# Computer Network I

## Reti di Calcolatori I

Università di Napoli Federico II – Scuola Politecnica e delle Scienze di Base
Corso di Laurea in Informatica


Riccardo Caccavale

(riccardo.caccavale@unina.it)

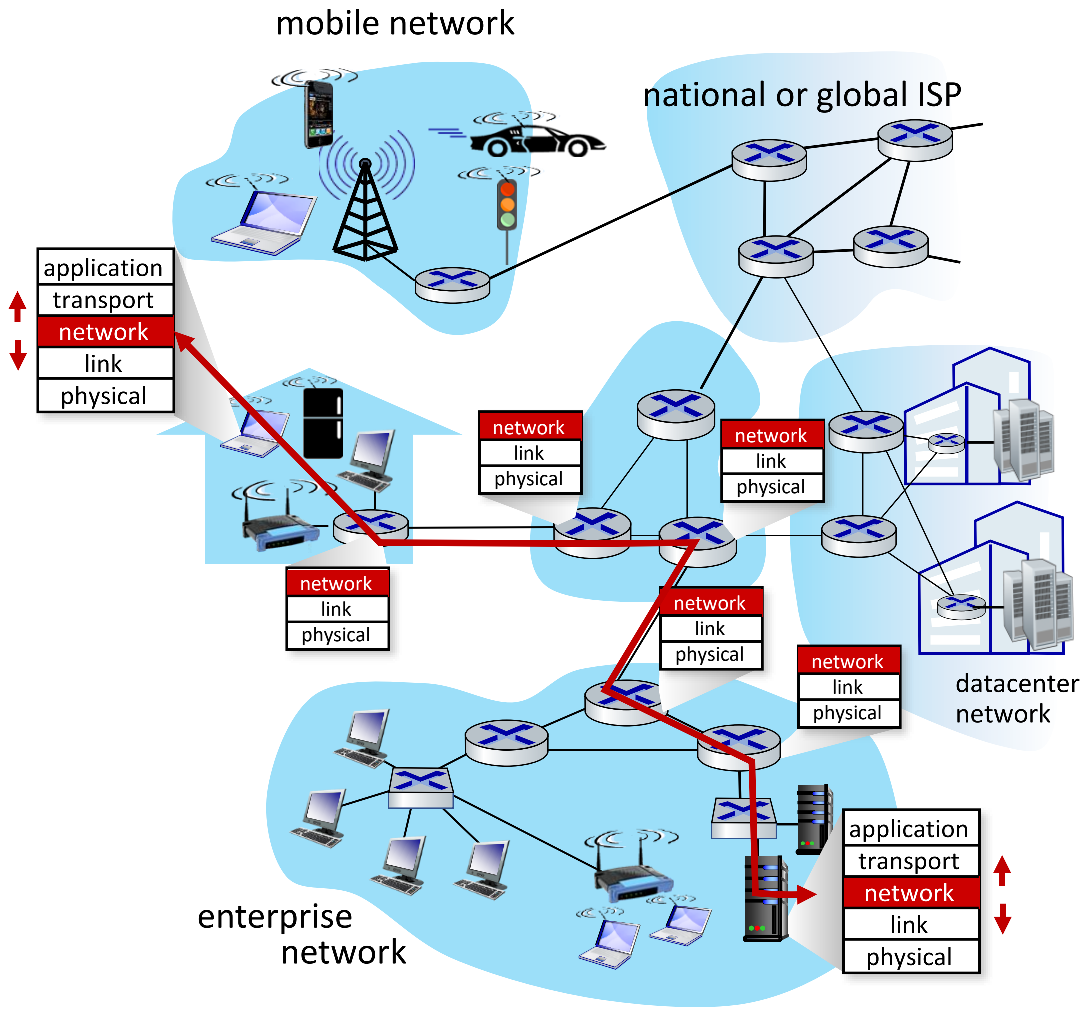

Università degli Studi di Napoli Federico II

# Network Layer

## From Transport to Network

- The job of the network layer:
  - At the **sending host**: to take segments from the transport layer, **to encapsulate each segment into a datagram**, and to send the datagrams into the network.
  - At the **receiving host**: to receive the datagrams from the network, **extracts the transport-layer segments from datagrams**, and delivers the segments up to the transport layer.

