

# Network Computing courses

Maël Auzias

ENSIBS - UBS

October 2014



Figure: [teaching.auzias.net](http://teaching.auzias.net)

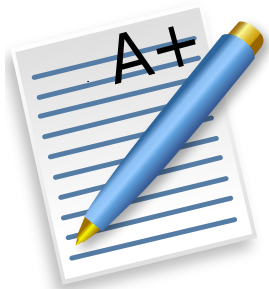
# Course details

## Objectives

- How do *computers* communicate?
- What are the mechanisms **under** an HTTP request or a telegram message?
- Networks are all around us, better study them!



# Course details



## Evaluation

- Short test at the beginning of every lesson (5 min) ?
- Project
- Final exam (1 hour)
- All same weighting

## Material

- Slides available at [teaching.auzias.net](https://teaching.auzias.net) (github too)

# Presentation Outline

- 1 Introduction
  - Definitions and presentation
  - Network classification
  - HTTP request/response example
  - Models overview (OSI and TCP/IP)
- 2 Lower layers
  - Physical
  - Data Link
  - Network
  - Transport
- 3 Upper layers
  - Session
  - Presentation
  - Application

# Definitions

- **Network:** an **interconnected** group or system

# Definitions

- **Network:** an **interconnected** group or system
- **Internet:** world wide **interconnected system of networks**  
RFC791 (1981)

# Definitions

- **Network:** an **interconnected** group or system
- **Internet:** world wide **interconnected system of networks**  
[RFC791 \(1981\)](#)
- **IP:** Internet **Protocol** that provides the functions necessary to deliver a package of bits from a source to a destination over a network

# Definitions

- **Network:** an **interconnected** group or system
- **Internet:** world wide **interconnected system of networks**  
[RFC791 \(1981\)](#)
- **IP:** Internet **Protocol** that provides the functions necessary to deliver a package of bits from a source to a destination over a network
- **(world wide) Web: network** consisting of a collection of Internet websites using HTTP



# Definitions

- **HTTP:** Hypertext Transfer Protocol **Protocol**, application-level protocol for distributed, collaborative, hypermedia information systems [draft HTTP2 \(July 2014\)](#)

# Definitions

- **HTTP:** Hypertext Transfer Protocol **Protocol**, application-level protocol for distributed, collaborative, hypermedia information systems [draft HTTP2 \(July 2014\)](#)
- **TCP:** Transmission Control **Protocol** is intended for use as a highly reliable host-to-host [RFC761 \(January 1980\)](#)

# Definitions

- **HTTP:** Hypertext Transfer Protocol **Protocol**, application-level protocol for distributed, collaborative, hypermedia information systems [draft HTTP2 \(July 2014\)](#)
- **TCP:** Transmission Control **Protocol** is intended for use as a highly reliable host-to-host [RFC761 \(January 1980\)](#)
- **UDP:** User Datagram **Protocol** provides a procedure for application programs to send messages to other programs with a minimum of protocol mechanism [RFC768 \(August 1980\)](#)

# Definitions

- **Router:** network **hardware** providing routing services

# Definitions

- **Router:** network **hardware** providing routing services
- **Routing:** **algorithm processed** to decide where to forward a packet

# Definitions

- **Router:** network **hardware** providing routing services
- **Routing:** **algorithm processed** to decide where to forward a packet
- **Forwarding:** **action** of moving a packet from an NIC to another

# Definitions

- **Router:** network **hardware** providing routing services
- **Routing:** **algorithm processed** to decide where to forward a packet
- **Forwarding:** **action** of moving a packet from an NIC to another
- **NIC:** Network Interface Card
- **Switch (hub):** network **hardware** that connect systems together using packet switching

# Definitions

- **Router:** network **hardware** providing routing services
- **Routing:** **algorithm processed** to decide where to forward a packet
- **Forwarding:** **action** of moving a packet from an NIC to another
- **NIC:** Network Interface Card
- **Switch (hub):** network **hardware** that connect systems together using packet switching
- **Packet switching:** forward-like method regardless of the content (destination-based)



# Definitions

- **Node (network):** any entity that can send/receive packets from a network through a NIC

# Definitions

- **Node (network):** any entity that can send/receive packets from a network through a NIC
- **Client: computer** able to send requests to a server

# Definitions

- **Node (network):** any entity that can send/receive packets from a network through a NIC
- **Client: computer** able to send requests to a server
- **Request: application message** destined for a server (*order*)

# Definitions

- **Node (network):** any entity that can send/receive packets from a network through a NIC
- **Client: computer** able to send requests to a server
- **Request: application message** destined for a server (*order*)
- **Server: computer** able to respond a client's requests

