

# How INTERNET works !

1

**basics**

# Internet ...

2

- It's a collection of networks interconnected with each other all over the planet
- It has specific and shared rules of operation, followed by all the networks in order to make the system work properly
- It is governed neither by a group nor by an institution or a company
- It is not centrally managed as all the single networks which forms INTERNET have an autonomous management system
- It is not a synonymous of World Wide Web

# Communication Data Protocol

3

- A communication data protocol is a formal description of digital message **formats** and **rules** for exchanging digital data throughout a network made of computing systems
- Communications among computing systems need a standard and shared mechanism for message format and exchange mechanism
- Protocols define the shared rules to handshake messages (the same as human dialogue)

# Internet works thanks to TCP/IP

4

- The task of TCP protocol consists in splitting «files» into many little packets, appending new field: destination address and a progressive number. When the packet arrives this protocol have to reassemble correctly the file starting from all the parts
- IP protocol forwards packets through the network choosing the best route, if necessary it might split the packets into little frames while it is working on riassembling it, in order to give it back to TCP layer

# TCP/IP in a more comprehensive contest

5

- **Standardization process has achieved a solution named protocol stacks**
  - There is no one protocol that manages all the phases of a communication, but a series of changeover, each of which deals with a different and well-defined task
  - The approach is more easy
  - It is possible to choose different protocols belonging to the same layer according to the service you want to call

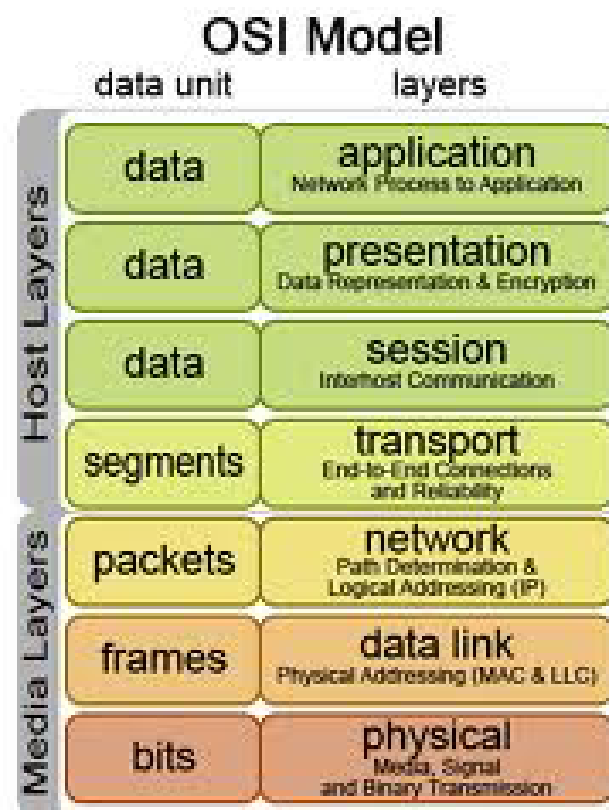
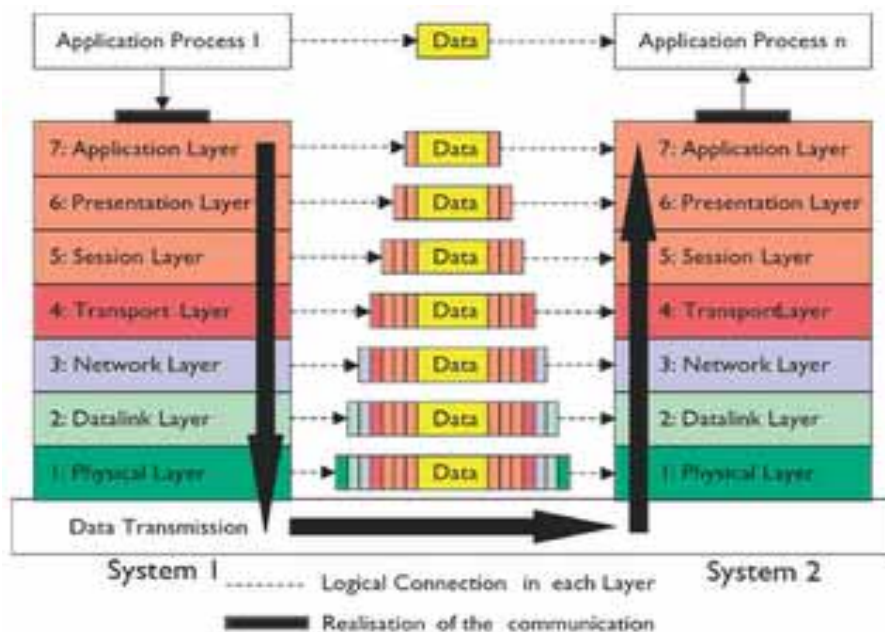
# Protocols stack

