# Networks: Tutorial 07-08

Shinnazar Seytnazarov, PhD

Innopolis University

s.seytnazarov@innopolis.ru

Febuary 11, 2022

# Topic of the lecture

- Introduction to Transport Layer
- IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- What's inside a router
- Virtual LANs

# Topic of the tutorial

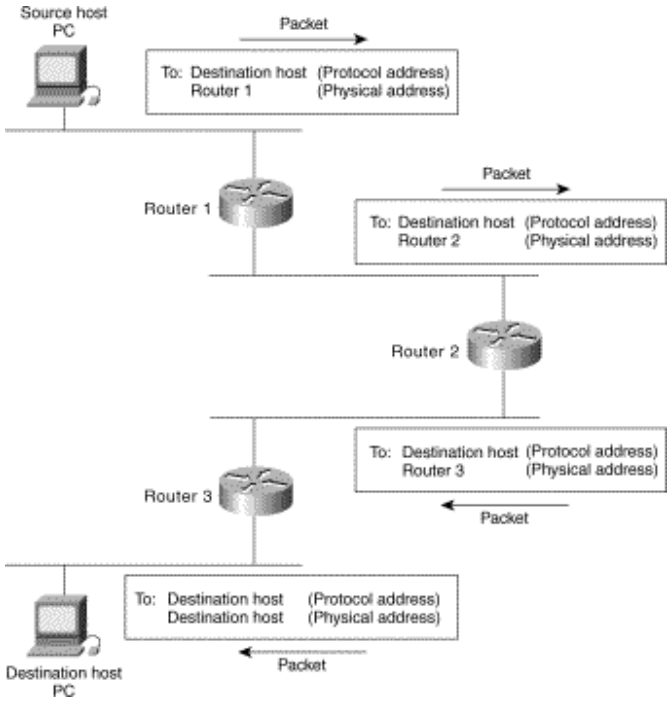
- What's inside a router (Physical routers detail)

# Topic of the lab

- Configuration of the home network

# Routing

# Routing


Source host
PC
Packet
To: Destination host (Protocol address)
Router 1 (Physical address)
Router 1
Packet
To: Destination host (Protocol address)
Router 2 (Physical address)
Router 2
To: Destination host (Protocol address)
Router 3 (Physical address)
Router 3
Packet
To: Destination host (Protocol address)
Destination host (Physical address)
Packet
Destination host
PC

# Router

- A router is a device that determines the next network point to which a packet should be forwarded toward its destination
- Allow different networks to communicate with each other
- A router creates and maintain a table of the available routes and their conditions and uses this information to determine the best route for a given packet.
- A packet will travel through a number of network points with routers before arriving at its destination.

# Router Tasks

- **Forwarding**
  - Move packet from input link to the appropriate output link
  - Purely local computation
  - Must go be very fast (executed for every packet)

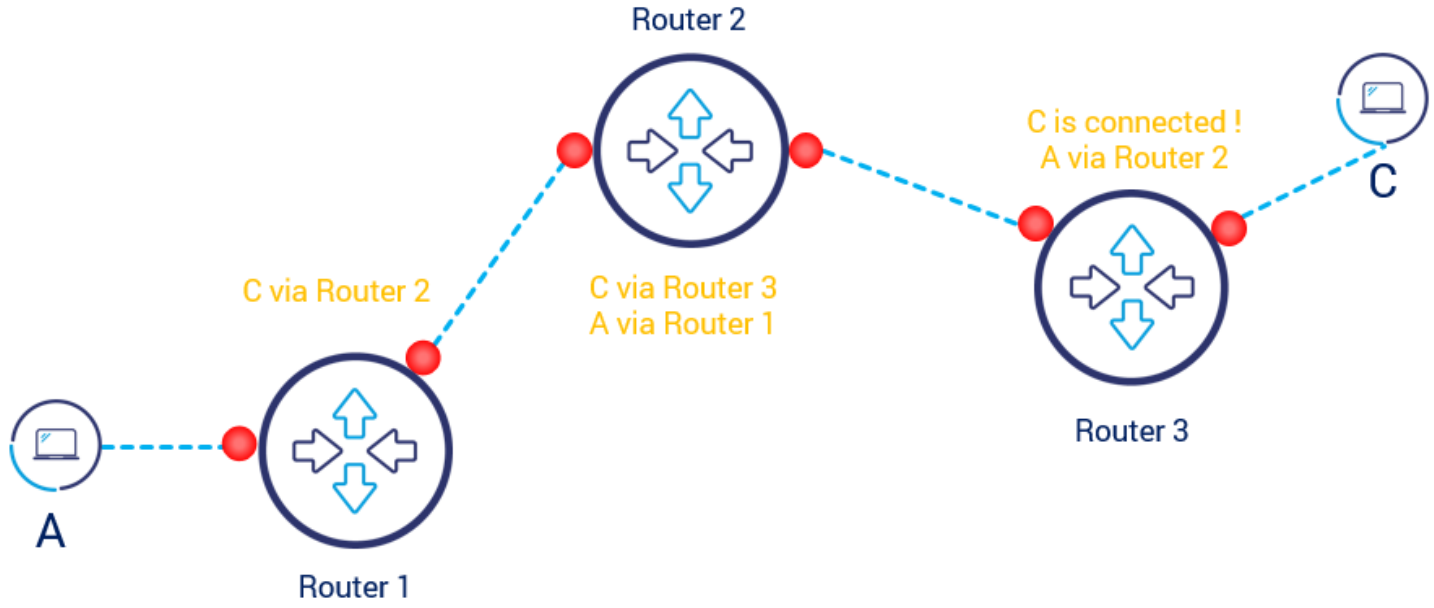
- **Routing**
  - Make sure that the next hop actually leads to the destination
  - Global decisions; distributed computation and communication
  - Can go slower