# Definitions

- **Node (network):** any entity that can send/receive packets from a network through a NIC
- **Client: computer** able to send requests to a server
- **Request: application message** destined for a server (*order*)
- **Server: computer** able to respond a client's requests
- **Request: application message** destined for a client (*result*)

# Definitions

- **Node (network):** any entity that can send/receive packets from a network through a NIC
- **Client: computer** able to send requests to a server
- **Request: application message** destined for a server (*order*)
- **Server: computer** able to respond a client's requests
- **Request: application message** destined for a client (*result*)
- **Fat client: application** where most functions are processed by the client itself

# Definitions

- **Node (network):** any entity that can send/receive packets from a network through a NIC
- **Client: computer** able to send requests to a server
- **Request: application message** destined for a server (*order*)
- **Server: computer** able to respond a client's requests
- **Request: application message** destined for a client (*result*)
- **Fat client: application** where most functions are processed by the client itself
- **Thin client: application** where most functions are carried out on a central server

# What kind of networks is it ?

- **BAN:** Body Area Network



# What kind of networks is it ?

- **BAN:** Body Area Network
- **PAN:** Personal Area Networks

# What kind of networks is it ?

- **BAN:** Body Area Network
- **PAN:** Personal Area Networks
- **(W)LAN:** (Wireless) Local Area Networks (home, office, school or airport)

# What kind of networks is it ?

- **BAN:** Body Area Network
- **PAN:** Personal Area Networks
- **(W)LAN:** (Wireless) Local Area Networks (home, office, school or airport)
- **MAN:** Metropolitan Area Networks, can cover a whole city

# What kind of networks is it ?

- **BAN:** Body Area Network
- **PAN:** Personal Area Networks
- **(W)LAN:** (Wireless) Local Area Networks (home, office, school or airport)
- **MAN:** Metropolitan Area Networks, can cover a whole city
- **WAN:** Wide Area Networks cover a broad area (Internet)

# Topologies

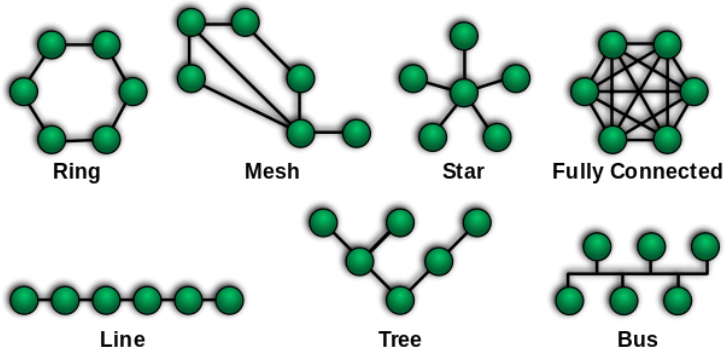


Figure: [upload.wikimedia.org](https://upload.wikimedia.org)

# Topologies

- **Point-to-point:** two entities directly connected to each other (tunnel).

---

<sup>1</sup>Hong Kong protesters use a mesh network to organize 

# Topologies

- **Point-to-point:** two entities directly connected to each other (tunnel).
- **Ring:** data go around the ring, unidirectional way network.

---

<sup>1</sup>Hong Kong protesters use a mesh network to organize 

# Topologies

- **Point-to-point:** two entities directly connected to each other (tunnel).
- **Ring:** data go around the ring, unidirectional way network.
- **Mesh:** all nodes cooperate in the distribution of data in the network<sup>1</sup>.

---

<sup>1</sup>Hong Kong protesters use a mesh network to organize 



# Topologies

- **Point-to-point:** two entities directly connected to each other (tunnel).
- **Ring:** data go around the ring, unidirectional way network.
- **Mesh:** all nodes cooperate in the distribution of data in the network<sup>1</sup>.
- **Star:** all messages go through the same central node, reducing network failure.

---

<sup>1</sup>Hong Kong protesters use a mesh network to organize 

# Topologies

- **Point-to-point:** two entities directly connected to each other (tunnel).
- **Ring:** data go around the ring, unidirectional way network.
- **Mesh:** all nodes cooperate in the distribution of data in the network<sup>1</sup>.
- **Star:** all messages go through the same central node, reducing network failure.
- **Fully connected:** all nodes are connected to all other nodes.

---

<sup>1</sup>[Hong Kong protesters use a mesh network to organize](#) 

# Topologies

- **Point-to-point:** two entities directly connected to each other (tunnel).
- **Ring:** data go around the ring, unidirectional way network.
- **Mesh:** all nodes cooperate in the distribution of data in the network<sup>1</sup>.
- **Star:** all messages go through the same central node, reducing network failure.
- **Fully connected:** all nodes are connected to all other nodes.
- **Line:** bidirectional link between two nodes. Node can only send packet going through its neighbors.

