Introduction to Computer Networks and Internet

Architecture et Réseaux

Dino Lopez – Yves Roudier – Bénoit Miramond

Foreword

- The material presented in this course is a compilation of several Internet howto's and websites, command manual pages, API documentations, IETF RFCs, and books.
- Specially, the following slides are directly inspired from the official slides and book "Computer Networking: A Top Down Approach". Chapter 1. 6th edition. Jim Kurose, Keith Ross. Pearson.
  - Exercises are also inspired from this book
- In this course should allow you to understand the general architecture and components of a network, as well as to put in practice during the labs sessions your knowledges

Organisation générale du cours

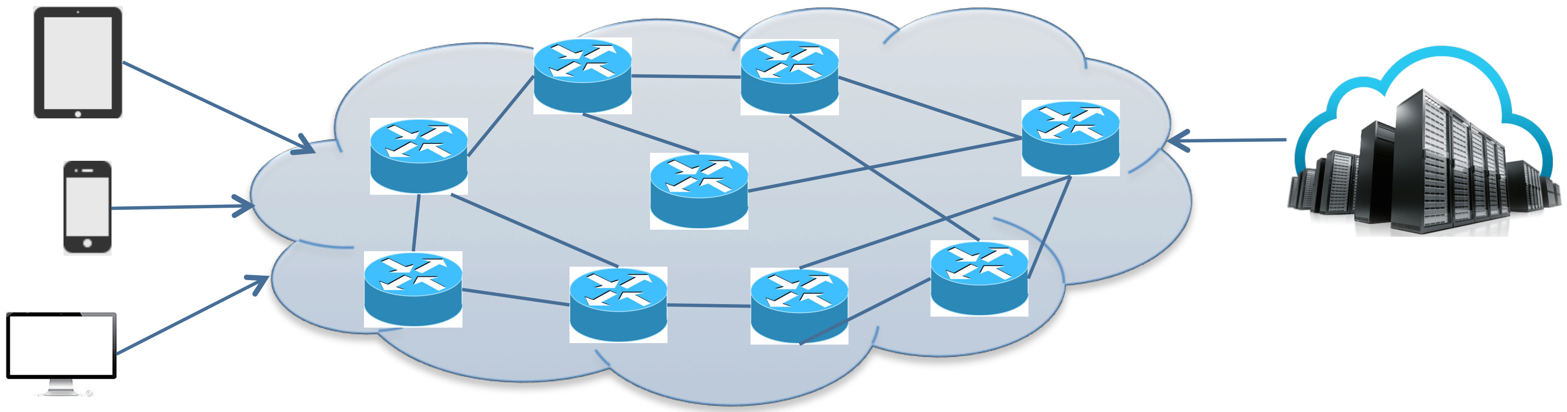
- Ce cours est divisé 2 parties
  - 1ère partie - Dino Lopez & Pr. Yves Roudier – Introduction à l'**Architecture Réseau** et Programmation Sockets
  - 2ème partie – Benoît Miramond – **Architectures de programmes**
- Pour la première partie
  - 2 encadrants des TDs pour 4 groupes
  - Un encadrant passe 1h30 par groupe
  - Les groupes sans encadrants doivent rester dans leur salle de TD et commencer ou continuer le TD de manière autonome.

Interaction Application Distribué – Infrastructure réseau

Architecture des App. Dist. : Protocoles et Standards

Le Protocole HTTP : Pierre angulaire des Services Web

Infrastructure et Protocoles Réseau
Interconnexion de Dispositifs



Evaluation

- Chaque partie est évalué d'une manière qui corresponde le mieux au contenu à présenter
- En ce qui concerne la première partie
  - Examens courts écrit (0,5h)
  - QCMs surprises
  - Comportement générale en cours et TPs -> Les TPs peuvent être notés

Java vs Python

About the programming language

- This lecture is not about a programming course.
  - However, you will have to develop small scripts to experiment with the network protocol and architecture
  - What we expect from you: Review your programming skills, mainly by yourself
- Programming will be mainly done in Python
  - API to deploy emulated networks leveraging Linux tools are written in Python
  - Much more better than executing a bunch of commands by hand
  - Programming basic Python scripts is fairly easy. Still, we expect from you to review by yourself some Python tutorials if needed

Some Python – Java equivalences - Blocs

Java

- Blocs are created with { … }

if (var1 == 4 && var2.equals(“xyz”))
{
   System.out.println(“Hello”);
} else {
   System.out.println(“Not Hello”);
}

Python

- Blocs are created with indentations
  - 4 spaces (bar space) != 1 tab 4 spaces-length

```
if (var1 == 4 and var2 ==
“xyz”):
    print “hello”
else:
    print “not hello”
```

Python – Java equivalences – declaration

Java

```
String var = “text”;

Object[] myArray = new
Object[10];
myArray[0] = 2;

Public static String
getStr ( String test) {
    return test+” test”;
}
```

Python

```
//there are no private
variables
var = “text”

myArray = array(‘i’, [0
for i in range(10)]);
myArray[0] = 2
myArray.append(10)

def getStr (test):
    return test + “ test”
```

Python – Java equivalences – Class

Java

```
public class MyClass
extends MyFirstClass {
    public MyClass (int
arg1) {
    }

    private void
metMeth () {
    }
}
```

Python

```
class MyClass (MyFirstClass):
    def __init__ (self,
arg1):
        // This is the
constructor

    def _metMeth ():
        // method
```

Python – Java equivalences – Main

Java

```
public class MyClass extends MyFirstClass {
    public MyClass (int arg1) {
    }

    public static void main (String[] args) {
        // useful tasks
    }
}
```

Python

```
if __name__ == “__main__”:
    //useful tasks
```

Python – Java equivalences – loops

Java

```
int[] values = new
int[100];

for (int i=0;
i<values.length; i++) {

System.out.println( values
[i] );
}
```

Python

```
values = array('i', [0
for i in range(10)]);

For item in values:
    print ( item )
```

Python – Java equivalences – Class

- While loops
- Logical operators
- Some mathematical built-in functions
- …

Introduction to Mininet

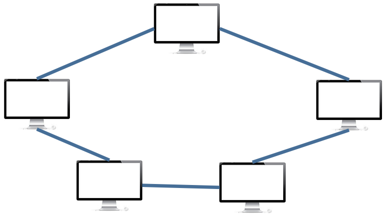
General Networking Context

Internet elements

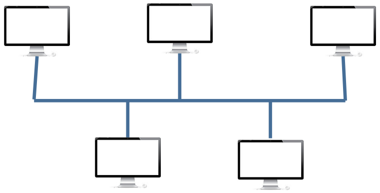
- Internet is a network which interconnects computing devices
  - Internet Service Providers
  - Network of networks
- Computing devices: hosts or end systems
  - Laptops, PC, servers
  - Today: Mobile devices, smart objects
- Communication links
  - Twisted pair, coaxial cable, radio frequency
  - ?
- Packet switches
  - Routers, link-layer switches
  - Several other devices
- Protocols
  - Standard and proprietary protocols