# Types of Routing

- Static Routing
- Dynamic Routing

# Static Routing

- We can tell the router explicitly how to route to destination
- For example: if want to go to C, go via Router 2.
- This is so called static routing where the destinations are written to router's configuration statically (by hand of network admin).
- On picture below, orange entries need to be provided to router by hand.



# Static Routing

- Now, imagine that you need to configure all internet destinations, over 600 000 routes!
- Would we need to provide over 600K entries by hand?
  - No way.

- There is smarter idea and is called:

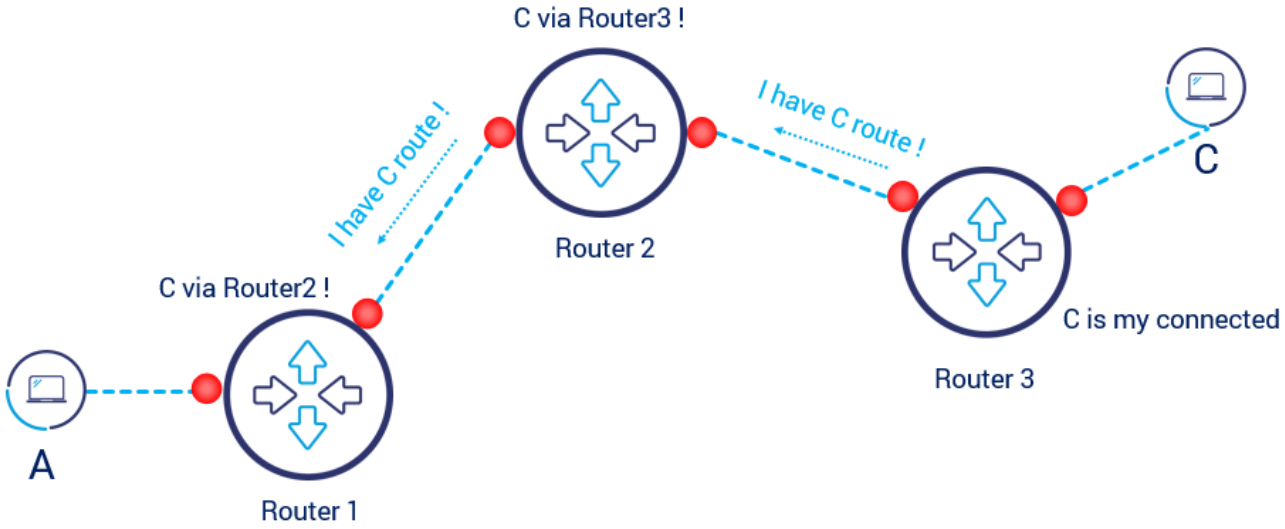
**Dynamic Routing**

# Dynamic Routing

- With dynamic routing we can configure the routers to learn the IP destinations from other routers which participate in the same routing process

# Dynamic Routing

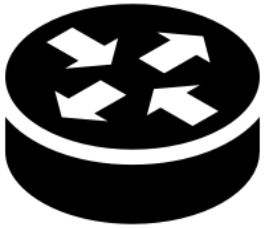
- **For Example:** Router 3 has Laptop C connected to it, so it advertises the C network address to it's neighboring routers.
- Neighboring router Router 2 receives an advertisement and injects C prefix into its routing table and in turn sends the C network information to its respective neighbors (Router 1).
- In this way without human intervention whole network is able to learn the destinations and create routing topology.