---

<sup>1</sup>[Hong Kong protesters use a mesh network to organize](#)          

# Topologies

- **Point-to-point:** two entities directly connected to each other (tunnel).
- **Ring:** data go around the ring, unidirectional way network.
- **Mesh:** all nodes cooperate in the distribution of data in the network<sup>1</sup>.
- **Star:** all messages go through the same central node, reducing network failure.
- **Fully connected:** all nodes are connected to all other nodes.
- **Line:** bidirectional link between two nodes. Node can only send packet going through its neighbors.
- **Bus:** all nodes are connected to the same media. Only one at a time can send packet, that all other receives.

---

<sup>1</sup>[Hong Kong protesters use a mesh network to organize](#) 

# Topologies

- **Point-to-point:** two entities directly connected to each other (tunnel).
- **Ring:** data go around the ring, unidirectional way network.
- **Mesh:** all nodes cooperate in the distribution of data in the network<sup>1</sup>.
- **Star:** all messages go through the same central node, reducing network failure.
- **Fully connected:** all nodes are connected to all other nodes.
- **Line:** bidirectional link between two nodes. Node can only send packet going through its neighbors.
- **Bus:** all nodes are connected to the same media. Only one at a time can send packet, that all other receives.
- **Tree:** hierarchical topology, such as, i.e., binary tree.

# <sup>1</sup>Hong Kong protesters use a mesh network to organize

# Bonus

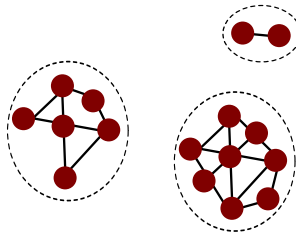


Figure: Disconnected MANET illustration [1]

# Bonus

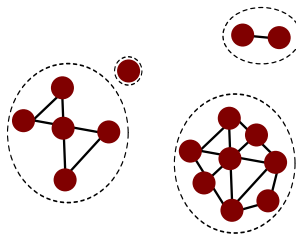


Figure: Store-carry-and-forward [1]

# Bonus

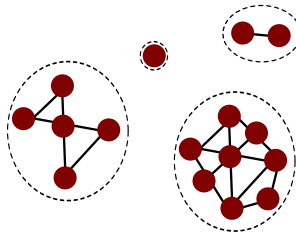


Figure: Store-carry-and-forward [1]



# Bonus

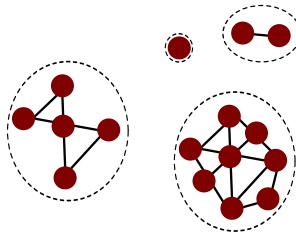


Figure: Store-carry-and-forward [1]

# Bonus

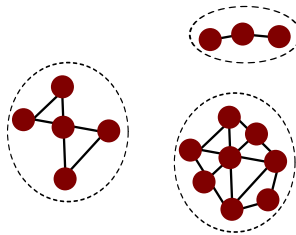


Figure: Store-carry-and-forward [1]

# HTTP request/response example

Enter [getbootstrap.com](https://getbootstrap.com) in your browser

# HTTP request/response example

Enter [getbootstrap.com](http://getbootstrap.com) in your browser

Source	Destination	Protocol	Length	Info
192.168.0.48	208.67.222.222	DNS	76	Standard query 0x4797 A getbootstrap.com
208.67.222.222	192.168.0.48	DNS	108	Standard query response 0x4797 A 192.30.252.154 A 192.30.252.153

Figure: DNS request/response

# HTTP request/response example

Enter [getbootstrap.com](http://getbootstrap.com) in your browser

Source	Destination	Protocol	Length	Info
192.168.0.48	208.67.222.222	DNS	76	Standard query 0x4797 A getbootstrap.com
208.67.222.222	192.168.0.48	DNS	108	Standard query response 0x4797 A 192.30.252.154 A 192.30.252.153