6

- Each layer “talks” to a layer just underneath, it should not care about layers that are located even more at the bottom of the stack (layer abstraction concept)
- To send an e-mail you don’t need to know if you will use a wireless or a fiber link, but you need to know if your message has been correctly delivered to the destination
- The same when you send out a letter, the user requires a service but he does not know how the letter will be transported and which streets or cities it will get across

# ISO/OSI stack (Open System Interconnection)

7



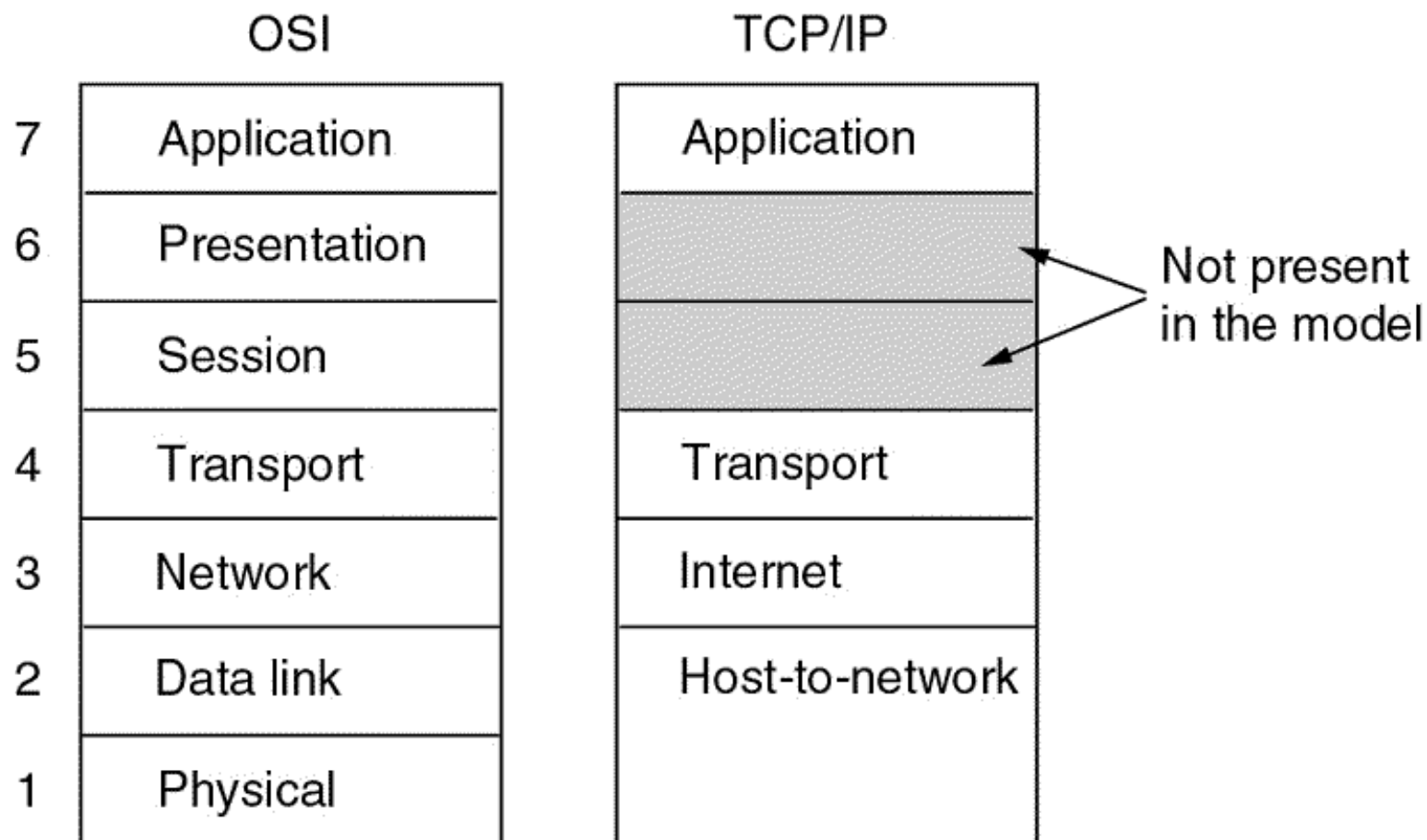
# Internet stack

8

- Similar to ISO/OSI stack
- It has less layers - only 5 in the protocol stack
- Layers sometimes include functionalities belonging to more than one ISO/OSI level, sometimes include only part of a particular functionality

# ISO/OSI vs. Internet

9



# Physical and Data link

10

- The same as in ISO/OSI stack
- **Physical layer**
  - It deals with the definition of:
    - ✦ Bit-level electrical/light communication across the channel
    - ✦ Voltage/current levels, signal strength, connector and media
    - ✦ Signals modulation type
- **Data link layer**
  - It deals with flow regulation, error detection and control, loss of frames. These functions are applied to a single link