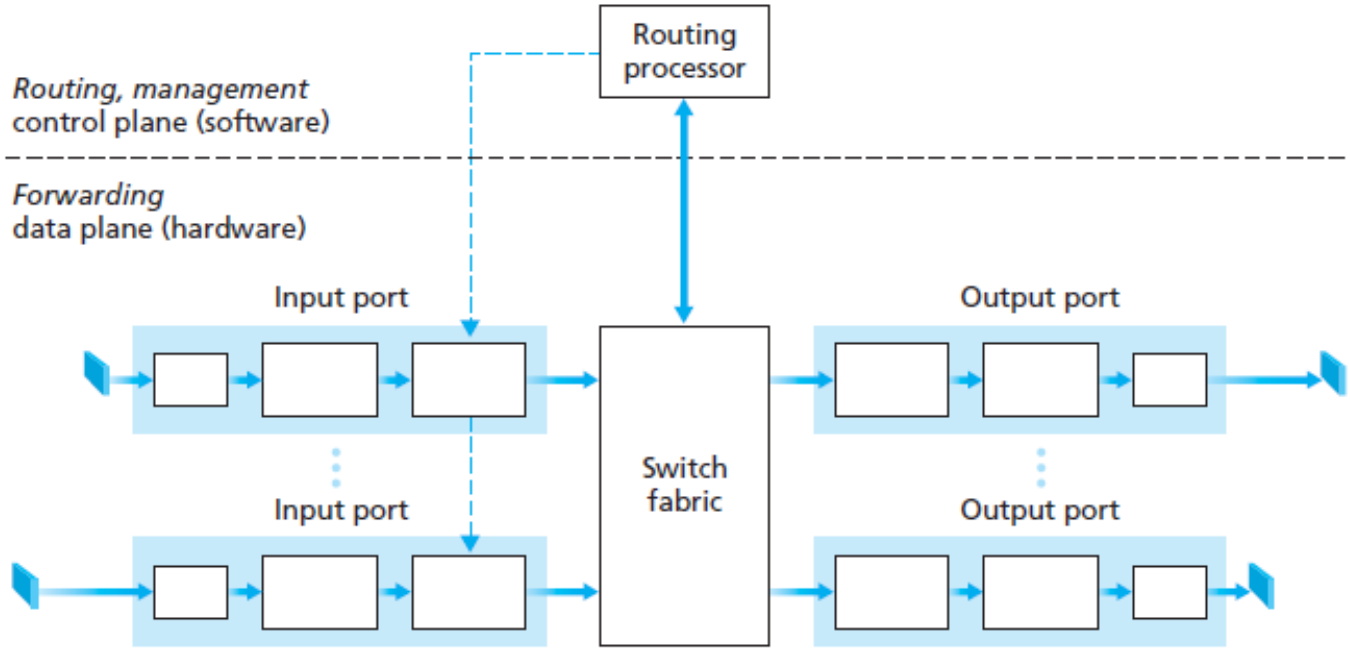


# Routing Protocols

- Static Routing
- Dynamic Routing
  - IGP (Interior Gateway Protocol): Route data within an Autonomous System
    - RIP (Routing Information Protocol)
    - RIP-2 (RIP Version 2)
    - OSPF (Open Shortest Path First)
    - IGRP (Interior Gateway Routing Protocol)
    - EIGRP (Enhanced Interior Gateway Routing Protocol)
    - IS-IS (Intermediate System to Intermediate System Protocol)
  - EGP (Exterior Gateway Protocol): Route data between Autonomous Systems
    - BGP (Border Gateway Protocol)

# Router Architecture

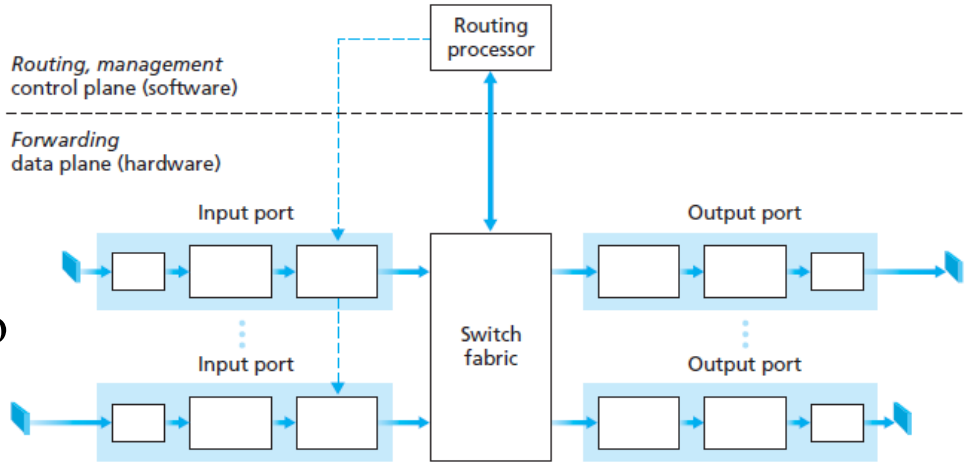





- We can have physical device or can completely implement in software

# Components of Router

- Input Port
  - Entry point for incoming packet.
- Output Port
  - Exit point of the packet.
- Switching Fabric
  - Switch the packet from I/P port to O/P port.
- Routing Processor
  - Participate in routing protocol to make forwarding table.



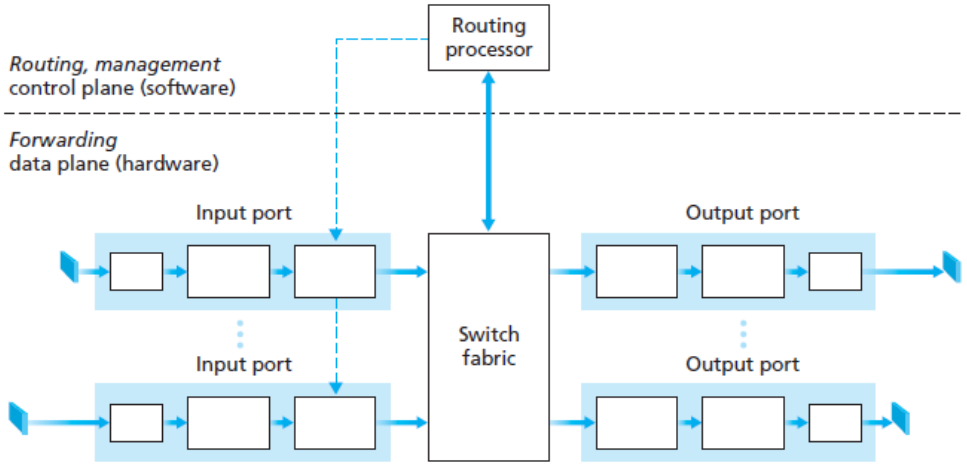
# Input Port

- An input port with an arriving packet first signalled the routing processor via an interrupt.
- The packet was then copied from the input port into processor memory.
- The routing processor then extracted the destination address from the header, looked up the appropriate output port in the forwarding table, and copied the packet to the output port's buffers.
- Classify the packet in to QoS traffic classes.

