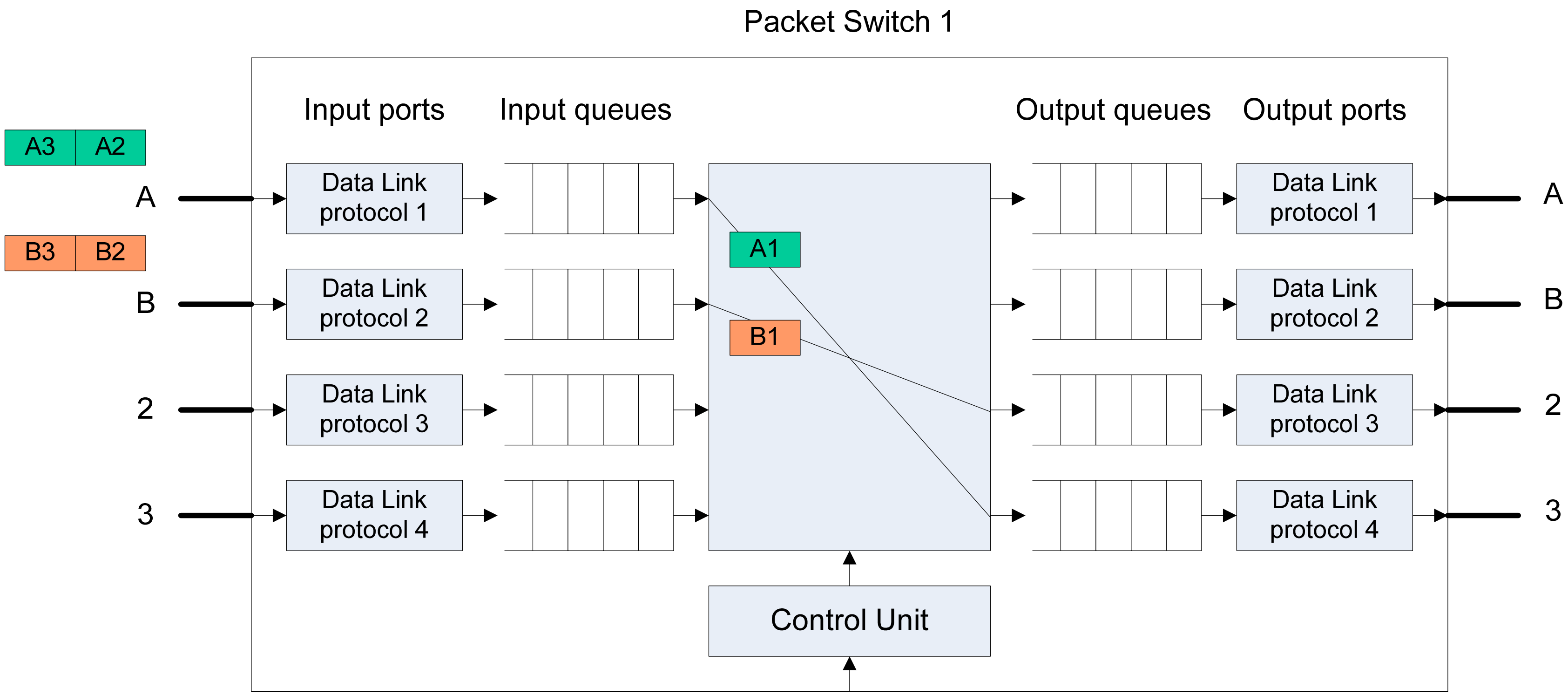
- (Note: these are unrealistic values, but easy to calculate)