Figure: DNS request/response

Source	Destination	Protocol	Length	Info
127.0.0.1	127.0.0.13	TCP	74	36159 > http [SYN] Seq=0 Win=43690 Len=0 MSS=65495 SACK_PERM=1 TSval=12
127.0.0.13	127.0.0.1	TCP	74	http > 36159 [SYN, ACK] Seq=0 Ack=1 Win=43690 Len=0 MSS=65495 SACK_PERM
127.0.0.1	127.0.0.13	TCP	66	36159 > http [ACK] Seq=1 Ack=1 Win=43776 Len=0 TSval=122257 TSecr=12225
127.0.0.1	127.0.0.13	HTTP	356	GET /index.html HTTP/1.1
127.0.0.13	127.0.0.1	TCP	66	http > 36159 [ACK] Seq=1 Ack=291 Win=44800 Len=0 TSval=122259 TSecr=122
127.0.0.13	127.0.0.1	HTTP	354	HTTP/1.1 200 OK (text/html)
127.0.0.1	127.0.0.13	TCP	66	36159 > http [ACK] Seq=291 Ack=289 Win=44800 Len=0 TSval=122259 TSecr=1
127.0.0.1	127.0.0.13	HTTP	357	GET /favicon.ico HTTP/1.1
127.0.0.13	127.0.0.1	HTTP	565	HTTP/1.1 404 Not Found (text/html)
127.0.0.1	127.0.0.13	TCP	66	36159 > http [ACK] Seq=582 Ack=788 Win=45952 Len=0 TSval=122269 TSecr=1

Figure: HTTP request/response

# How does messages reach destination ?

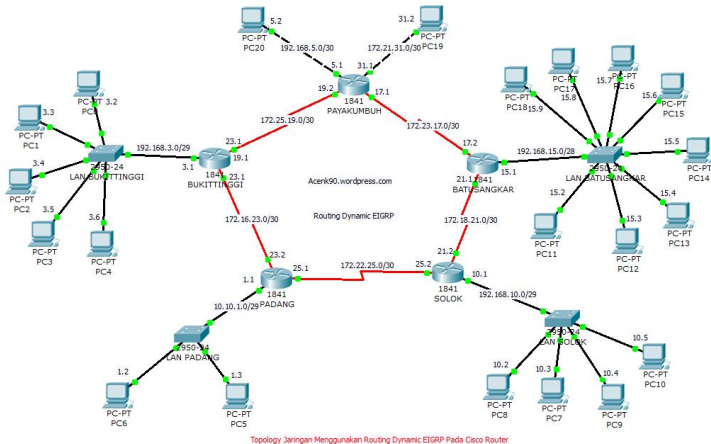
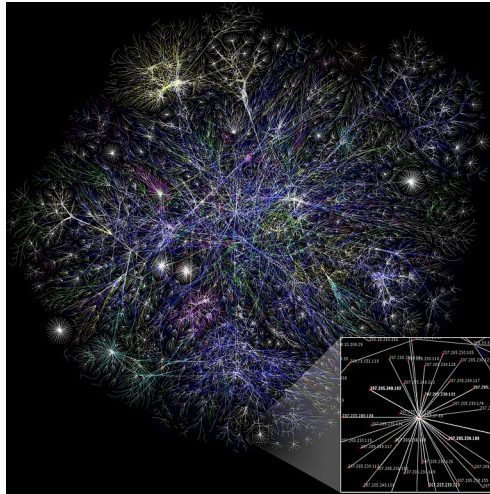


Figure: [acenk90.files.wordpress.com](http://acenk90.files.wordpress.com)

## More like this...



# How does it work ? From signal to application...

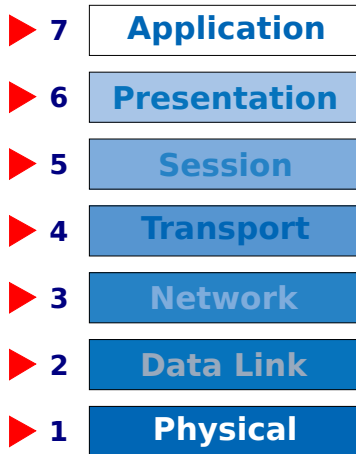


Figure: OSI model



# Presentation Outline

- 1 Introduction
  - Definitions and presentation
  - Network classification
  - HTTP request/response example
  - Models overview (OSI and TCP/IP)
- 2 Lower layers
  - Physical
  - Data Link
  - Network
  - Transport
- 3 Upper layers
  - Session
  - Presentation
  - Application

# From analog/logical signals up to messages



# Course details

# Course details

# Course details

# Presentation Outline

- 1 Introduction
  - Definitions and presentation
  - Network classification
  - HTTP request/response example
  - Models overview (OSI and TCP/IP)
- 2 Lower layers
  - Physical
  - Data Link
  - Network
  - Transport
- 3 Upper layers
  - Session
  - Presentation
  - Application

# Course details



# Course details

# Course details

# References



Maurice J. Khabbaz, Assi Chadi M., and Fawaz Wissam F.  
Disruption-Tolerant Networking: A Comprehensive Survey on  
Recent Developments and Persisting Challenges.  
*IEEE communications surveys and tutorials*, 2012.