# Switching Fabric

- The switching fabric connects the router's input ports to its output ports.

- This switching fabric is completely contained within the router – a network inside of a network router!



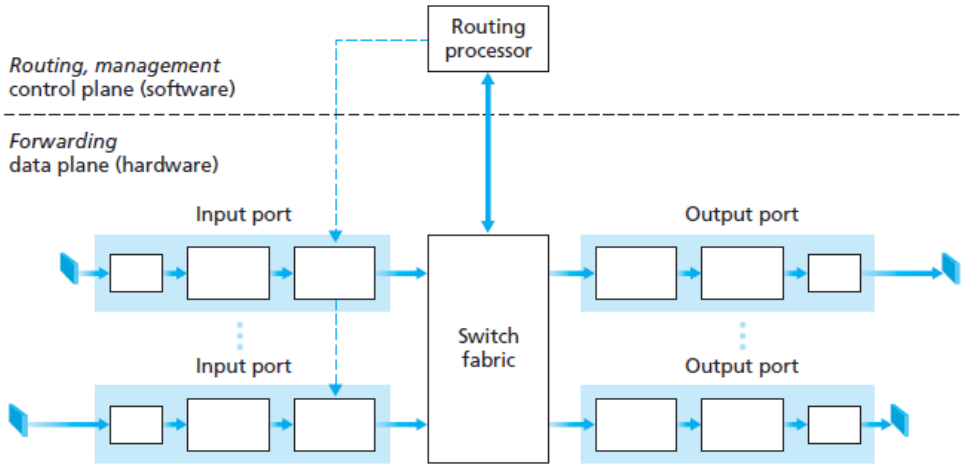
# Output Ports

- An output port stores packets received from the switching fabric.
- Packets heading for same output link need to be stored in a buffer; so as to avoid packet loss.
- It transmits these packets on the outgoing link by performing the necessary link-layer and physical-layer functions.
- Supports sophisticated scheduling algorithms to support priorities and guarantees.

# Routing Processor

- The routing processor
  - executes the routing protocols,
  - maintains routing tables and attached link state information, and
  - computes for forwarding table for the router.

- Runs the software to configure and manage the router.



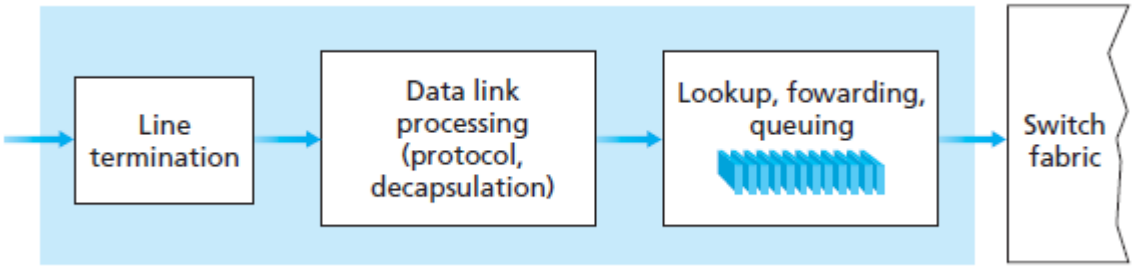
# Router Forwarding Plane

- A router's input ports, output ports, and switching fabric together implement the forwarding functions
- These forwarding functions are sometimes collectively referred to as the **router forwarding plane**.
- It can be implement as hardware or as a software
- Forwarding plane hardware can be implemented either using a router vendor's own hardware designs, or constructed using purchased merchant-silicon chips
  - For Example: sold by companies like Intel and Broadcom
- The **router control plane** functions are usually implemented in software and execute on the routing processor

# Datapath/Control functions

- Router functions can also be divided as
  - Datapath functions
    - Functions applied to every packet.
    - E.g. header lookup, forwarding, scheduling.
    - Handled by input, output port and switching fabric.
  - Control functions
    - Functions not applied to every packet.
    - E.g. system configuration, management, table update.
    - Handled by routing processor
  - Goal for high speed requires increase in the rate at which datapath functions are performed.