- Inside the network there are **nodes that forward datagrams to the adjacent nodes** (routers) up to the destination host.
  - Note: the routers are shown with a **truncated protocol stack**. Potentially routers do not run applications nor transport-layer protocols.



# Network Layer

## Network Layer in Internet

- The network layer is mainly responsible for **host-to-host delivery**.

- What **services could be offered** along with delivery:
  - *Guaranteed delivery*: a packet sent by a source host will eventually arrive at the destination host.
  - *Guaranteed delivery with bounded delay*: packets will be delivered and within a specified host-to-host delay bound (for example, within 100 msec).
  - *In-order packet delivery*: packets will arrive at the destination in the order that they were sent.
  - *Guaranteed minimal bandwidth*: possibility to specify a minimal bit rate (for example, 1 Mbps) such that, if the rate of the sending host is within it, then all packets are eventually delivered to the destination host.
  - *Security*: encryption/decryption of all datagrams at the source/destination.
- Actually, **none of these services** is offered by networks.

# Network Layer

## Network Layer in Internet

- Conversely, Internet's network layer provides **just one service**, the so called **best-effort service**.

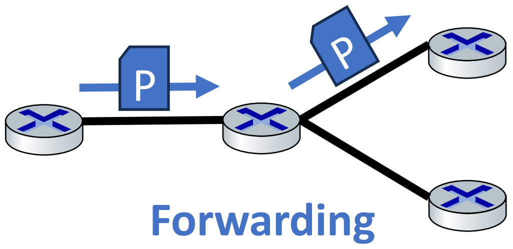
- **Best-effort service** (or best-effort delivery): the network **tries its best** to deliver a packet from its source to its destination.
  - Packets are neither guaranteed to be received in the order in which they were sent, nor to be received at all.
  - There is no guarantee about delays or minimal bandwidth.

- Despite its simplicity, the best-effort service model, in combination with good bandwidth **have proven to be adequate**.
  - For example, applications, such as Netflix and voice-and-video-over-IP, real-time conferencing, Skype, etc. all works with it.

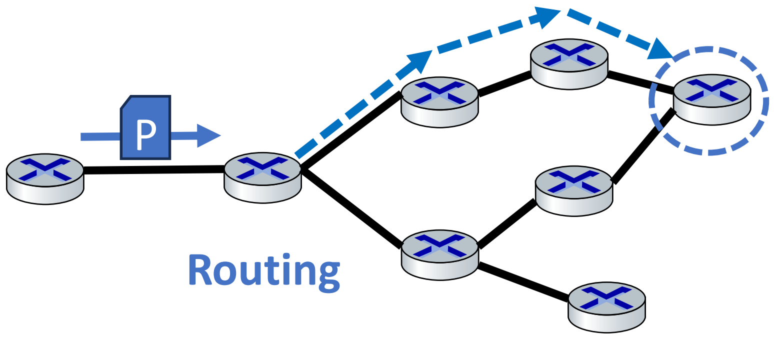
# Network Layer

## Routers in the Network

- The primary role of the network layer **is to perform host-to-host delivery**, i.e., to move packets from a sending host to a receiving host.
- This process is performed by **routers** (network nodes) that are **special nodes having multiple incoming/outcoming links**. Routers provide two network-layer functions:
  - **Forwarding**: when a packet arrives at a router's input link, the router **must move the packet to the appropriate output** link. It is also possible to:
    a. **Block a packet** from exiting a router (e.g., if the packet originated at a known malicious sending host, or if the packet were destined to a forbidden destination host).
    b. **Duplicate a packet** and sent it over multiple outgoing links.




  - **Routing**: to **decide the route or path to be taken** by packets as they flow from a sender to a receiver. The algorithms that calculate these paths are referred to as **routing algorithms**.