Device interconnection: the network topologies

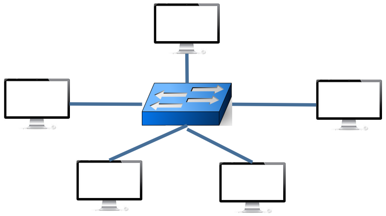
Ring



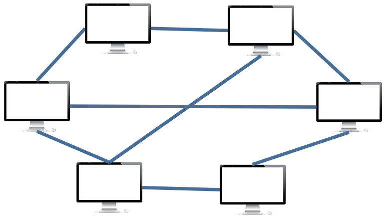
Bus



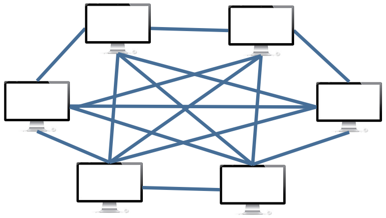
Star



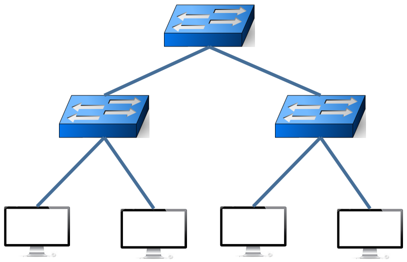
Mesh



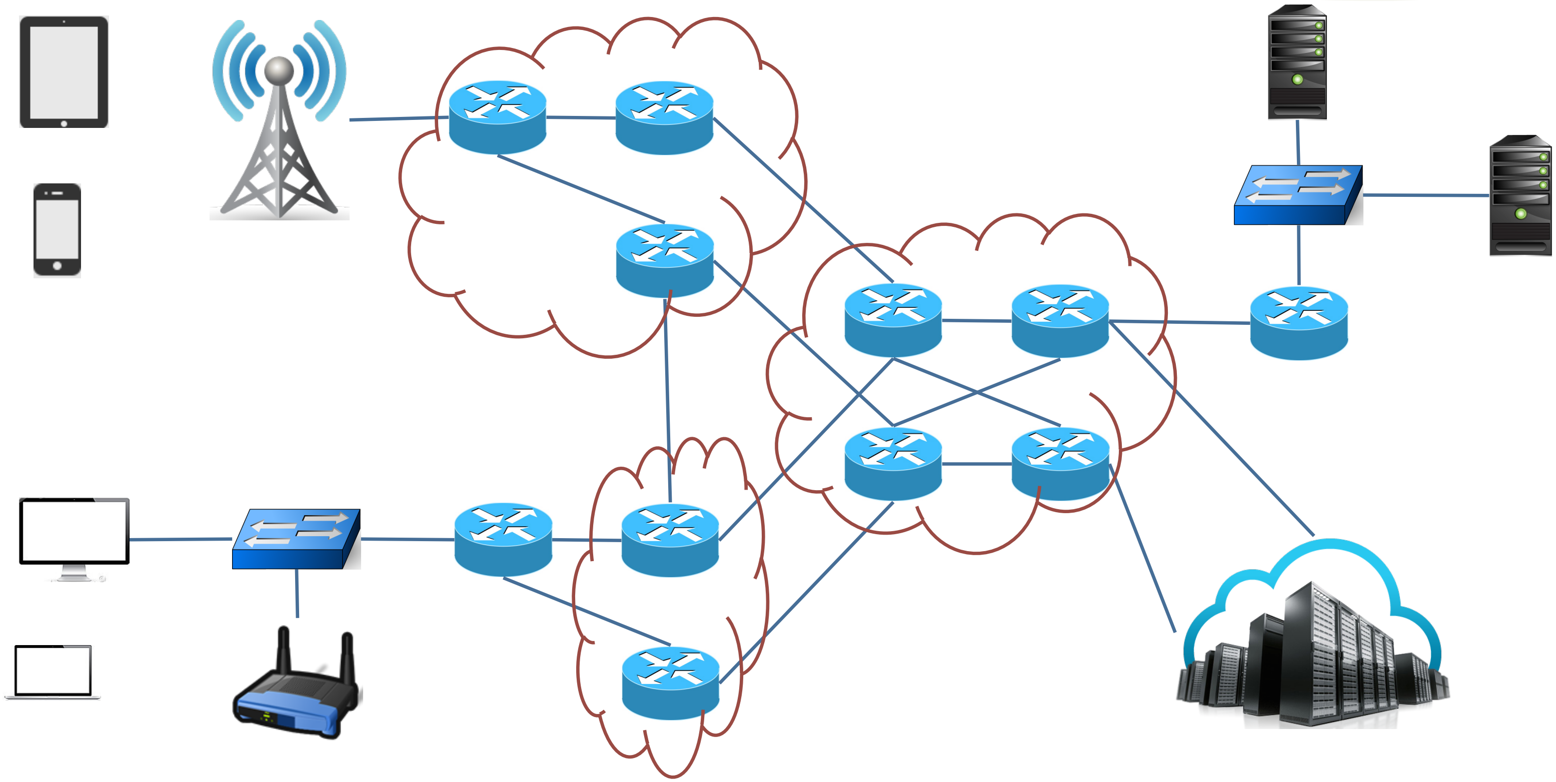
Full mesh



Tree



Global View



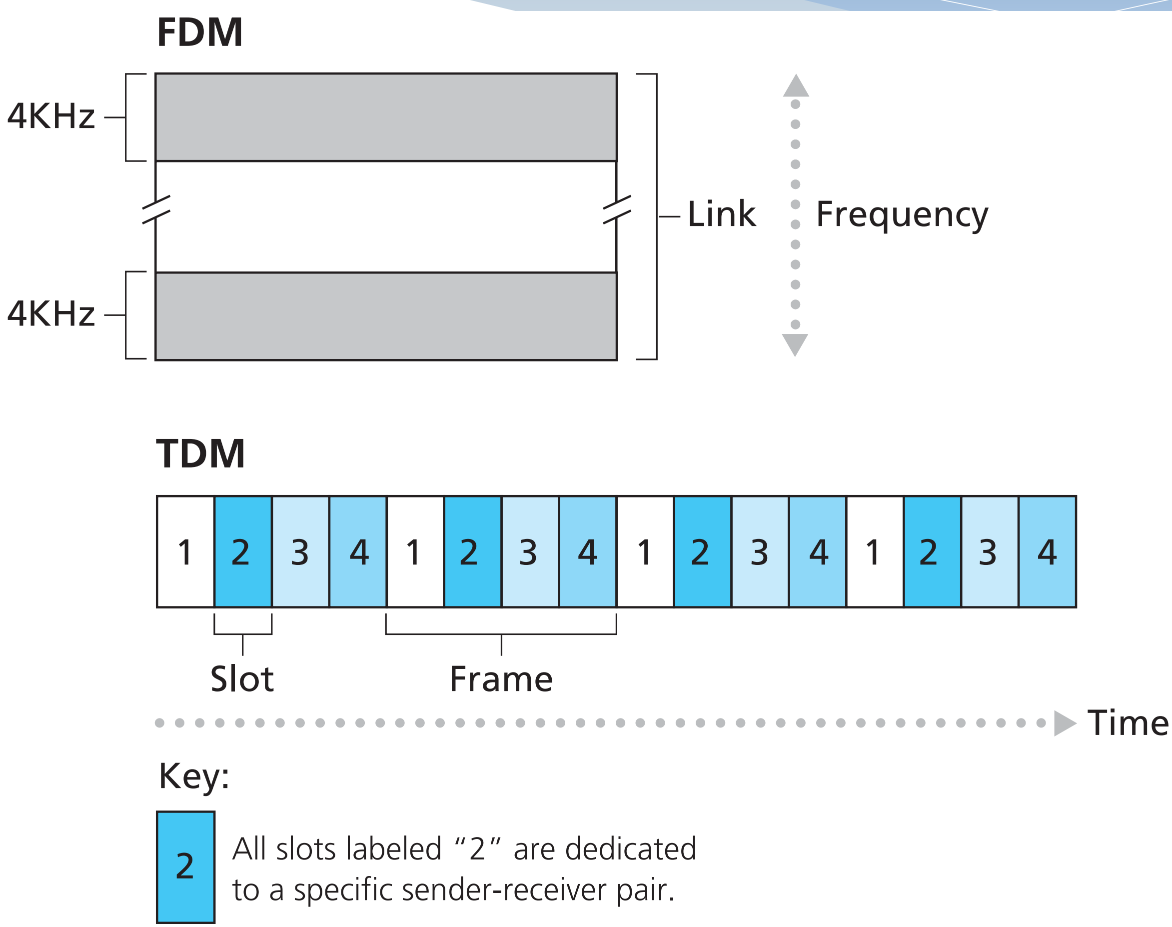
Sending packets in packet-switched networks

- Store-and-forward devices
- A device receives the bits of a packet and store the information in a buffer
- Once the entire packet has been received, the packet is forwarded
- A packet composed of *L* bits, which is transmitted through a port with output rate *R* bits/sec, will need *L/R* seconds to be transmitted = transmission delay
  - The total transmission delay of a packets going through N links (i.e. N-1 forwarding devices) will need *NL/R* seconds

Circuit switching

- In circuit switched networks, the resources are reserved for the entire duration of the communication
  - In Packet switched networks, the resources are not reserved…
  - Silent periods lead to wasted resources
- The end-to-end reserved path is known like a circuit
- A circuit can be implemented using
  - Frequency Division Multiplexing (FDM)
  - Time Division Multiplexing (TDM)

FDM vs TDM

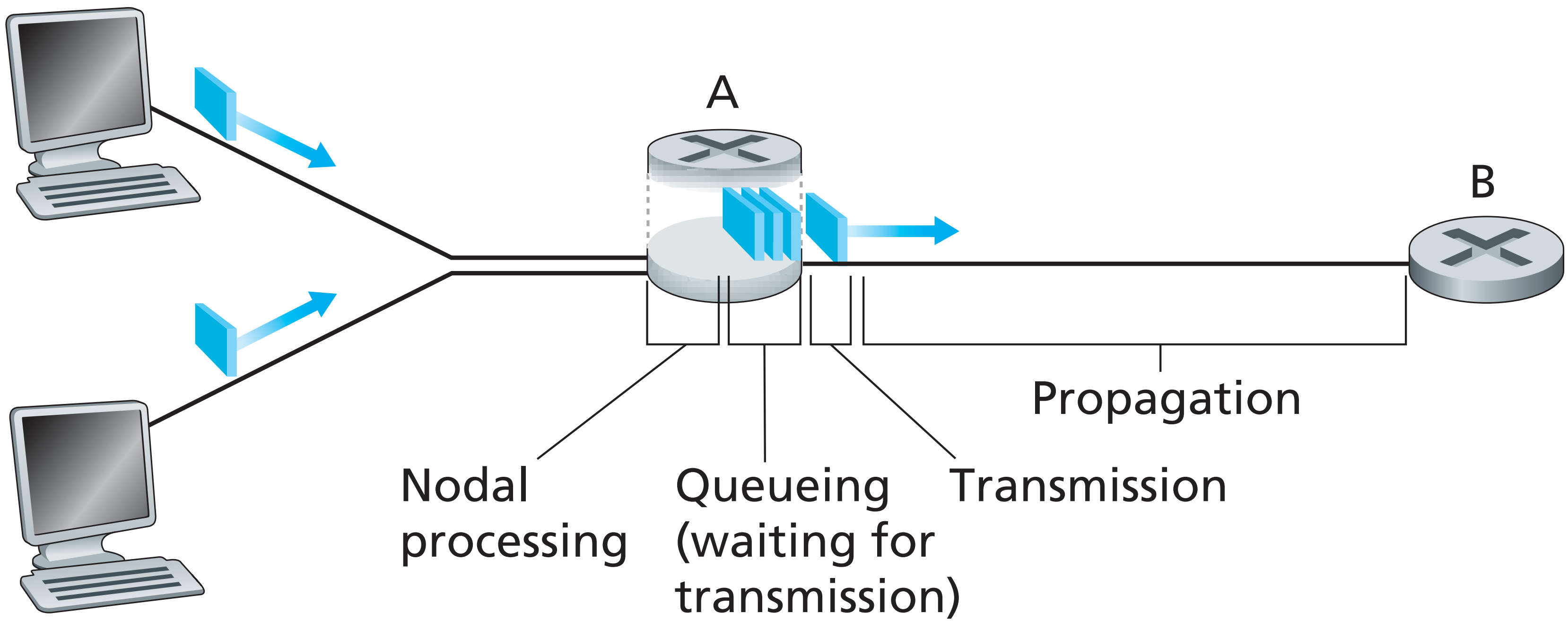


From "Computer Networking: A Top-Down Approach". PEARSON, 6th edition. James F. Kurose and Keith W. Ross