# Input Processing

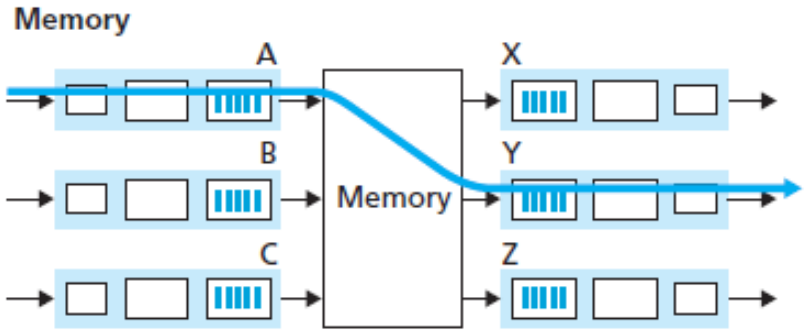


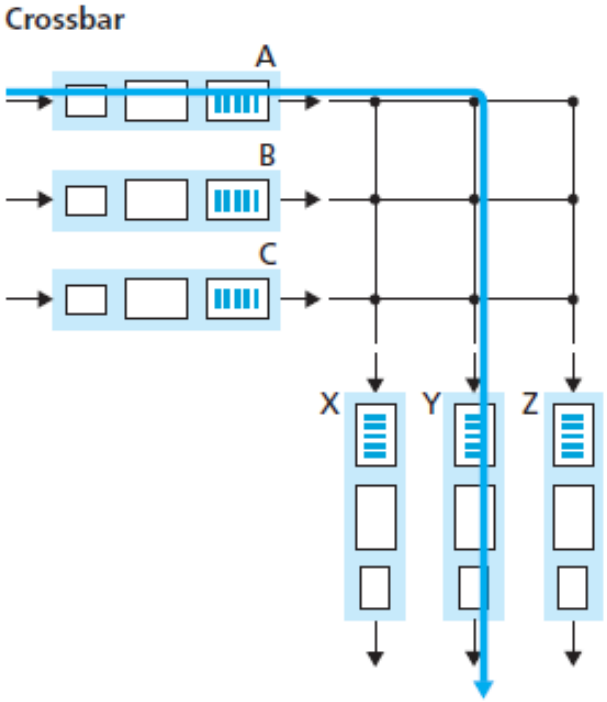

- The lookup performed in the input port is central to the router's operation
- The router uses the forwarding table to look up the output port to which an arriving packet will be forwarded via the switch fabric.
- The forwarding table is computed and updated by the routing processor, with a shadow copy typically stored at each input port.
- With a shadow copy, forwarding decision can be made locally, at each input port, without invoking the centralized routing processor on a per-packet basis and thus avoiding a centralized processing bottleneck.

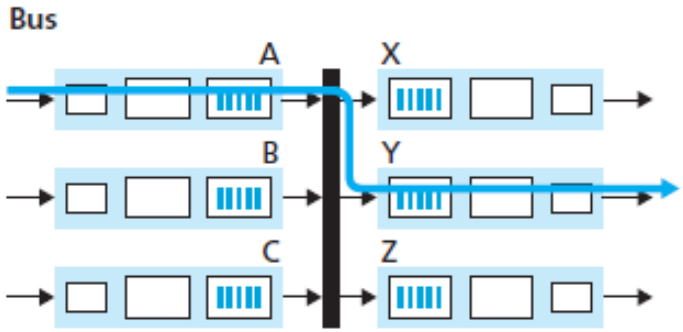
# Switching

- The switching fabric is the heart of a router
- Through this fabric the packets are actually switched (that is, forwarded) from an input port to an output port.
- Switching can be accomplished in a number of ways

# Types of Switching



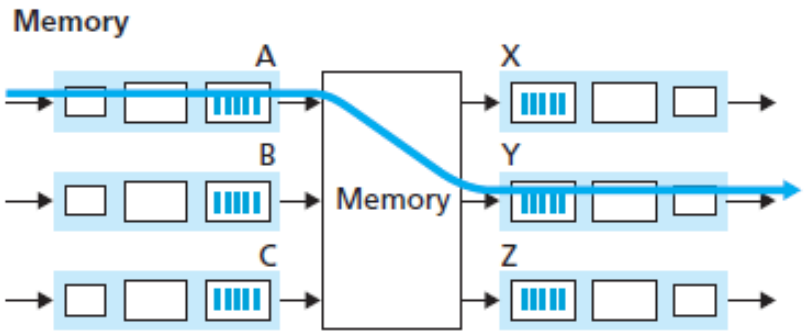



# Switching via Memory

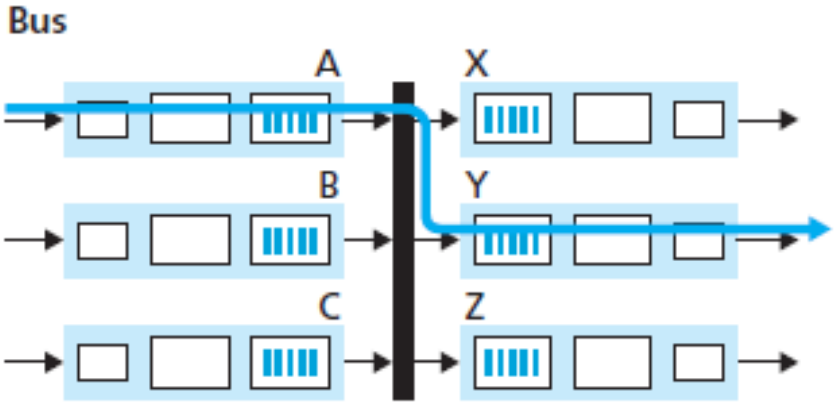
- The simplest, earliest routers were traditional computers, with switching between input and output ports being done under direct control of the CPU.