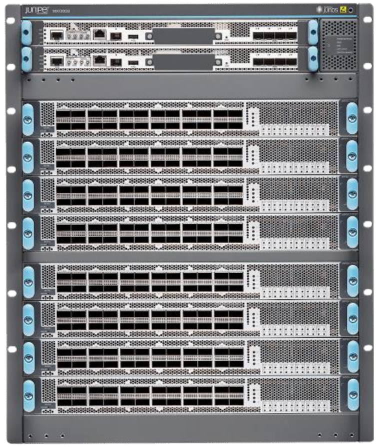


# Network Layer
## Routers

- Routers may be very **different** depending on their functions:
  - Home or business usage.
  - Wireless or wired connections.
  - **Edge routers**: a router that distributes data packets between one or more networks (e.g., connecting a network with the ISP).
  - **Core routers**: used to distribute packets within the same network rather than across multiple networks.
    - These are also used **on the backbone of the Internet** and its job is to carry out heavy data transfers.

Cisco RV016-G5
business router

Tp-link AX6600
home router



Juniper MX2020,
edge router

Cisco CRS-1
backbone core router

# Network Layer

## Routers

- **Forwarding** (data plane) refers to the **router-local** action of transferring a packet from an input link interface to the appropriate output link interface.
    - This is a **fast operation** (typically a few nanoseconds), and thus is typically implemented in hardware.

- **Routing** (control plane) refers to the **network-wide** process that determines the end-to-end paths that packets take from source to destination.
    - This is a **slower process** (typically seconds), and it is often implemented via software.

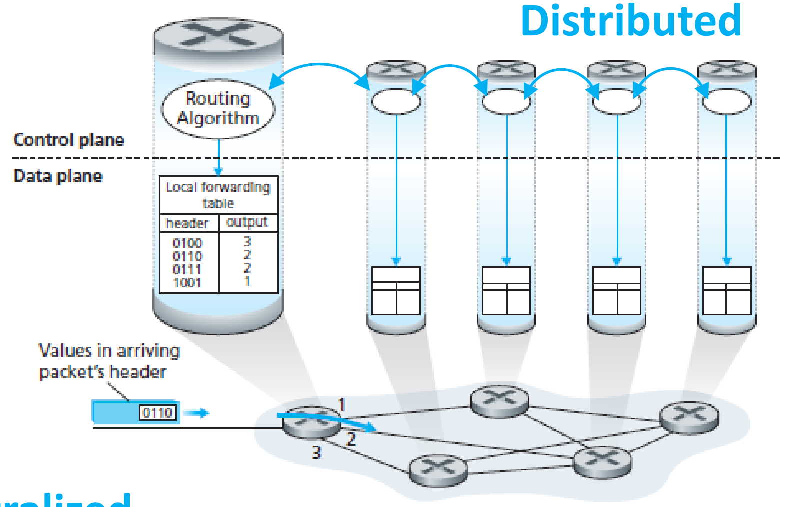
- At the data plane we have the **forwarding table**: a table that specifies to which output link a packet should be forwarded in order to reach the destination.
    - A router forwards a packet by **examining the value of one or more fields** in the header of arriving packets.
    - The value stored in the forwarding table entry indicates **the outgoing link interface** to which that packet is to be forwarded.

# Network Layer
## Routers

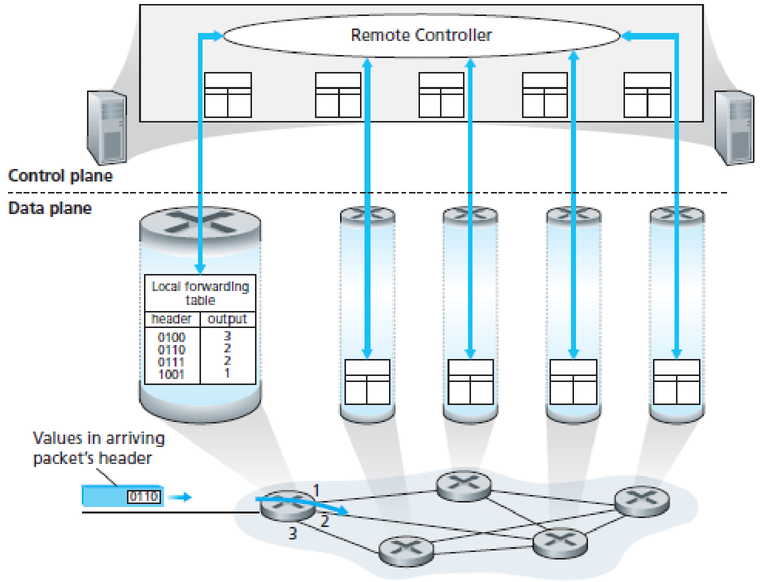
- Since multiple routers can be encountered before to reach a destination, the **content of a forwarding table** must be determined **by collecting information from different routers**.