Delay in packet-switched networks

- When a packet is sent, it will travel through a series of N links and N-1 forwarding devices
- The packets suffer from several delay in a single router. The total delay seen by a packet is a node (the nodal delay) is composed of
  - Processing delay: where to direct a packet
  - Queuing delay: how long a packet waits in the queue before reach the link
  - Transmission delay: ??
  - Propagation delay: the speed at which bits propagates in a link
- The network might not be symmetric
  - The path followed by a packet from Host A to Host B can be different from the path between Host B and Host A
  - In a single forwarding device, two different ports might have different instantaneous queue size
  - Forward delay can be different to the backward delay

Delay in packet-switched networks



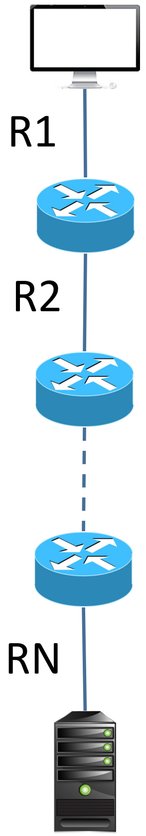
From “Computer Networking: A Top-Down Approach”. PEARSON, 6th edition. James F. Kurose and Keith W. Ross

Queuing delay

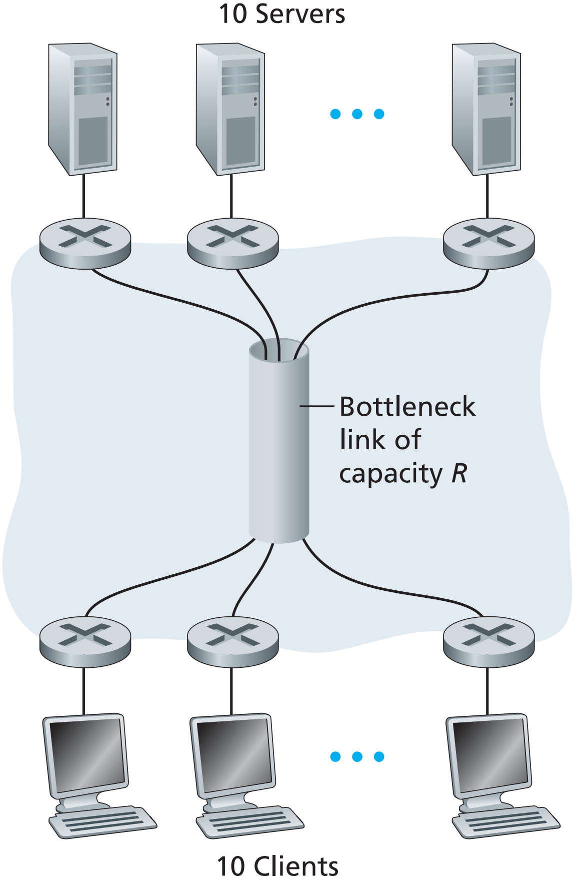
- The traffic intensity is defined as the ratio *aL/R,* where *a* is the average arrival rate of packets (pkts/s)
- The number of packets stored in a buffer will grow when the incoming arrival rate exceeds the output link capacity
  - aL/R > 1
- What about aL/R <= 1? Queuing delay depends on the nature of the packets arrival
  - If the minimum inter-arrival packet period is L/R packets, then, there is no queue
  - Packet burst (all packets arrive at the same time) will lead to queuing delay
- When a buffer becomes full, new incoming packets cannot be stored and they are dropped (leading to the so-called packet losses)

End-to-End throughput

E2E average throughput = min(R1,R2,…,RN)

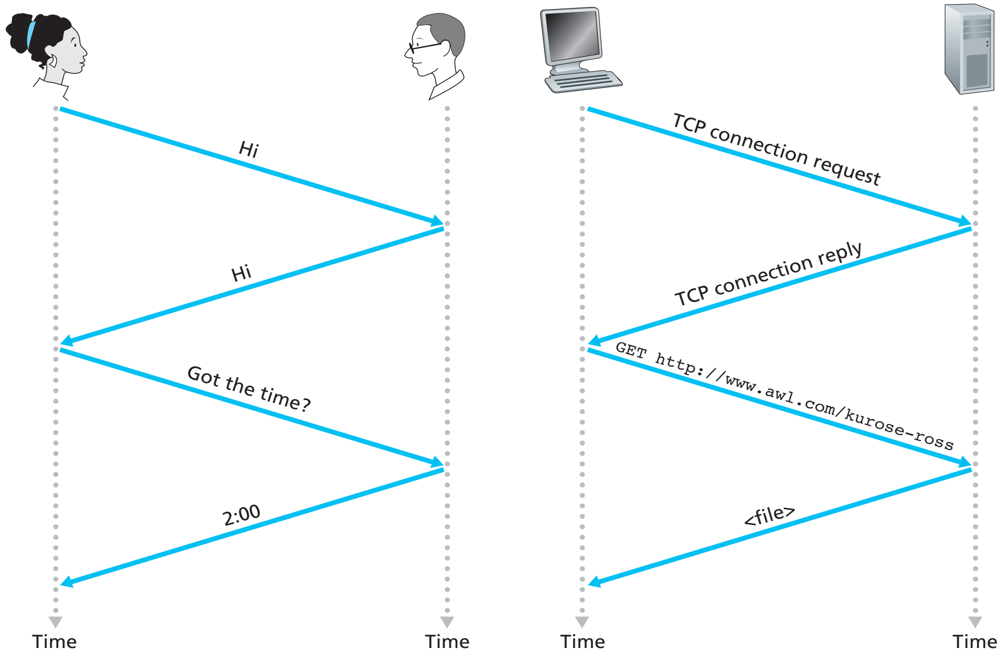


E2E average throughput = ??