- Input and output ports functioned as traditional I/O devices in a traditional operating system.



# Switching via a Bus

- In this approach, an input port transfers a packet directly to the output port over a shared bus, without intervention by the routing processor.

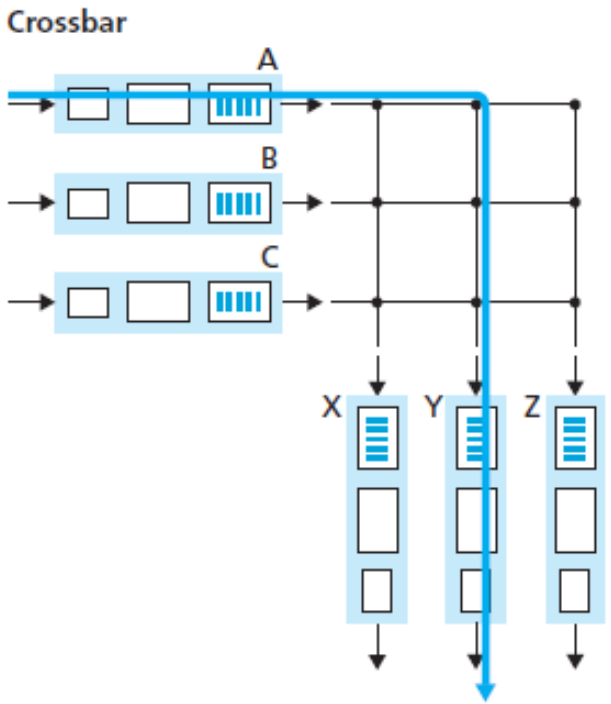


# Switching via a Bus

- The packet is received by all output ports, but only the port that matches the label will keep the packet.
- The label is then removed at the output port, as this label is only used within the switch to cross the bus.
- If multiple packets arrive to the router at the same time, each at a different input port, all but one must wait since only one packet can cross the bus at a time.
- Because every packet must cross the single bus, the switching speed of the router is limited to the bus speed

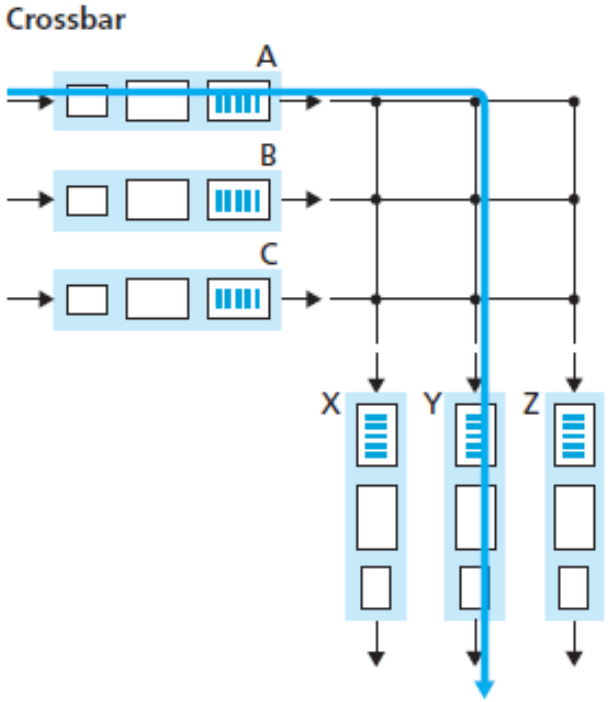
# Switching via an Interconnection Network

- One way to overcome the bandwidth limitation of a single, shared bus is to use a more sophisticated interconnection network
- A crossbar switch is an interconnection network consisting of 2N busses that connect N input ports to N output ports, as shown in the figure.
- Each vertical bus intersects each horizontal bus at a crosspoint, which can be opened or closed at any time by the switch fabric controller
- This logic is part of the switch fabric itself.



# Switching via an Interconnection Network

- When a packet arrives from port A and needs to be forwarded to port Y
- The switch controller closes the crosspoint at the intersection of buses A and Y, and port A then sends the packet onto its bus, which is picked up (only) by bus Y.
- **Note:** A packet from port B can be forwarded to port X at the same time, since the A-to-Y and B-to-X packets use different input and output busses.
- Thus, unlike the previous two switching approaches, crossbar networks are capable of forwarding multiple packets in parallel.



# Switching via an Interconnection Network

- However, if two packets from two different input ports are destined to the same output port, then one will have to wait at the input, since only one packet can be sent over any given bus at a time.

# Packet Scheduling: FCFS

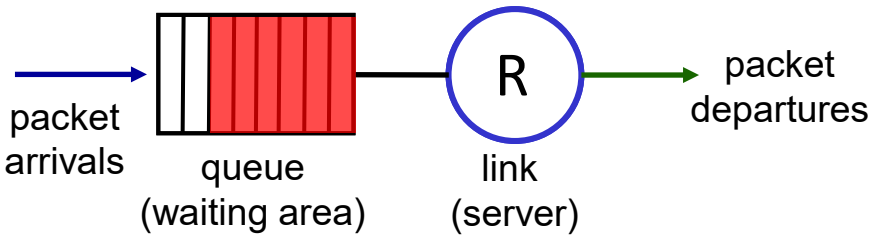
**packet scheduling**: deciding which packet to send next on link

- first come, first served
- priority
- round robin
- weighted fair queueing

**FCFS**: packets transmitted in order of arrival to output port

- also known as: First-in-first-out (FIFO)
- real world examples?