- This functionality can be performed in two ways:
    - **Decentralized** (or distributed): we can have each router endowed with a routing component that communicates with the routing component of other routers (this was the mainstream approach).
    - **Centralized**: we can have a physically separated (from the routers) remote controller computes that distributes the forwarding tables to be used by routers.

Distributed




Centralized



# Network Layer

## Routers

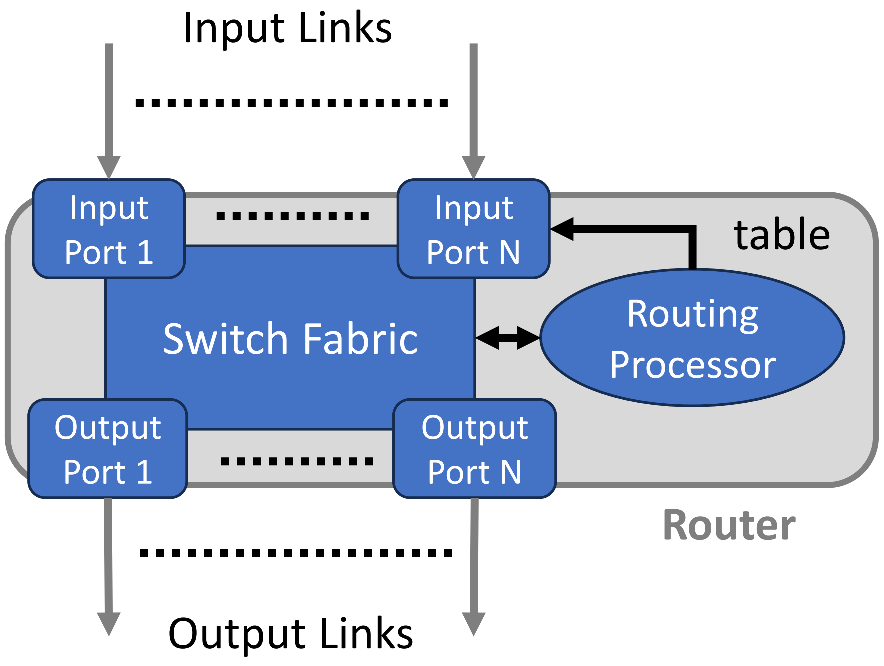
- In the second case, the remote controller might be implemented in a **remote data center** (with high reliability and redundancy) and might be managed by the ISP or some third party.

- This approach is at the basis of **software-defined networking** (SDN), where the network is “software-defined” because the controller that computes forwarding tables and interacts with routers is implemented as a software.
  - Some of these software are also open.

# Network Layer

## Routers: Main Components

- Main router's **components** are:
  - **Input ports** (different from transport-layer ports): are the **physical input interfaces** that operate with the lower link-layer of the connected link, their job is:
    - To consult the forwarding table (aka **lookup**) and **to prepare the switching factory** for the output port to choose.
    - To **forward control packets** (e.g., packets carrying routing information) to the routing processor.
    - The number of input **ports may range from dozens to hundreds** (e.g., the Juniper MX2020 edge router supports up to 960 10 Gbps Ethernet ports).
  - **Switching fabric**: connects input ports to its output ports.
  - **Output ports**: stores packets received from the switching fabric and transmits these packets on the outgoing link.
    - **Ports are often bidirectional**, i.e., output port is coupled with an input port.
  - **Routing processor**: that executes the routing protocols, maintains state-information about links, and computes the forwarding table.