Protocols

- Devices in Internet need (standardized) protocols to communicate



The Layering approach

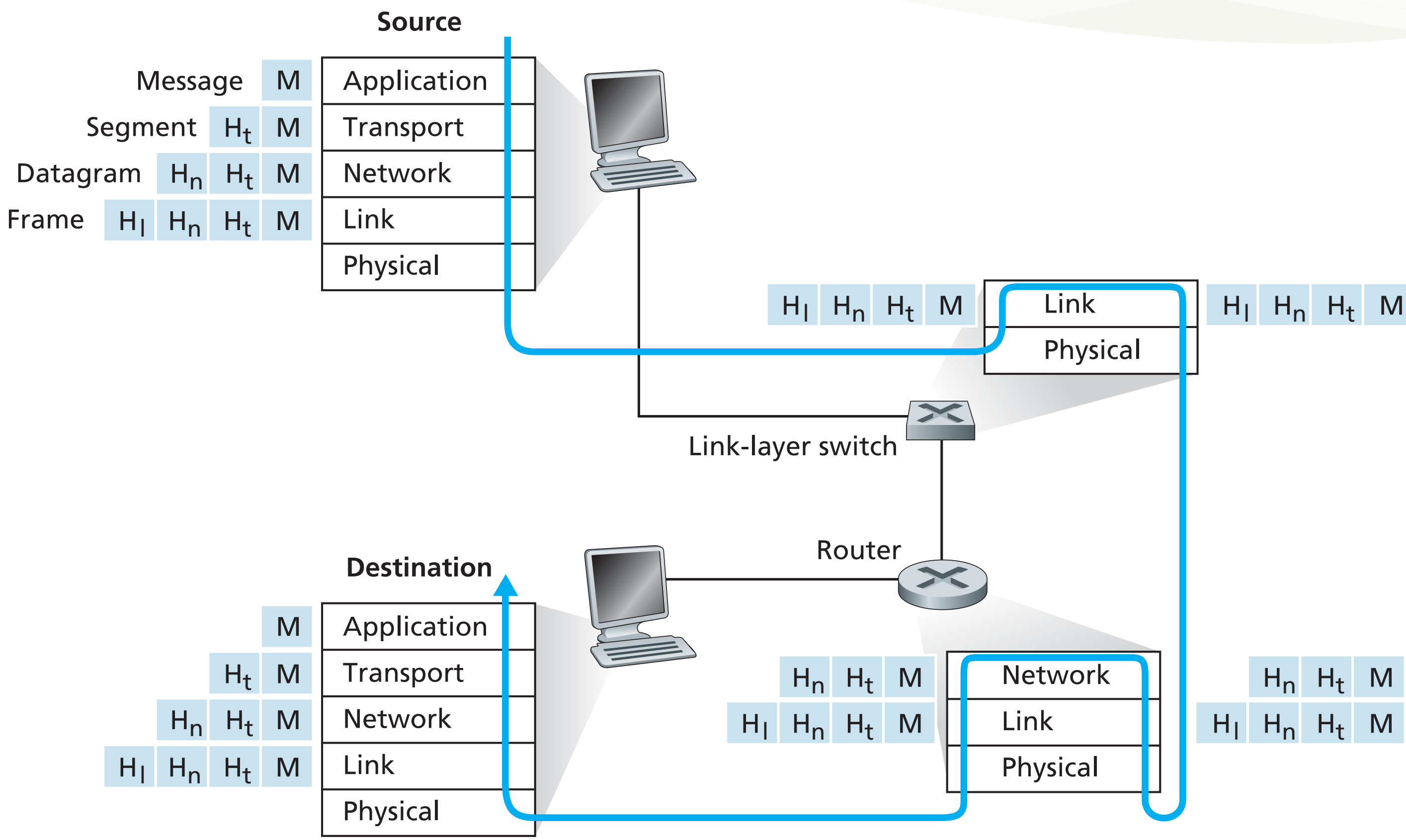
The OSI model

Application
Presentation
Session
Transport
Network
Link
Physical

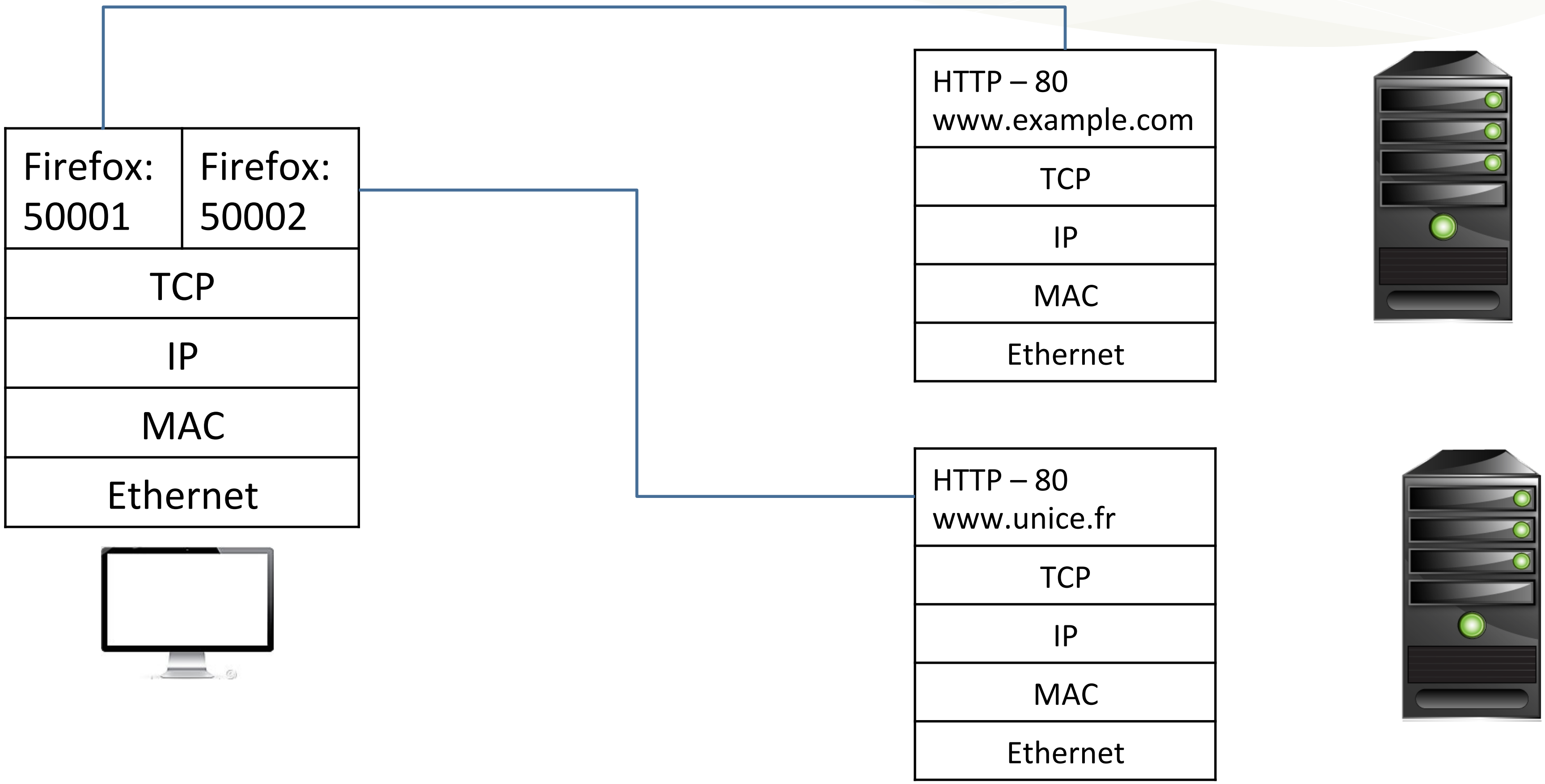
The Internet protocol stack

Application	HTTP
Transport	TCP,UDP
Network	IP
Link	ARP
Physical	Ethernet

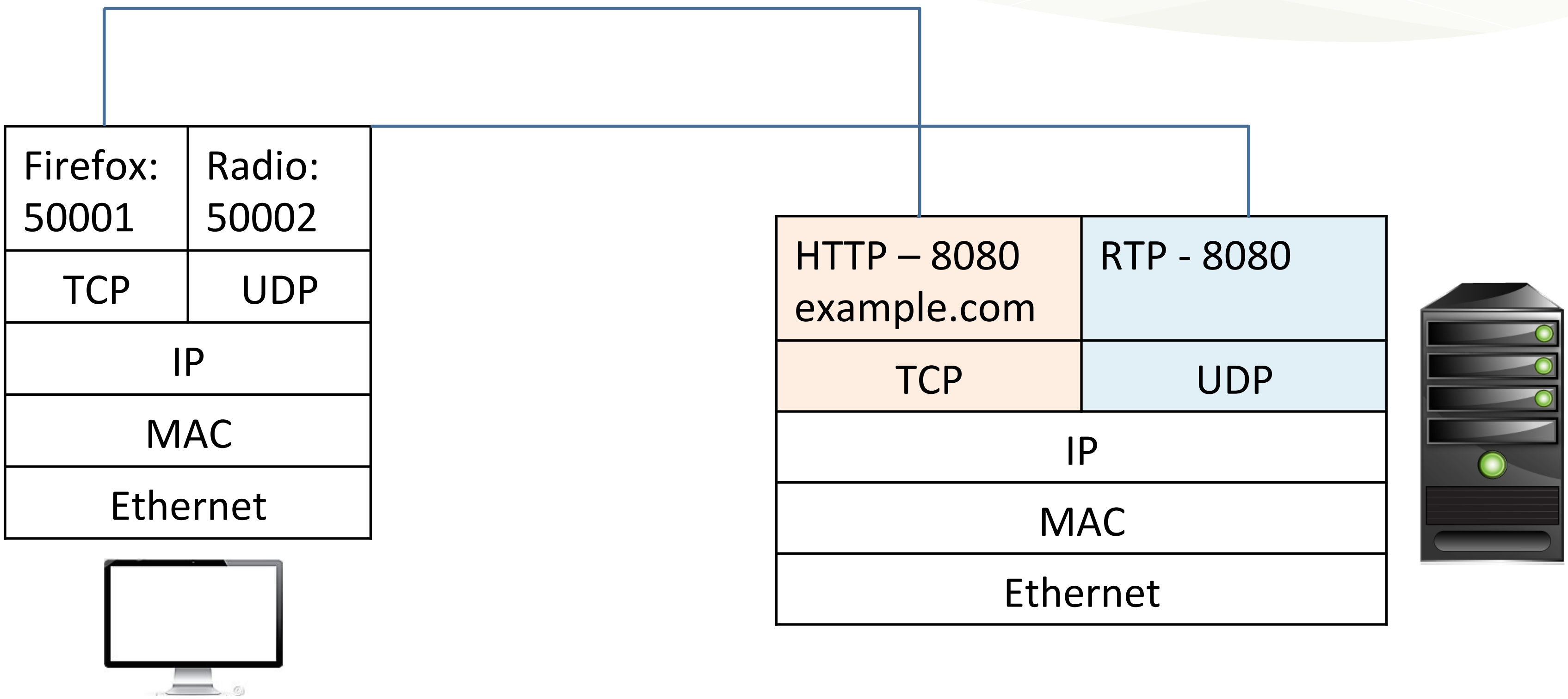
Encapsulation



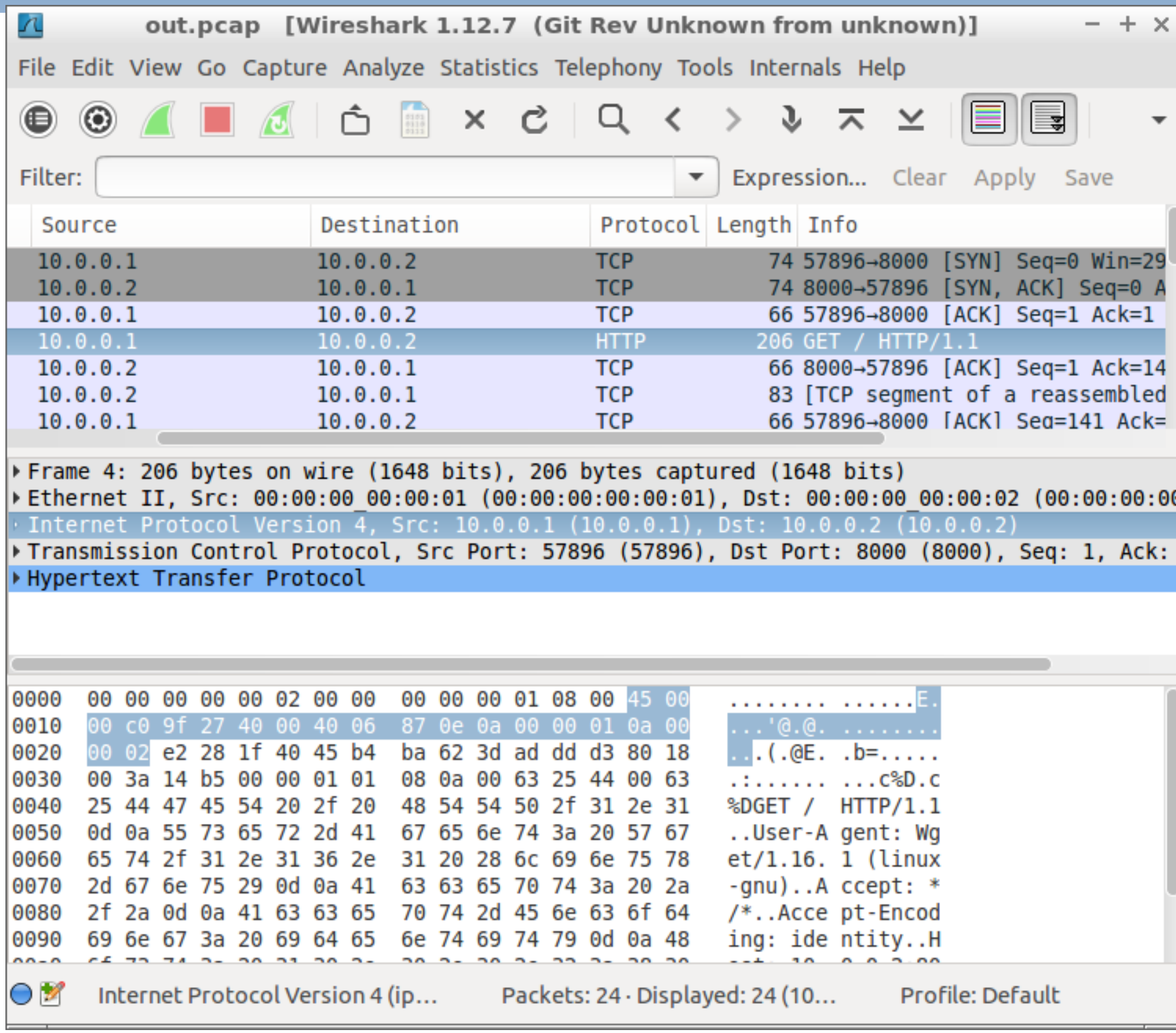
Multiplexing



Multiplexing



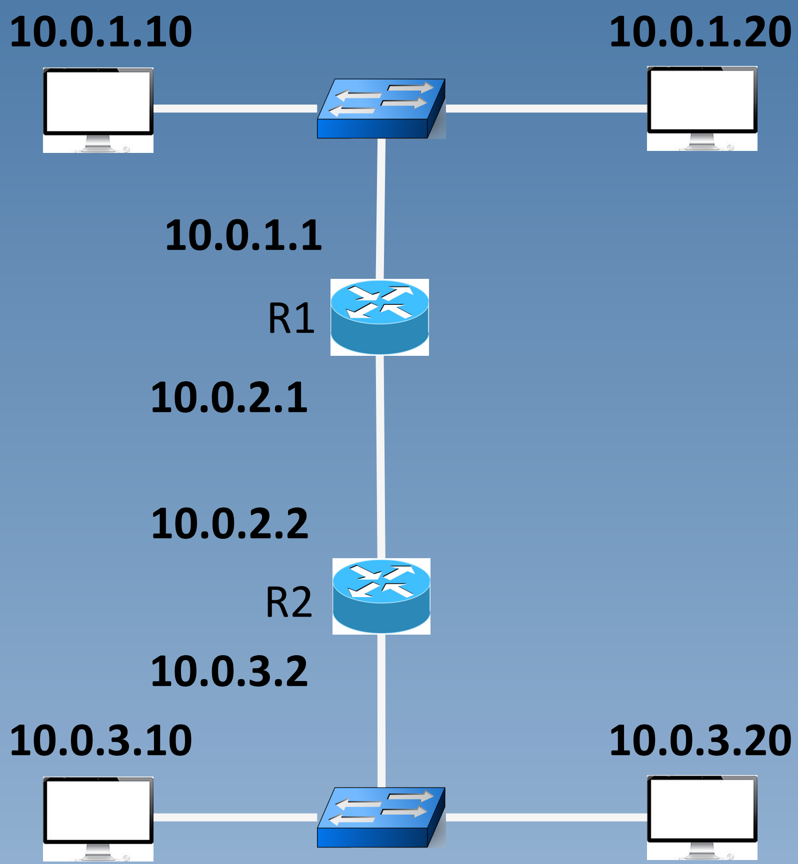
Encapsulation in real life



Introduction to Addressing

Targeting a network device

- To identify a router or an end host, to send it a packet for instance, such a device will need an address
- IP addresses identify the subnet and indicates the host ID of a device in Internet
  - Answer to the questions: Is the device located in my LAN or somewhere else? What is the ID of that device inside that/this LAN?
  - IP addresses allow inter and intra LAN communication
  - More about IPv4 and IPv6 later
- How to associate an IP address to a given hardware? This is the role of the MAC address
  - Intra LAN communications use MAC addresses + IP addresses



Inter LAN Communication

- IP addresses identify a device in the network.
- An IP address is composed of 4 bytes and represented in decimal format with a "." between two bytes.
- In an IP address, some bits identify the network address (network ID).
- Setting all the bits of the network ID to 1 gives the network mask (*netmask*) address

Classful addresses

- Classful networks uses an addressing scheme where the address space is divided in 5 classes