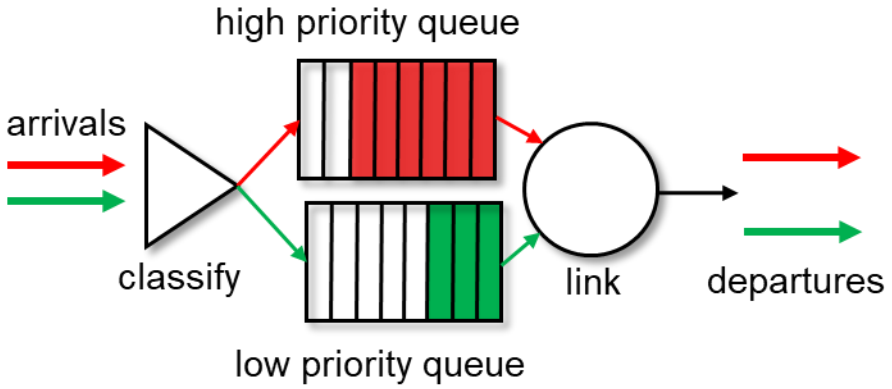
Abstraction: queue

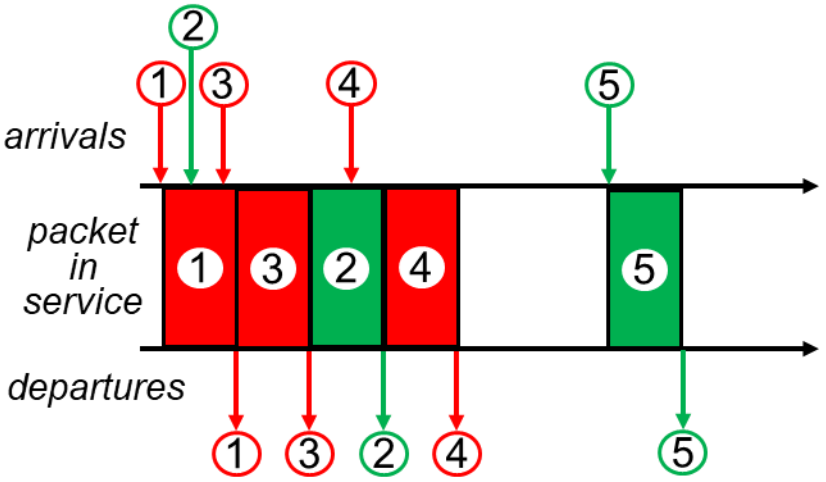


# Scheduling policies: priority

**Priority scheduling:**

- arriving traffic classified, queued by class
  - any header fields can be used for classification
- send packet from highest priority queue that has buffered packets
  - FCFS within priority class