# Network Layer

## Routers: Forwarding

- The **role of the forwarding table is to associate IP addresses to output ports**, so that packets can be forwarded to the right output link in order to be transmitted to the next node (and possibly be forwarded again).

- An **IP address is a 4-bytes (32-bits) number** that we usually see in decimal notation, but we can also see it in its binary form:

**192.168.1.1**

**11000000 10101000 00000001 00000001**

- Inside a forwarding table, **IPs are associated to output port numbers** for actual redirection.

# Network Layer

## Routers: Forwarding

- When a packet is received from the link into an input port **a lookup operation** is performed.

- The **port has a copy of the forwarding table** from the routing processor (received on a dedicated bus, e.g., a PCI bus) to avoid bottleneck due to continuous invoking of the centralized routing processor on a per-packet basis.

Example of a 4-ports router's table

| Prefix of IP addresses | Link Interface |
|---|---|
| 11001000 00010111 00010*** ******** | 0 |
| 11001000 00010111 00011000 ******** | 1 |
| 11001000 00010111 00011*** ******** | 2 |
| otherwise | 3 |

11001000 00010111 00010000 00000000
TO
11001000 00010111 00010111 11111111

11001000 00010111 00011000 00000000
TO
11001000 00010111 00011000 11111111

11001000 00010111 00011001 00000000
TO
11001000 00010111 00011111 11111111

# Network Layer

## Routers: Forwarding

Example of a 4-ports router's table

| Prefix of IP addresses | Link Interface |
|---|---|
| 11001000 00010111 00010*** ******** | 0 |
| 11001000 00010111 00011000 ******** | 1 |
| 11001000 00010111 00011*** ******** | 2 |
| otherwise | 3 |

11001000 00010111 00010000 00000000
TO
11001000 00010111 00010111 11111111

11001000 00010111 00011000 00000000
TO
11001000 00010111 00011000 11111111

11001001 00010111 00011001 00000000
TO
11001000 00010111 00011111 11111111

- Notice that entries may not be mutually exclusive (for example, IP 11001000 00010111 00011000 10101010 matches links 1 and 2) if so, the router forwards it to the longest matching entry (**longest prefix matching rule**):

**Match with Link 1 = 24 bits**

11001000 00010111 00011000 10101010 → **Link 1 wins!**

**Match with Link 2 = 21 bits**

# Network Layer

## Routers: Forwarding

- This **lookup procedure is typically performed on the hardware** to be executed as fast as possible (e.g., in nanoseconds for Gigabit transmissions).
  - Embedded memories and advanced table-search algorithms are also involved.

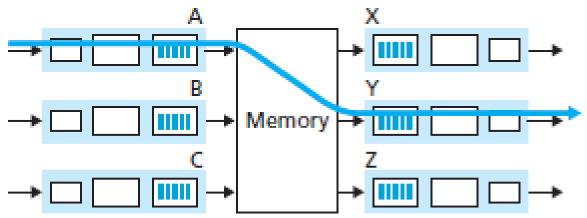
- Once a packet's output port has been determined (after lookup), **the packet can be sent into the switching fabric**. In some devices, packets may also be temporarily queued (buffered) if other input ports are using the fabric.

- This two-steps operation of **looking up a destination IP address (match)** and **forwarding (action)** is called **match-plus-action** and is performed in many networked devices, such as:
  - **Switches**: similar action as routers.
  - **Firewalls**: where **the action is to filter out specific incoming packets**.
  - **Network address translators** (NATs): where the action is **to rewrite port number** of specific incoming packets before forwarding.