Example 1: Packet Arrival



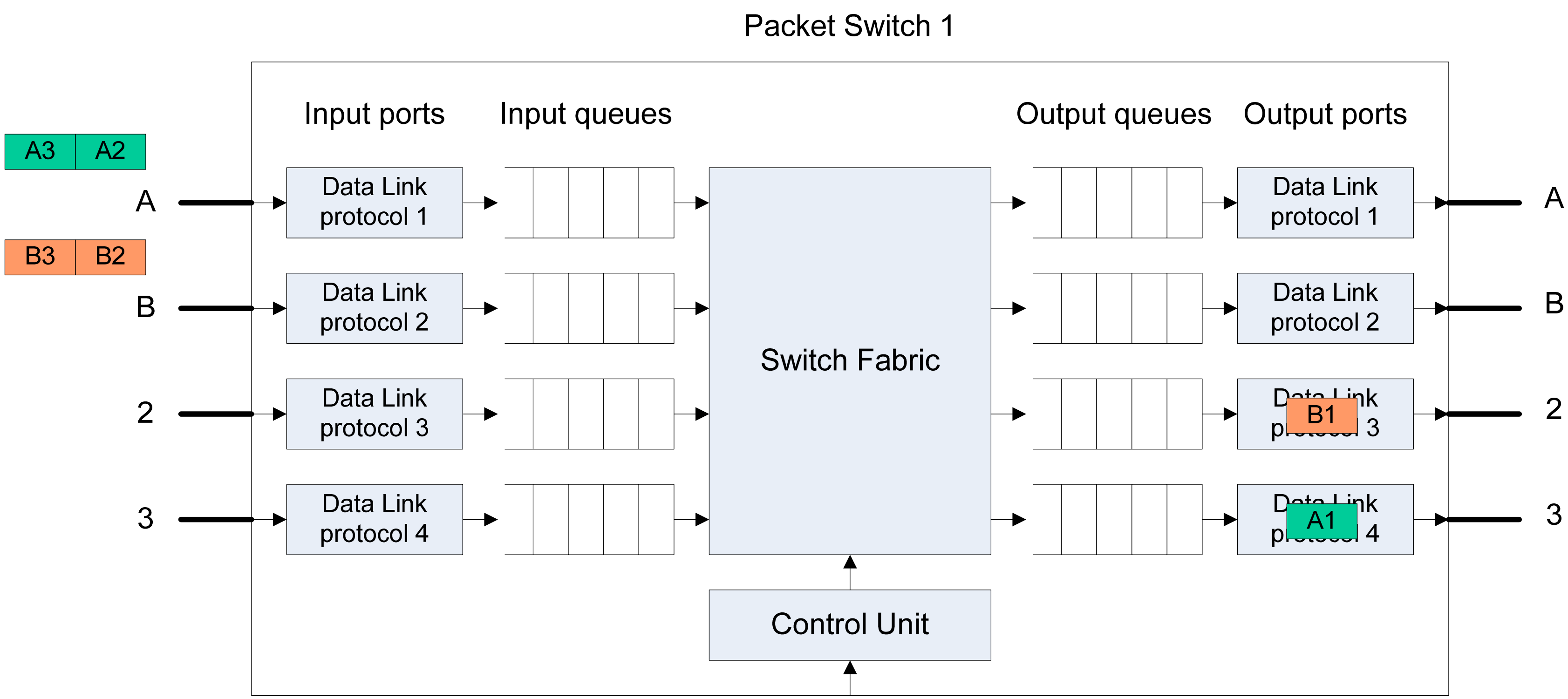
A1: processing delay = 0.01
B1: processing delay = 0.01