  - Class A: 8 bits for the network ID - Netmask: 255.0.0.0
    - Start address: 0.0.0.0 → 00000000.00000000.00000000.00000000
    - End address: 127.255.255.255 → 01111111.1111111.1111111.1111111
  - Class B: 16 bits for the network ID – Netmask: 255.255.0.0
    - Start address: 128.0.0.0 → 10000000.00000000.00000000.00000000
    - End address: 191.255.255.255 → 10111111.1111111.1111111.1111111
  - Class C: 24 bits for the network ID – Netmask: ??
    - Start address: 192.0.0.0 → 11000000.00000000.00000000.00000000
    - End address: 223.255.255.255→ 11011111.1111111.1111111.1111111
  - Class D: multicast
    - Start address: 224.0.0.0 → 11100000.00000000.00000000.00000000
    - End address: 239.255.255.255 → 11101111.1111111.1111111.1111111
  - Class E: reserved
    - Start address: 240.0.0.0 → 11110000. 00000000.0000000.00000000
    - End address: 255.255.255.255 → 11111111.1111111.1111111.1111111

Classful addresses (cont)

- The classful address scheme led to
  - Address exhaustion
  - Big routing tables
- Note that the first and the last addresses are not used to identify a host
  - The first address (*aka the zero address*) identifies the network address
    - The network address of a host with @IP 134.51.12.36 is 134.51.0.0
    - The network address of a host with @IP 212.54.12.36 is ??
  - The last address (*the all one address*) is the network broadcast address (e.g. to send a message to an entire LAN)
    - The broadcast network address for 134.51.0.0 is 134.51.255.255
    - The broadcast network address for 212.54.12.36 is ??
  - How many host can be addressed in a network?
    - 2^n-2 where n is the number of bits used for the host id's.