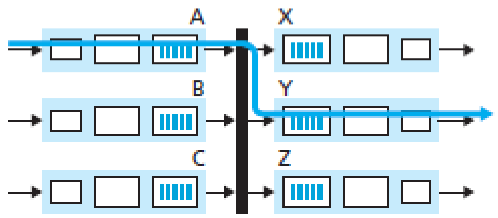
# Network Layer
## Routers: Switching Fabric

- In a switching fabric, switching can be performed in 3 ways:
  1. **Switching via memory**: ports are **considered as I/O devices** that write packets on memory cells then the **routing process copies the message** on the output port as specified by the forwarding table.
     - This approach is a bit slow (need for memory access) and was more common in early routers (which were standard computers).
  2. **Switching via a bus**: input port transfers a packet directly to the output port **over a shared bus**, without intervention by the routing processor.
     - Only one port per time can be served, but this method is often sufficient for routers that operate in small local area.
  3. **Switching via an interconnection network**: input/output ports **connected by a network** having cross-points which can be open/close and then redirect the packets.
     - Here multiple packets can be forwarded in parallel. This method is used in several modern routers.

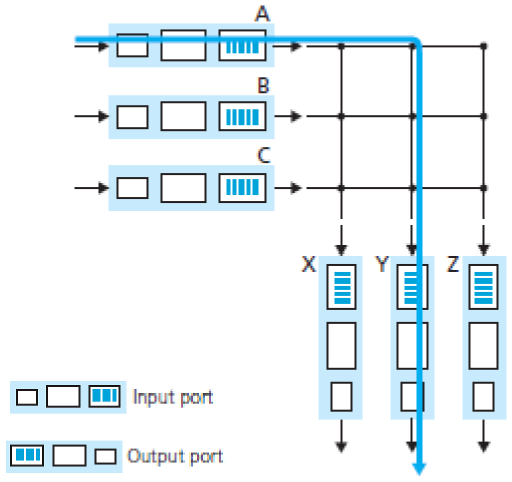
Memory




Bus



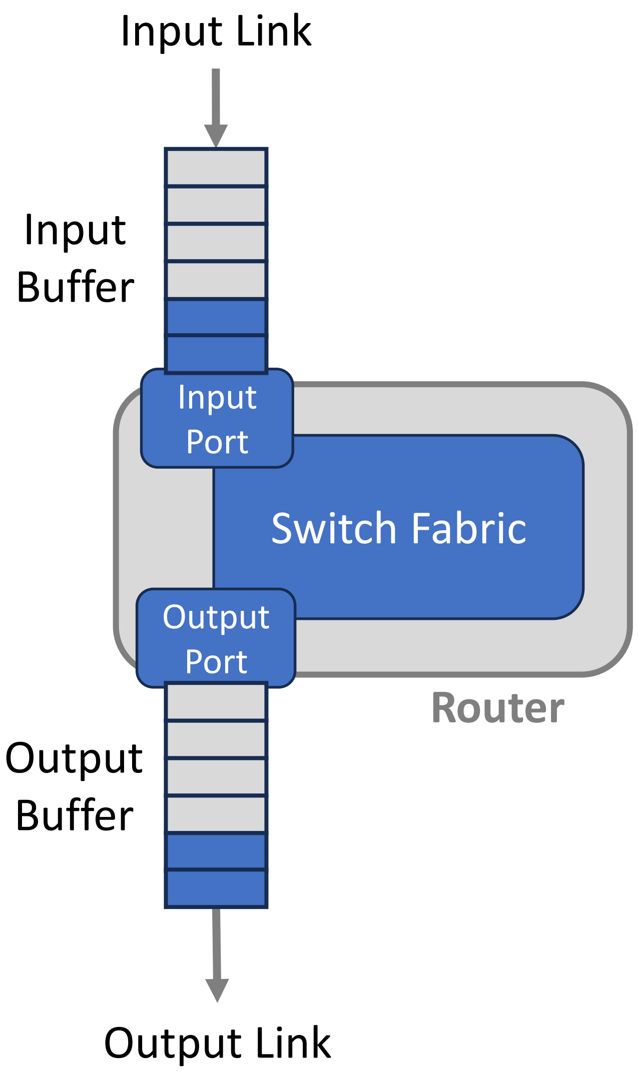

Network (crossbar)



# Network Layer

## Routers: Ports

- Since **switching takes time**, input and output **ports have queues to temporarily store packets** (as cars waiting for semaphores).
- The **extent of queueing is not fixed**, it may depend on traffic load, the speed of the switching fabric, or the line speed.
- In general, **packets may be received/sent faster/slower than switching**, so they may be accumulating into the input/output buffer (that may overflow).