Example 1: Switching Packets



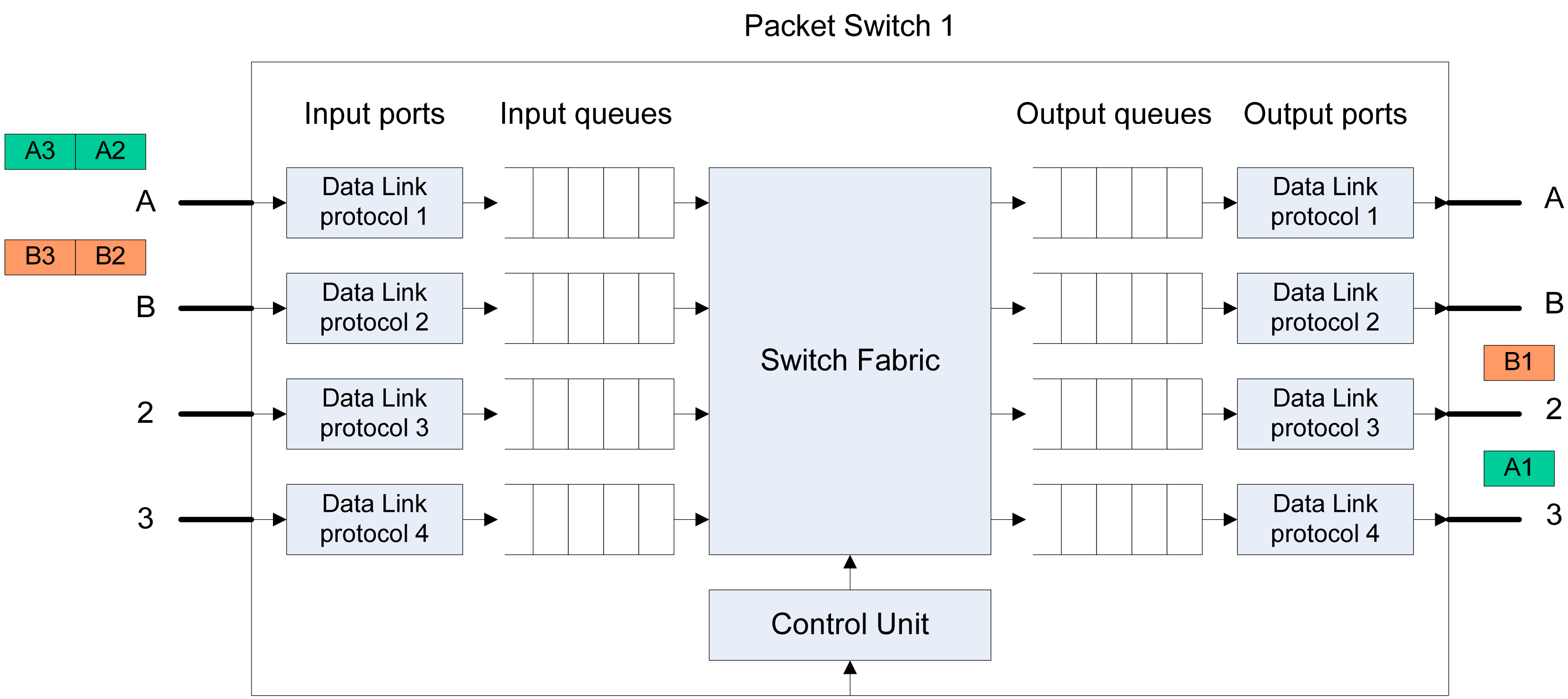
A1: processing delay = 0.01 + 0.01
B1: processing delay = 0.01 + 0.01

Example 1: Packet Output



A1: processing delay = 0.01 + 0.01 + 0.01
B1: processing delay = 0.01 + 0.01 + 0.01

Example 1: Packet Transmission



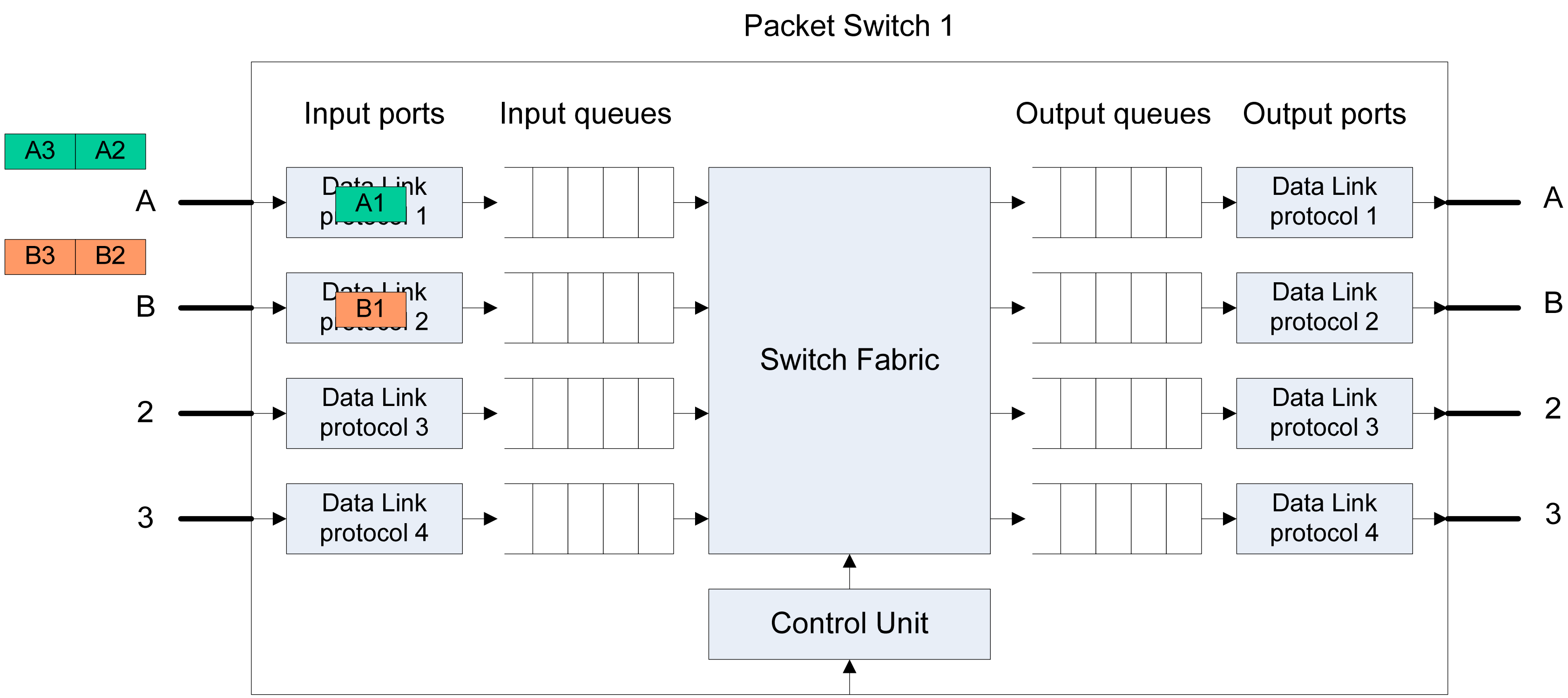
A1: processing delay = 0.03; queuing delay = 0