Classless Inter-Domain Routing

- CIDR is the current addressing scheme in Internet
- The number of bits used in a network address does not depend anymore on the first bits of the IP address
  - ➢ What is the network address for 134.59.17.36?
- CIDR uses a routing prefix to identify the number of bits composing a network address
  - ➢ Ex. 134.59.17.36/20
  - ➢ For 134.59.17.36/20, what is the network address?
    - ∗ Write in binary format: 10000110.00111011.00010001.00100100
    - ∗ Take the first 20 bits and the remaining ones set it to zero: 10000110.00111011.00010000.00000000
    - ∗ Netmask: 10000110.00111011.00010000.00000000 = 134.59.16.0

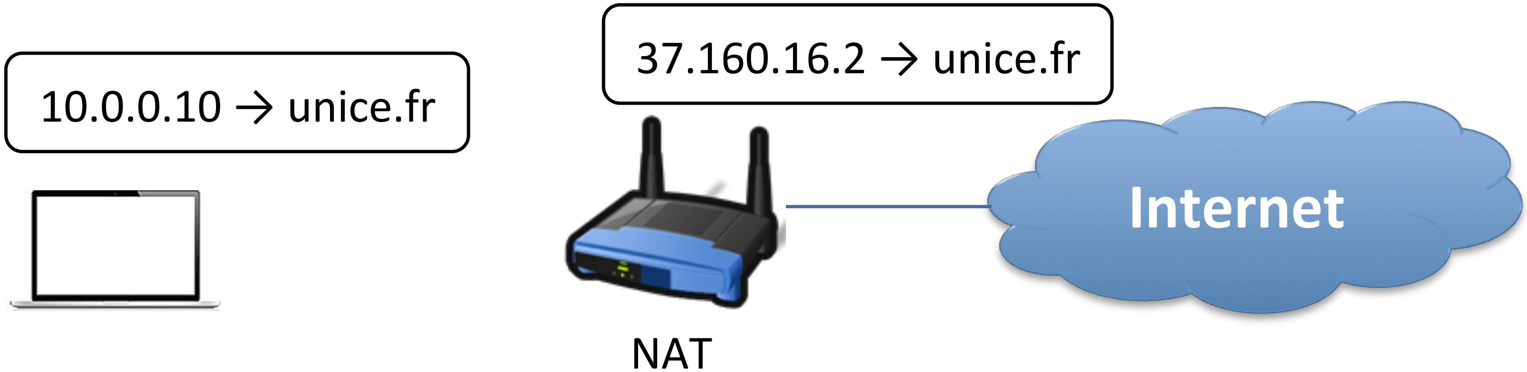
Classless Inter-Domain Routing (cont)

- For 134.59.17.36/20
  - What is the netmask?
    - Write the address in binary format: 10000110.00111011.00010001.00100100
    - Take the first 20 bits and set it to 1. Put the remaining ones to zero: 11111111.11111111.11110000.00000000
    - Write in decimal: 255.255.240.0
  - what is the network broadcast address?
    - Write in binary format: 10000110.00111011.00010001.00100100
    - Take the first 20 bits and set the remaining bits to one: 10000110.00111011.00011111.11111111
    - Write the value in decimal: 134.59.31.255
  - What is the first available IP address for a host?
    - Take the network address and put the last bit to one: 134.59.16.1
  - What is the last available IP address for a host?
    - Take the network broadcast address and set the last bit to zero: 134.59.31.254

Public vs Private IP addresses

- IP addresses can be public or private
  - Public addresses can be routed through the Internet
  - Private addresses must never be routed through the Internet
- Devices with an IP private address uses Network Address Translators (NATs) to reach the Internet
- IP addresses are in the following range
  - 10.0.0.0/8
  - 172.16.0.0/12
  - 192.168.0.0/16

Example with private addresses



Configuring Ethernet network interface cards in Linux through command line

- To display the configuration of network interfaces, which are in UP mode, use “ifconfig”
- To display the network interfaces, whatever they are in mode UP or DOWN, use “ifconfig -a”
- To enable (disable) a network interface, use “ifconfig *ethdev* up” (down), where ethdev is the name of the network interface as displayed by “ifconfig -a”
- To assign an IP address and netmask to a given interface. E.g. “ifconfig h1-eth0 10.0.0.1 netmask 255.255.255.0”

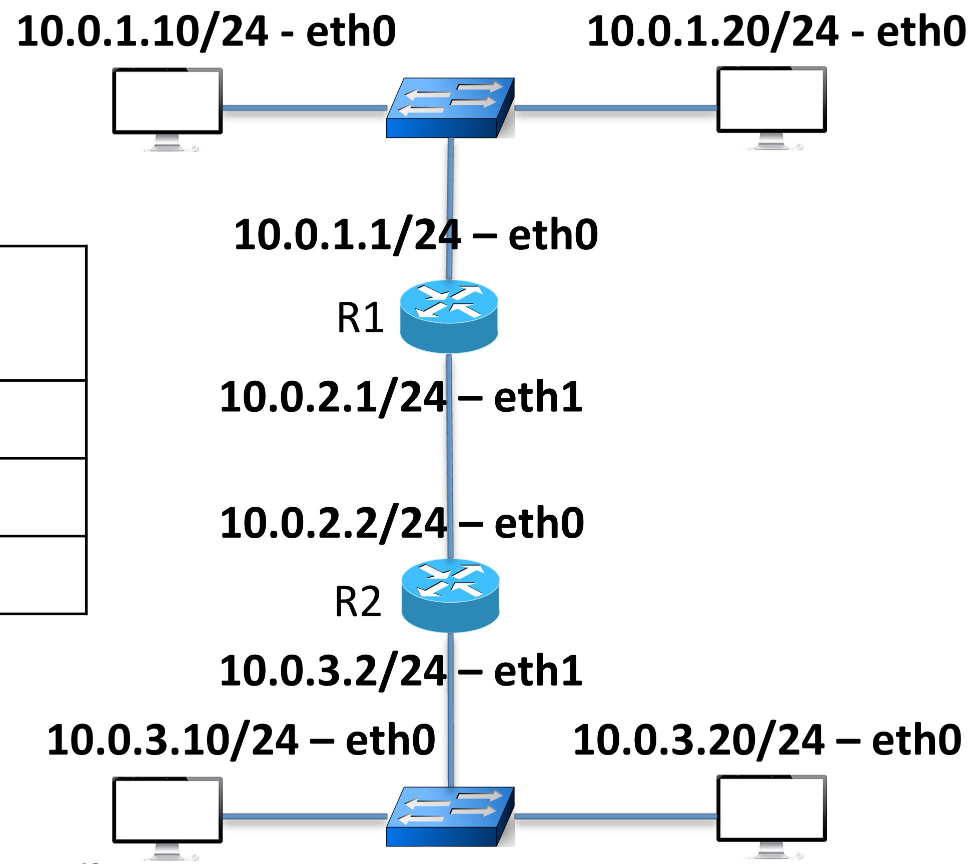
Basics on routing

- Each network has one or more gateways (routers)
- Each link of a router belongs to a different network address
- To reach an external network (external hosts), hosts and routers must keep a routing table

Routing table of an end host

Routing table of host 10.0.1.10
Note the use of a default route

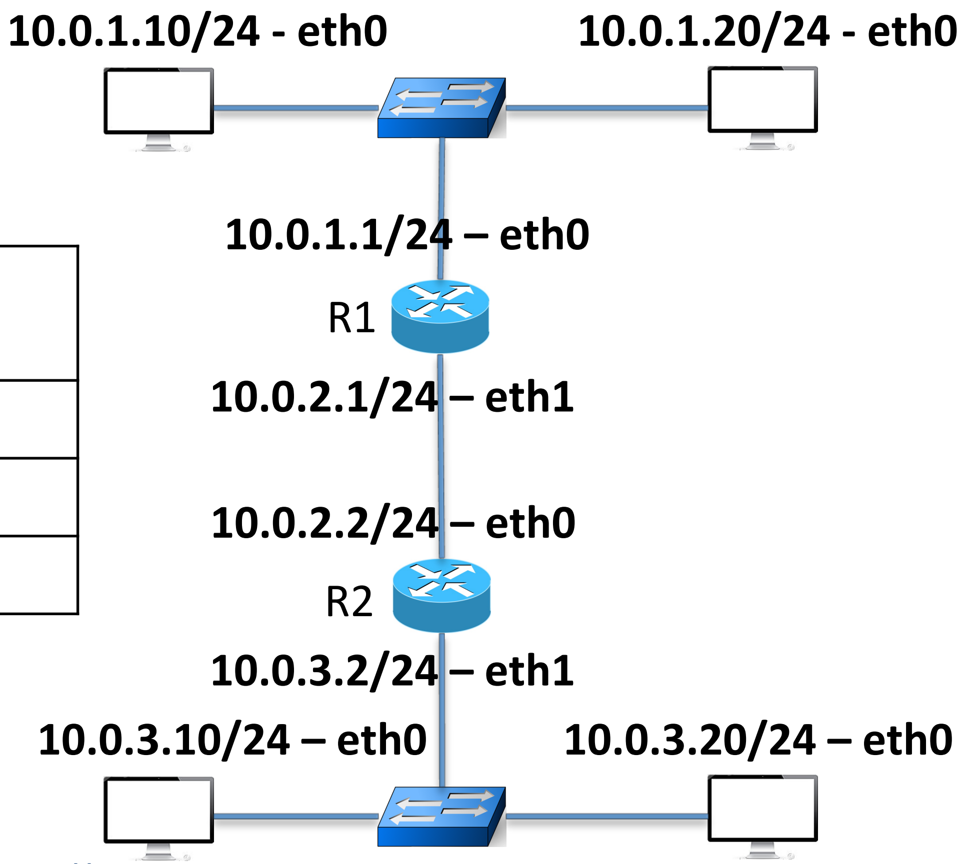
Targeted Network	IP Gateway	Device
10.0.1.0/24	*	Eth0
~~10.0.2.0/24~~	~~10.0.1.1~~	~~Eth0~~
*	10.0.1.1	Eth0