# Network Layer

## Routers: Ports

- Example: let's assume to have (1) **a router with N input and N output ports**, (2) that **each port is receiving packets** at the same time, and (3) that all input and output lines **have the same speed** of $R_{line}$ packets per second.
- Let's now consider a **worst-case scenario** in which **all packets have to be forwarded to the same port**.

- If the **switching fabric** have a rate ($R_{switch}$) we have:
  - If $R_{switch} \cong NR_{line}$, **queuing on input ports is negligible** as all packets are forwarded in time by the forward fabric, but **queuing on the output port is significant** as the incoming packets are N time more than the $R_{line}$ rate of the output link.
  - if $R_{line} < R_{switch} < NR_{line}$, there is **queuing on both ports** as input port must wait for the switch fabric while output port must wait for the link.
  - if $R_{switch} \cong R_{line}$, there is **negligible queueing on output** port but **significant queuing on the input** ports.

# Network Layer

## Routers: Packet Scheduling

- It is reasonable to have **multiple packets** (potentially from multiple input ports) to be forwarded **to a single output port**.

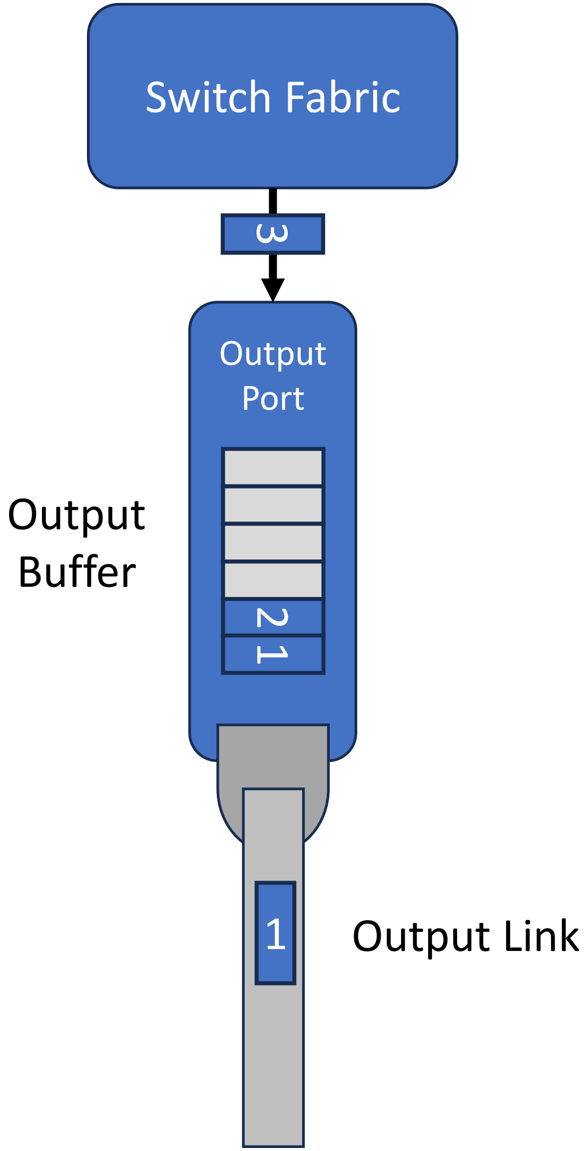
- The access of queued packets from buffer to the output link needs to be **scheduled**.

- There are basically 3 (famous) approaches:
  - **First-come-first-served** (FCFS, aka **first-in-first-out**, FIFO), which is **simple time-based approach**.
  - **Priority queuing**, which is based on the importance of the packets.
  - **Round-robin queueing**, where are divided into classes (based on priority) and each class is served in turn.