B1: processing delay = 0.03; queuing delay = 0

Queuing at Packet Switching Nodes

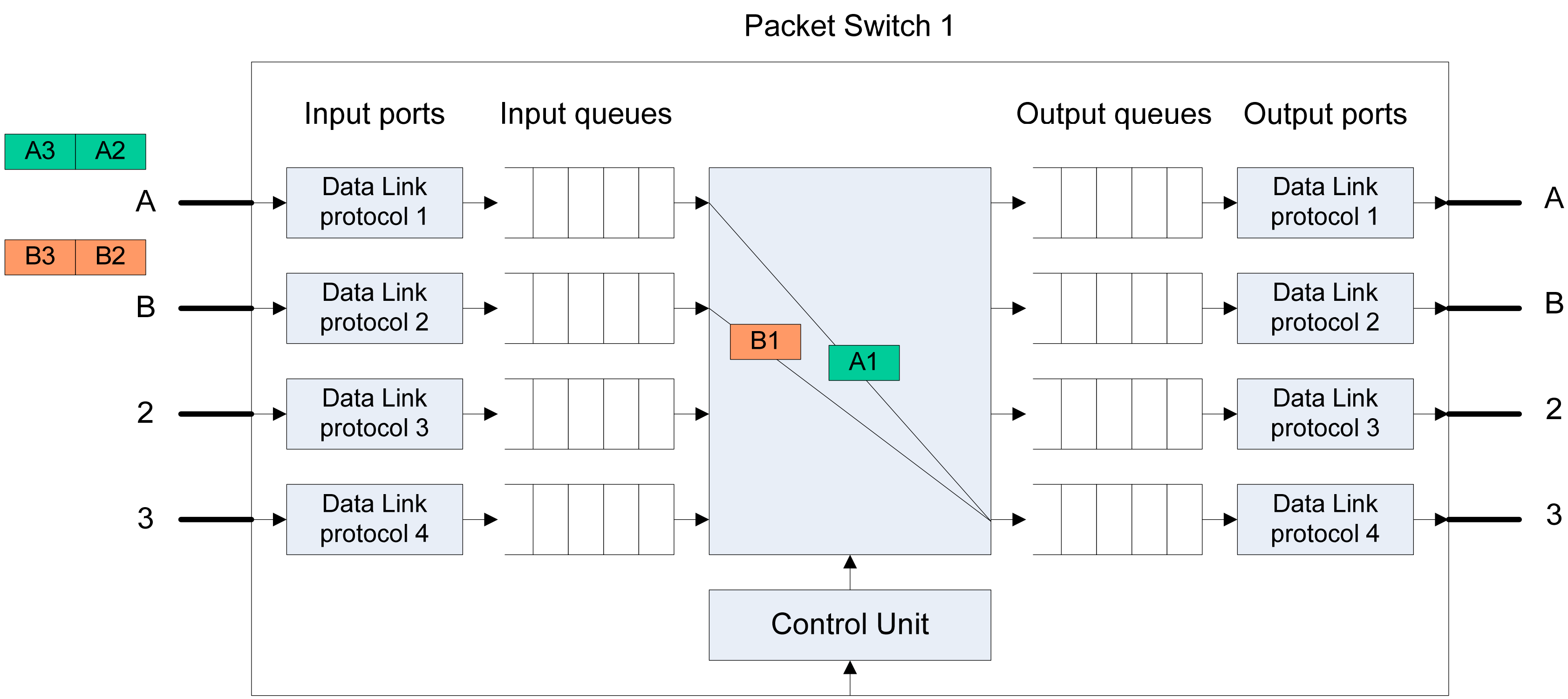
- **Another** example at Switch 1. Lets assume:
  - A is sending to H (via switch 1, 3 and 5)
  - **B is sending to H (via switch 1, 3 and 5)**
  - All other parameters are same as previous example