Routing table of a router

Routing table of router R2
Note the use of a default route

Targeted Network	IP Gateway	Device
10.0.3.0/24	*	Eth1
10.0.2.0/24	*	Eth0
*	10.0.2.1	Eth0



Statics route in a Linux box

R2 is a Linux box. We want to build its routing table

- To display the routing table, use “route -n”
- Add the first route to a directly attached net “route add -net 10.0.3.0 netmask 255.255.255.0 dev eth1”
- Add route to 10.0.2.0/24 : ??
- Add the default route: “route add default gw 10.0.2.1 dev eth0”
- Add a route to 1.2.3.0 “route add -net 1.2.3.0 netmask 255.255.255.0 dev eth0”
  - It was a mistake, remove such a route entry “route del -net 1.2.3.0 netmask 255.255.255.0 dev eth0”

Routing table at R2

Targeted Network	IP Gateway	Device
10.0.3.0/24	*	eth1
10.0.2.0/24	*	eth0
*	10.0.2.1	eth0

Lookup operations in hardware routers

EXAMPLE ROUTING TABLE

Entry No.	Address (Binary)	Output Port
1	101XX	A
2	0110X	B
3	011XX	C
4	10011	D

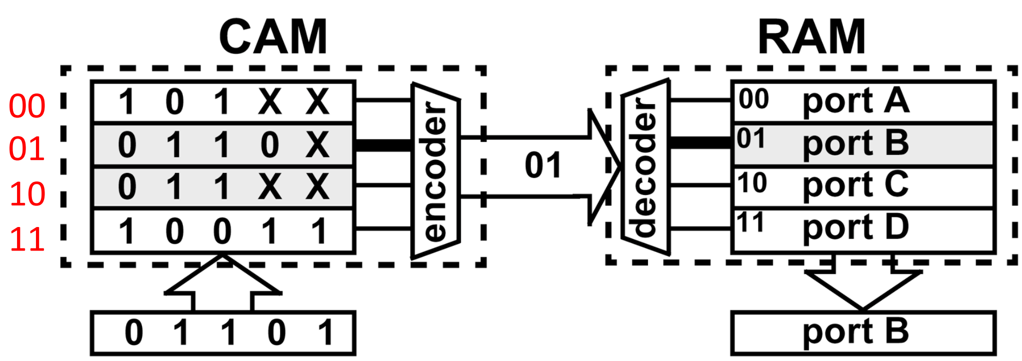
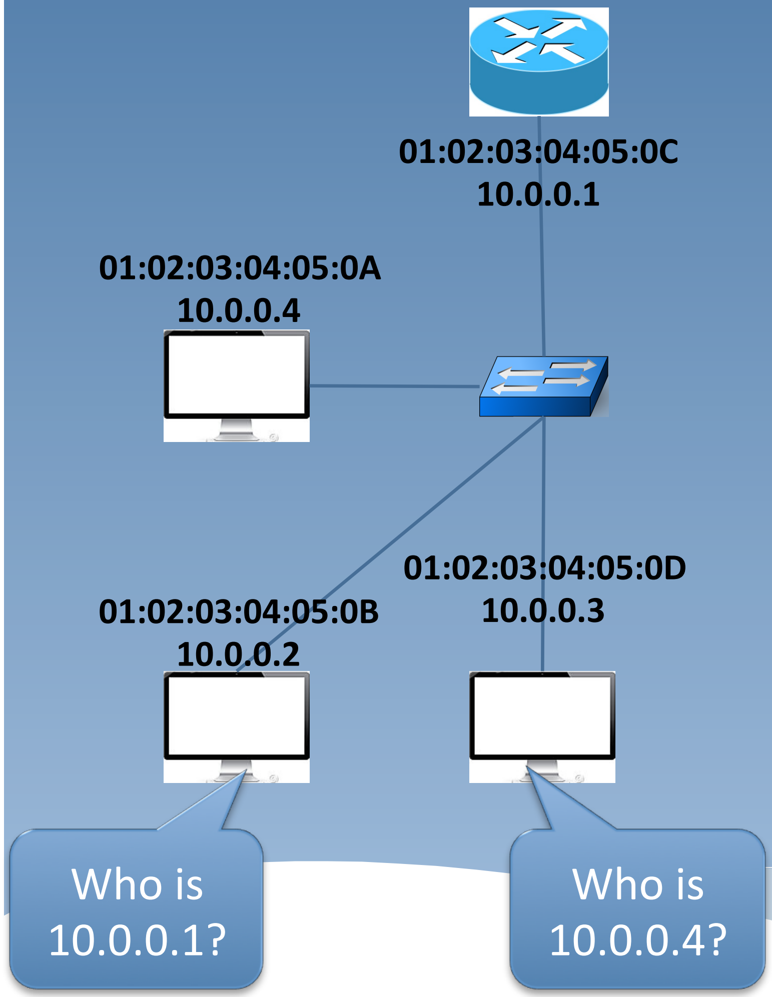


Image from: K. Pagiamtzis and A. Sheikholeslami, "Content Addressable Memory (CAM) Circuits and Architectures: A Tutorial and Survey," IEEE J. Solid-State Circuits, vol. 41, no. 3, pp. 712-727, Mar. 2006.

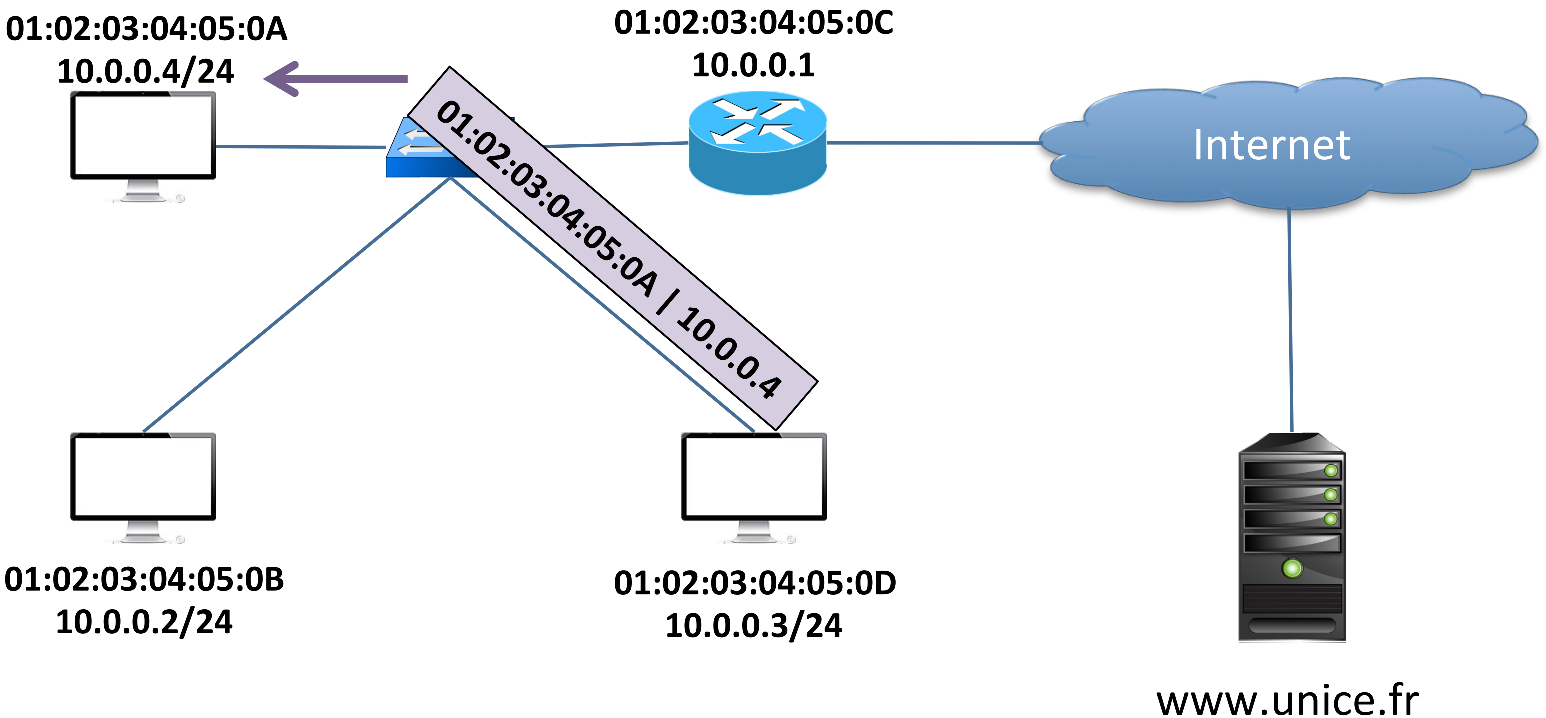


Intra LAN Communication

Intra LAN communication needs a MAC address to know which device is configured with a given IP address