# Network Layer

## Routers: Packet Scheduling - FIFO

- If the **output link is busy** (transmitting something) the packets arriving at the output port must be **buffered**.
- If **there is no sufficient buffering space** to hold the arriving packet, we must rely on a **packet-discarding policy**.
  - A typical policy is to **drop the recently arrived packets (drop-tail)** but in more sophisticated approaches **also already buffered packets can be removed** to make space for the arriving ones.
- A packet is removed from the queue **only if it has been completely transmitted** over the outgoing link (served).
- In **FIFO scheduling** packets are selected for transmission **in the same order** in which they have arrived at the output port.



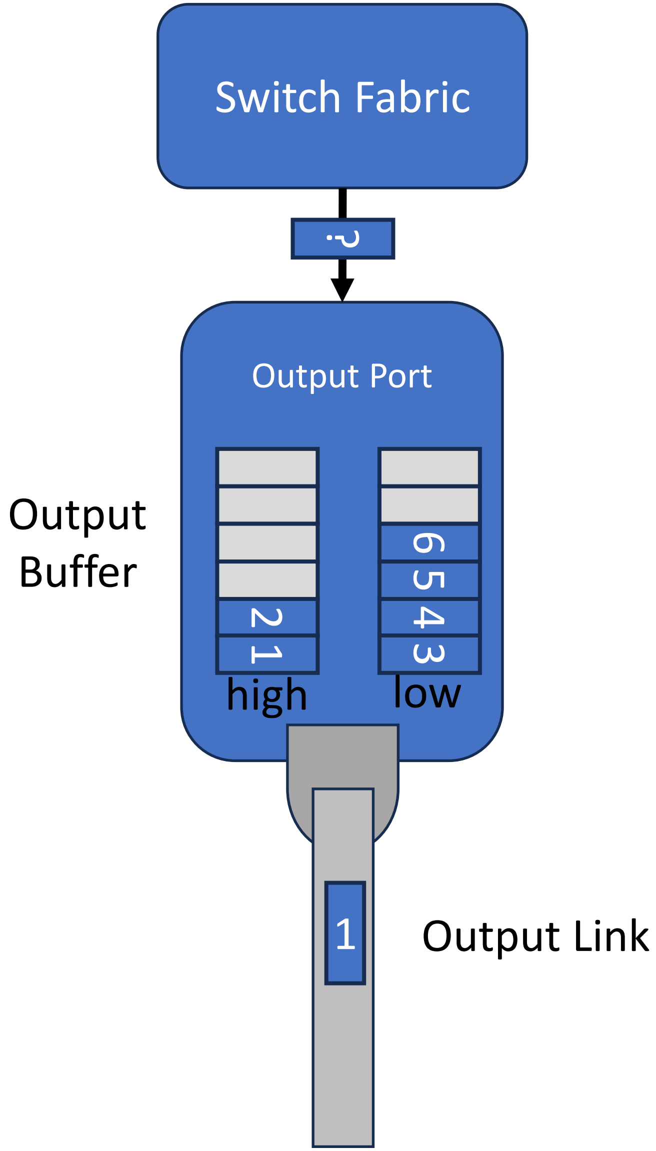
# Network Layer

## Routers: Packet Scheduling - Priority

- In **priority queuing**, packets arriving at the output link are classified (e.g., through TCP/UDP port numbers) into priority classes upon arrival at the queue.

- A **network operator may configure a queue** so that specific packets (e.g., carrying network management information, real-time voice-over-IP, etc.) may receive priority over user traffic or non-real-time packets.

- **Each priority class typically has its own queue:**
  - Packets from the most prioritized non-empty class **are transmitted first**.
  - The choice among packets in the same priority class is typically done **in a FIFO manner**.



# Network Layer

## Routers: Packet Scheduling – Round-robin

- In **round robin queuing**, packets are still sorted into classes, but **classes are alternated** rather than selected by priority.
- A common implementation is called ***weighted fair queuing*** (WFQ) where arriving packets are classified and queued in the appropriate waiting area.
- Each class is then associated to a specific weight that dictates the **rate with which the class is selected** over the others.