Example 2: Packet Arrival



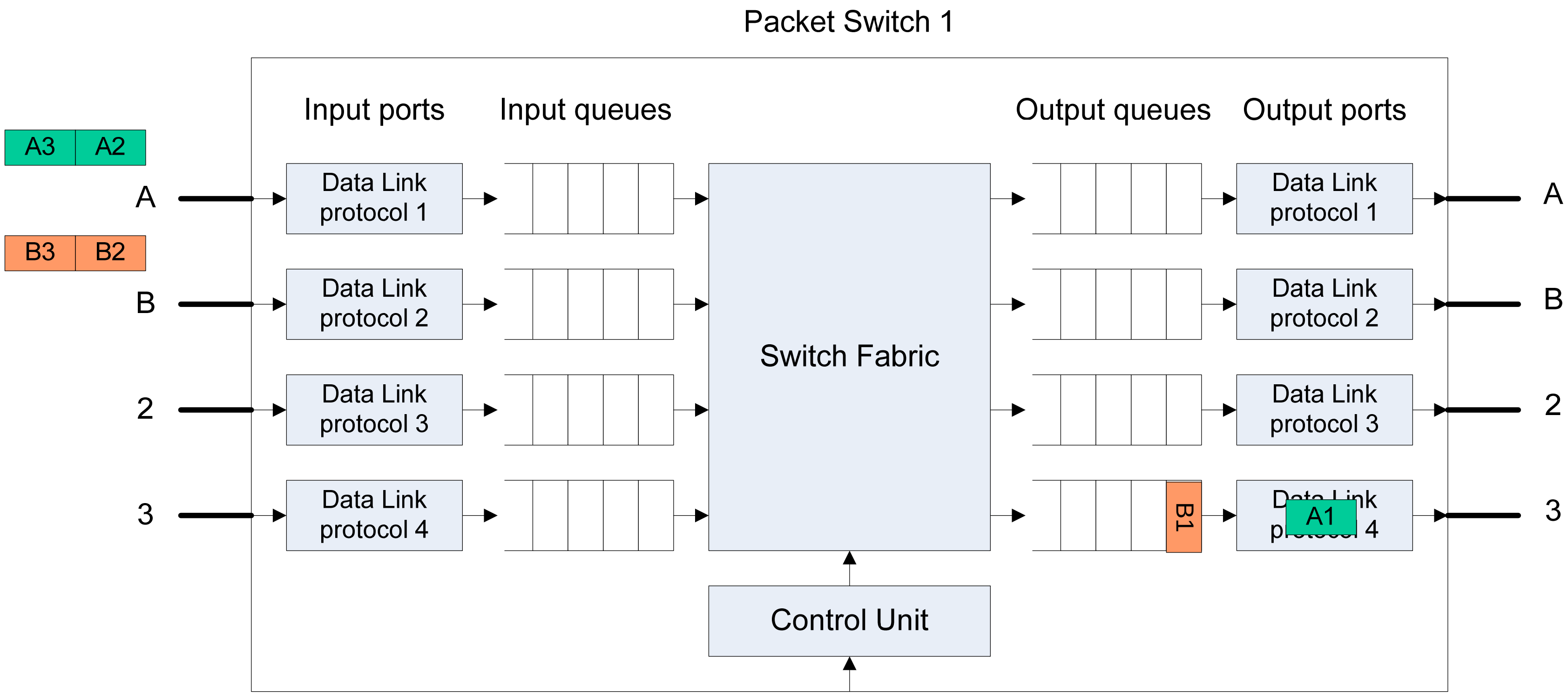
A1: processing delay = 0.01
B1: processing delay = 0.01