A MAC address is composed of 6 bytes and represented in hexadecimal with ":" or "-" between two bytes

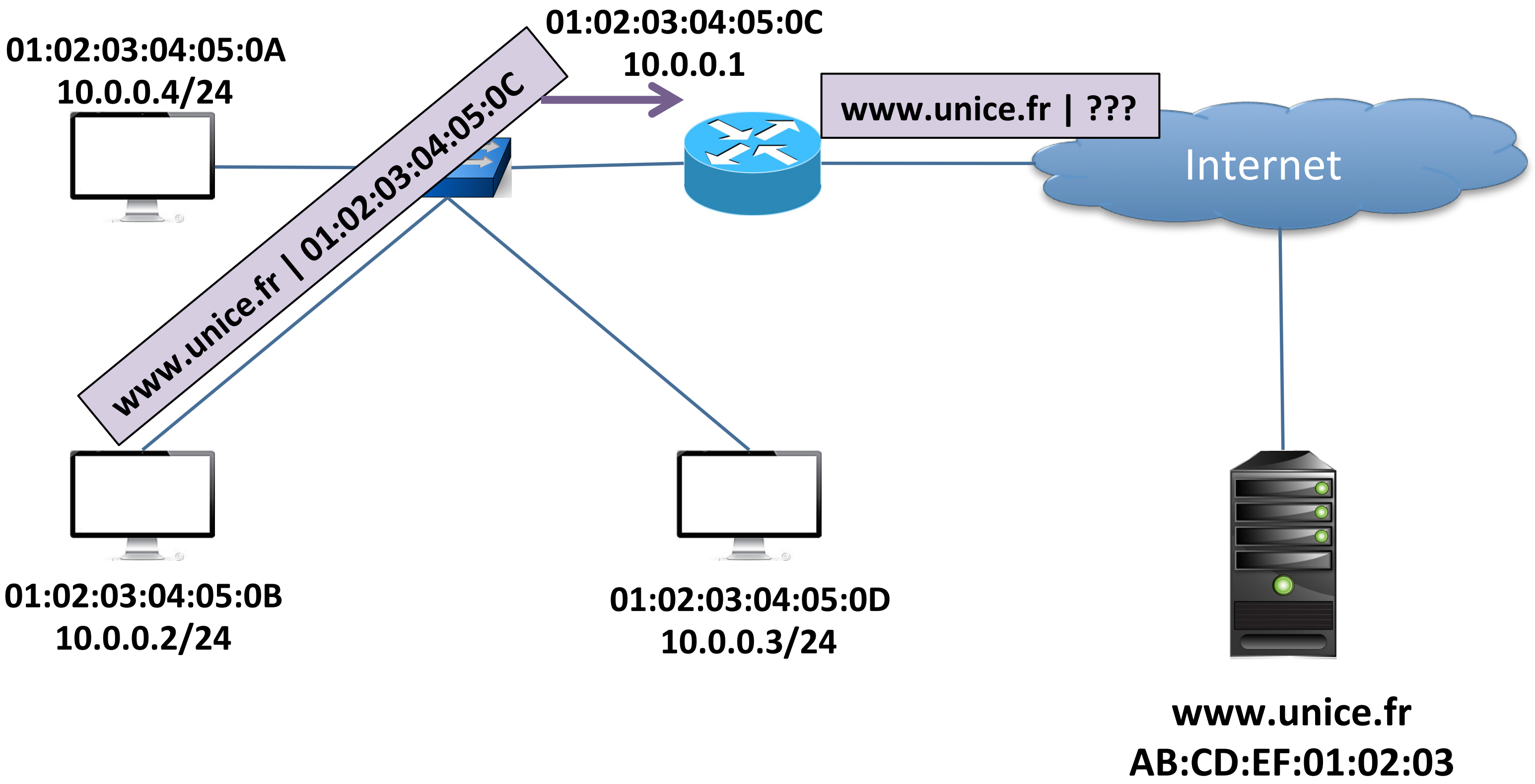
Example 1: 10.0.0.3 → 10.0.0.4



Overview of Layer 2 devices

- Switches ports do not have IP addresses
- How does a switch do to forward packets?
  - Switches have a forwarding database (FDB)
  - The forwarding database associates the port number with the MAC address of the host plugged in that port
  - Switches learn the MAC addresses when the host connected to it sends a packet
  - If the MAC address destination is already in the FDB, the switch forward the packet in the right port. Otherwise, the packet is sent through all the ports, except the one where the packet comes from.
  - Packets with the broadcast MAC address are sent through all the ports except the one where the packet comes from.

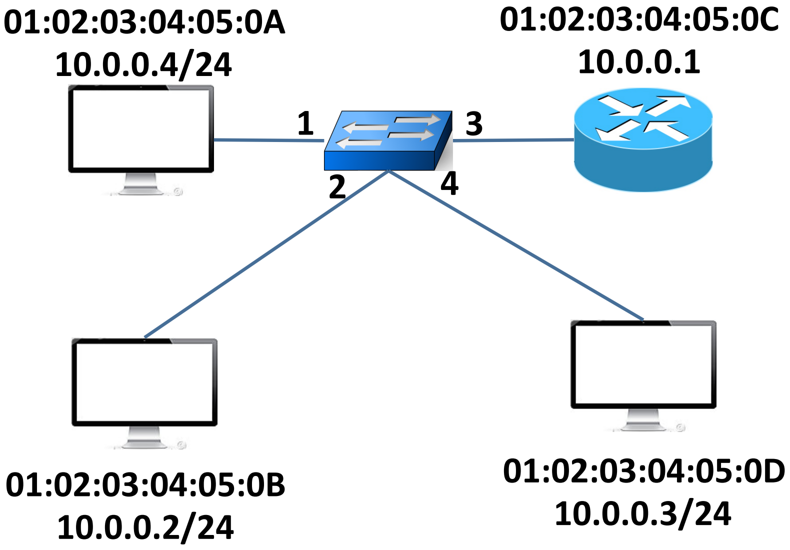
Example 2: 10.0.0.2 → www.unice.fr



Forwarding table in a switch

- Expected forwarding database of the switch

@MAC	#Port
01:02:03:04:05:0A	1
01:02:03:04:05:0B	2
01:02:03:04:05:0C	3
01:02:03:04:05:0D	4

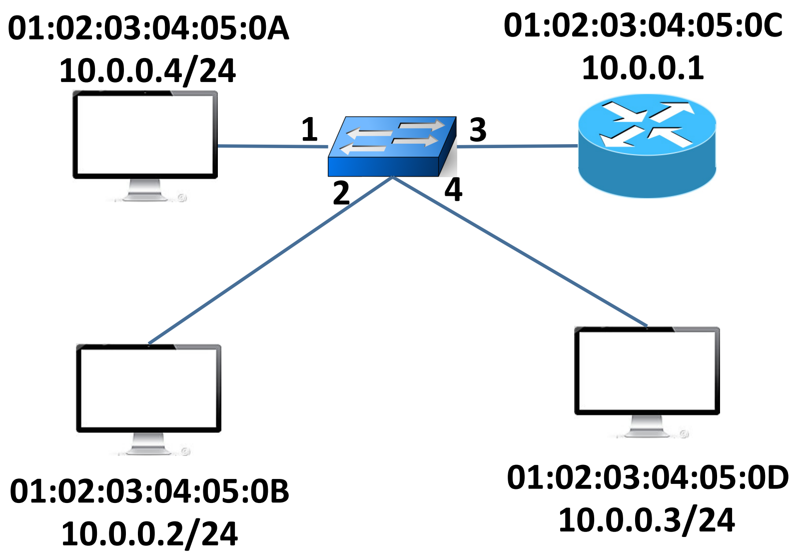


The Address Resolution Protocol (ARP)

- Suppose that you know the IP address of the targeted device inside your LAN, so how do you get its MAC address?
  - This is done by the ARP
  - You never get the MAC address of a device from another LAN (i.e. on the other side of the gateway)
- When a host needs to know the MAC address of a host inside the LAN, it broadcasts an ARP request
  - In the request, you will find the IP address of the targeted host.
- Every device will take the ARP request. The one that will see its IP address inside must send an ARP reply
  - The ARP reply contains the IP address of the sender and its MAC address
  - Applications can now communicate
- ARP is a Link-Layer protocol

ARP table

- Expected ARP table at host 10.0.0.4
  - Communication with the router and host 10.0.0.3
  - Add route to 10.0.0.2 “arp -s 10.0.0.2 01:02:03:04:05:0B”
  - Delete route to 10.0.0.5 “arp –d 10.0.0.5”
  - And show the ARP table “arp –n”



@IP	@MAC	dev
10.0.0.3	01:02:03:04:05:0D	eth0
10.0.0.1	01:02:03:04:05:0C	eth0
10.0.0.2	01:02:03:04:05:0B	eth0

Some special addresses

- To send a single message to the entire LAN, you must send a broadcast messages
  - The IP broadcast address is 255.255.255.255
  - The MAC broadcast address is FF:FF:FF:FF:FF:FF
- The IP loopback address, which allows to communicate two processes in a single machine by mean of the network protocol stack.
  - Packets with this destination never leaves the host
  - By default, 127.0.0.1
  - But defined to be 127.0.0.0/8 by RFC6890