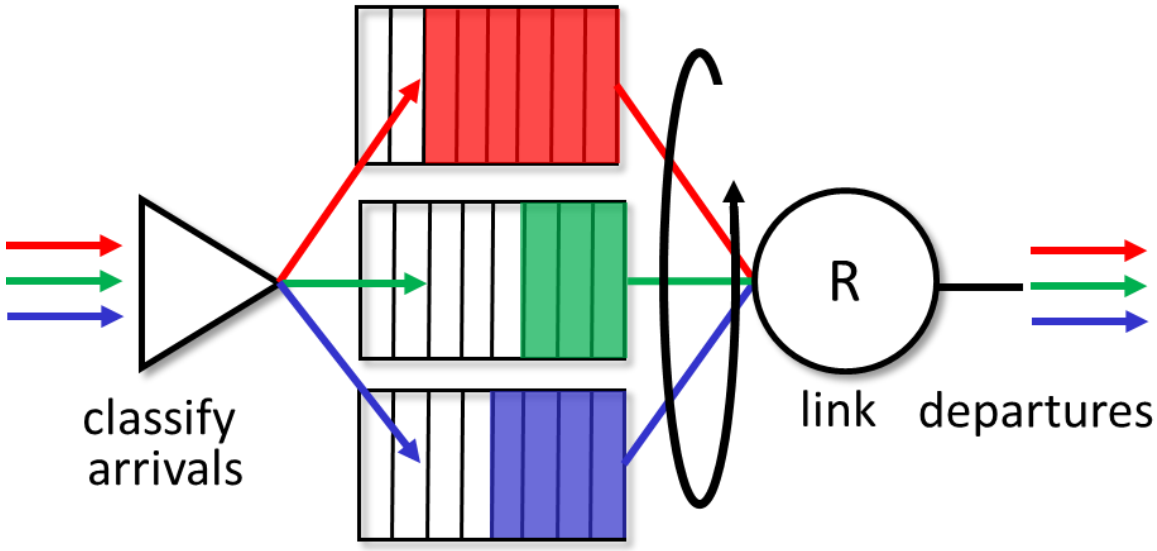


# Scheduling policies: round robin

**Round Robin (RR) scheduling:**

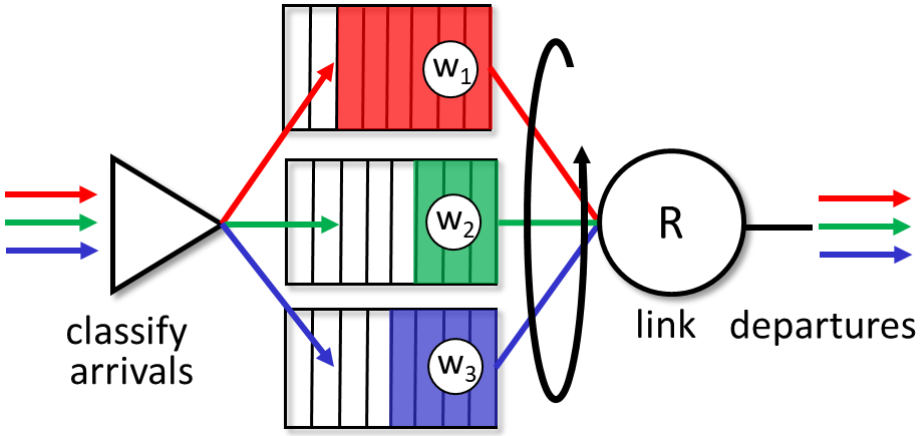
- arriving traffic classified, queued by class
  - any header fields can be used for classification
- server cyclically, repeatedly scans class queues, sending one complete packet from each class (if available) in turn



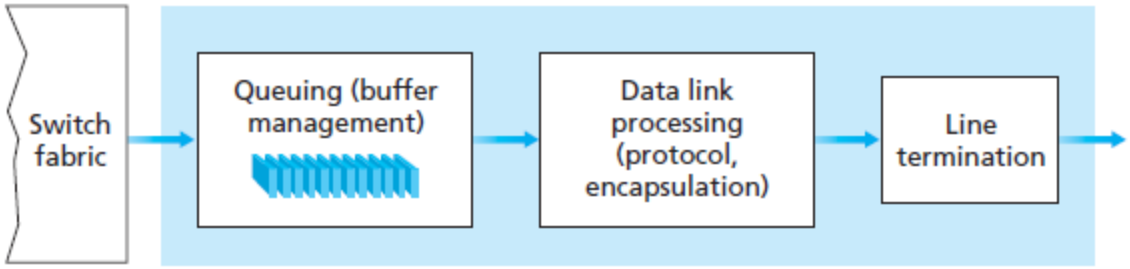
# Scheduling policies: weighted fair queueing

**Weighted Fair Queuing (WFQ):**

- generalized Round Robin
- each class, $i$, has weight, $w_i$, and gets weighted amount of service in each cycle: $$\frac{w_i}{\sum_j w_j}$$
- minimum bandwidth guarantee (per-traffic-class)



# Output Processing




- Output port processing, shown in the above figure, takes packets that have been stored in the output port's memory and transmits them over the output link.

- This includes selecting and de-queuing packets for transmission, and performing the needed link-layer and physical-layer transmission functions.

# Routing Table and Routing Algorithms

- Routing table is divided into two groups
  - Procedures used to determine the correct route for a datagram
  - Procedure used to add, change or delete routes

- The main data structure for storing the routes is a list

- Each entry in the list corresponds to a bucket
  - Contains pointers to a linked list of records for a route to destination
  - Each record on the list contains
    - Destination IP address
    - Subnet mask
    - Next-hop address
    - Network interface to use for sending the next-hop address

# How does Routing Table work?

- Routers have interfaces which connect with other network devices
- When a packet comes at an interface, router needs to find the right interface to send out the packet
- The destination IP field of packet is checked against information stored in the router
- If there is a match, the packet is sent out via the proper interface
- This process is repeated along the way on each router

# Trends: Route Lookup

- Speed of algorithm determined by:
  - Number of memory accesses to match one address.
  - Speed of memory.

- Traditional algorithm is to store routes in a tree; every path in the tree from root to leaf corresponds to an entry in forwarding table.

# Trends: Route Lookup

- Techniques to improve the speed
  - Hardware oriented techniques.
    - CAM: Content Addressable memory.
    - Increase memory to store entries. (stanford)
    - Intelligent memory. (harvard)
  - Table compaction technique.
    - Use better data structure for forwarding table. (sweden)
  - Hashing
    - Binary search on hash table. (Washington University in St. Louis)

# Trends: Output Port

- Speed up output queue
  - Increase the speed at which the queue can be accessed.
    - Use very wide memory.
    - Integrate port controller with queue on a single chip.

- Queuing
  - FCFS - cannot offer different QoS.
  - Fair Queuing - “weighted” service.

# Trends: Cost of port

- Cost of port depends upon
  - Amount and kind of memory
  - Processing power
    - Backbone routers use Processors (general/network processor). Extensible functionality.
    - Enterprise routers use ASICs. Lack of flexibility.
  - Communication between routing processor and the port.

# Acknowledgements


- Most part of this tutorial was prepared by M.Fahim, G.Succi, and A.Tormasov

# Reference

- This tutorial is based on the on the following resources as well as relevant material over the internet.
- https://electronicspost.com/router-architecture/