Example 2: Packet Switching



A1: processing delay = 0.01 + 0.01
B1: processing delay = 0.01 + 0.01

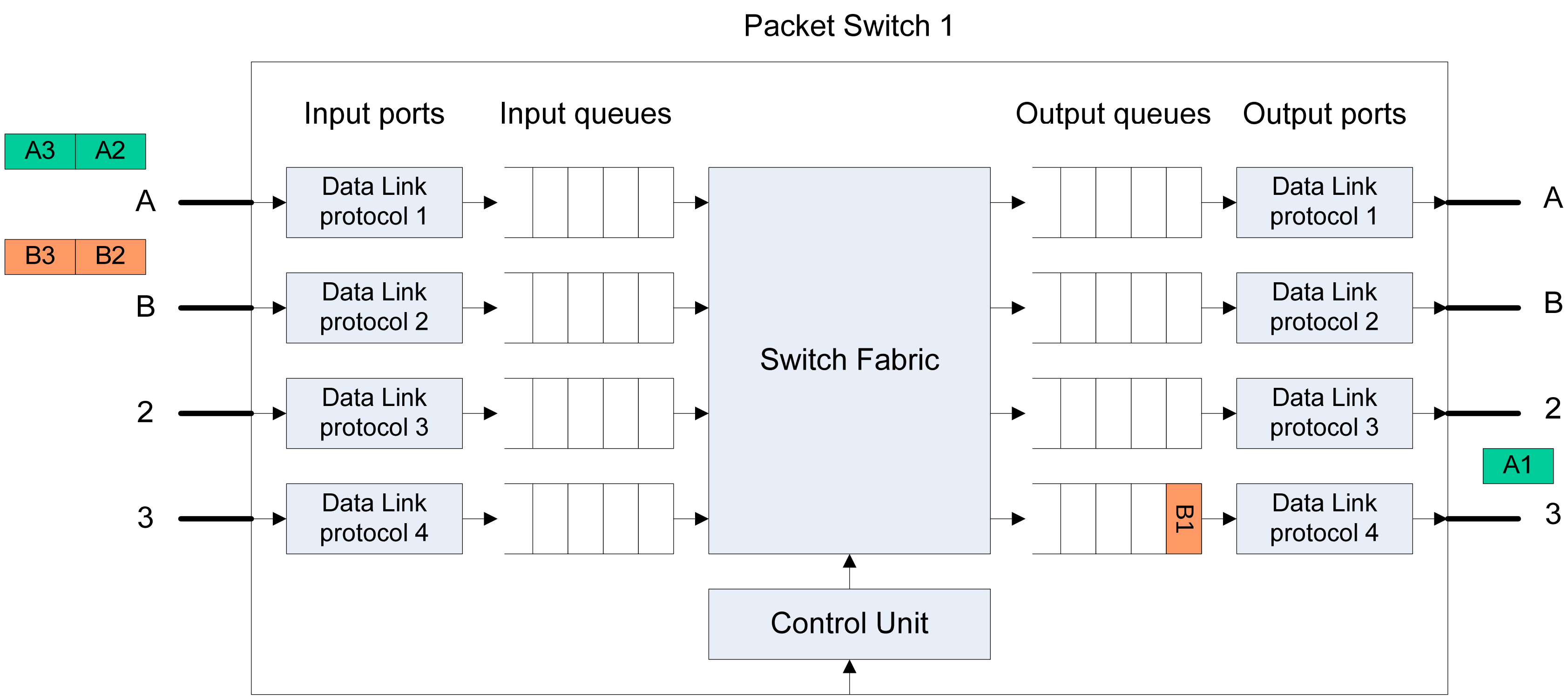
Example 2: Packet Queuing



A1: processing delay = 0.01 + 0.01 + 0.01

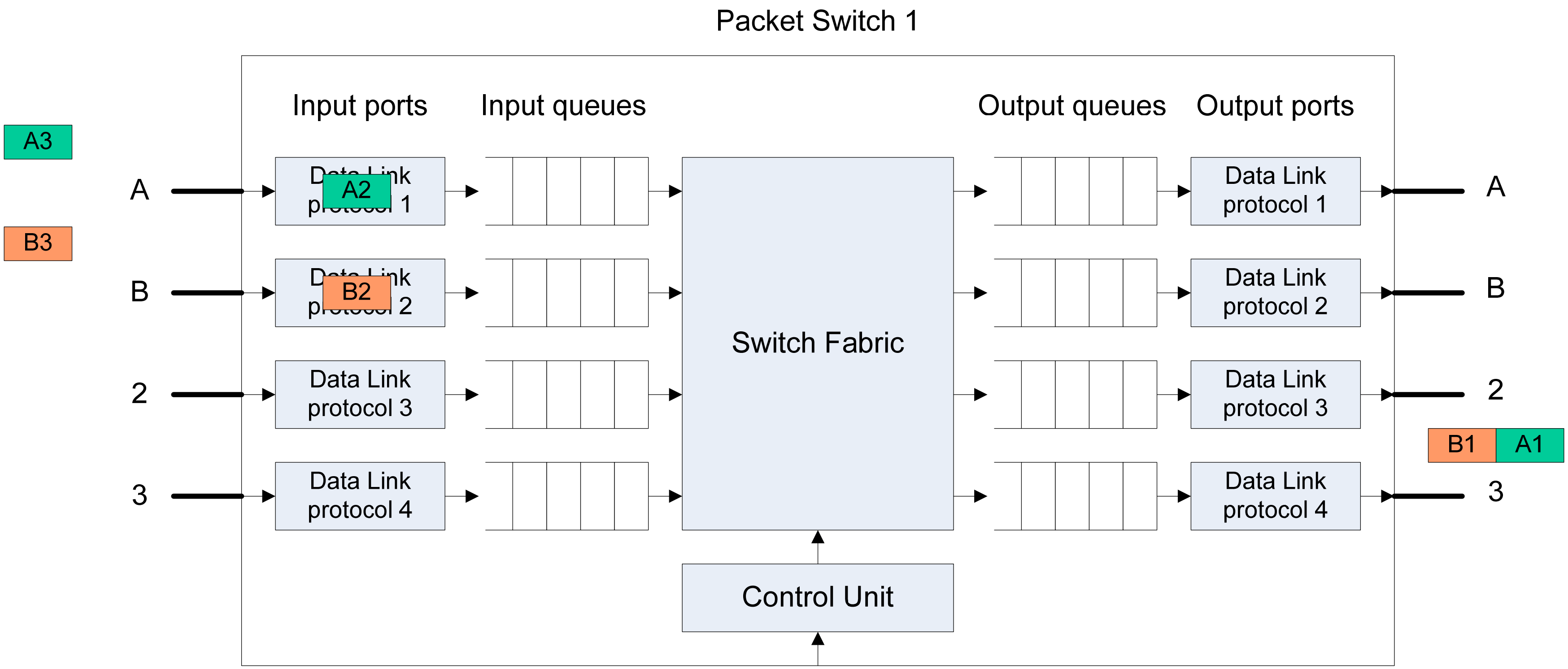
B1: processing delay = 0.01 + 0.01; queuing delay = 0.01

Example 2: Packet Queuing



B1 must be wait in the queue while A1 is transmitted. If output line rate is 1 packet per second, then B1 is in the queue for 1 second
Queuing Delay = 1.01 sec for B1

Example 2: Packet Transmission



A1: processing delay = 0.03

B1: processing delay = 0.03; **queuing delay = 1.01**

Queuing at Packet Switching Nodes

- Queuing delay is most complex of the four delay components (transmission, propagation, processing)
  - Depends on packet arrival rate and distribution (which is hard to know)
  - Depends on packet processing speed of switch
  - Depends on output data rates
  - May be different for each packet
- In packet switching communication networks, queuing delay will often change over time, and be different in different parts of a network
  - Generally, the more traffic to send, the higher the queuing delay