  - It could claim for a re-transmission of frames that have not been correctly delivered
  - Some examples: ethernet, wlan, token ring, PPP

# Network layer

11

- It deals with a correct forwarding of packets coming from a source up to the destination
- Relating to this layer it exists only one protocol. This is the main difference between the other layer in the Internet stack
- It does not provide flow control and sequence packet check
- This level make the network transparent to the applications

# IP protocol

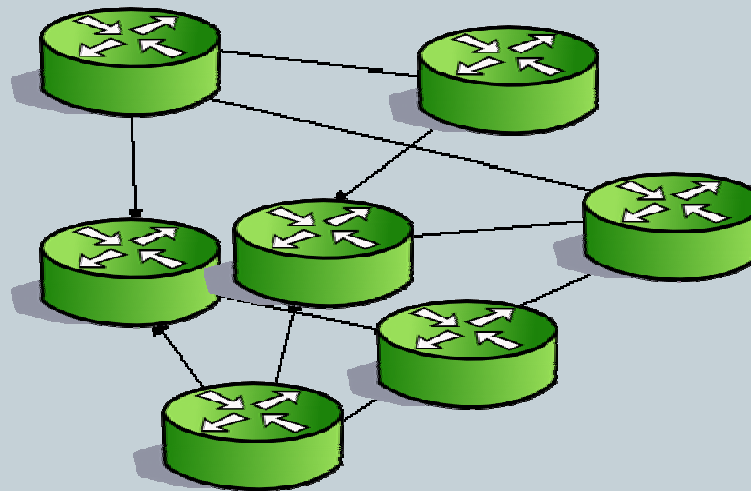
12

- This protocol defines how to deliver packets from a sender host to a recipient host
- Main features:
  - Connection less: each packet is handled independently one from each other;
  - Un-reliable: there is not any mechanism that can guarantees packet delivery (packets could be lost, duplicated, delayed or delivered out-of-order)

# IP Protocol

13

- The network is a set of nodes, named “Router” ... that is *forwarders*
- Each router receives a packet and transmits it using what it thinks is the best exit
- There are a large set of mechanism to forward packets



# Transport Layer

14

- It deals with defining the type of service that is required by applications
  - Errors check
  - flow and congestions control (if required)
  - out-of-order packet check (if required)
- The main protocols are
  - TCP
  - UDP

# TCP

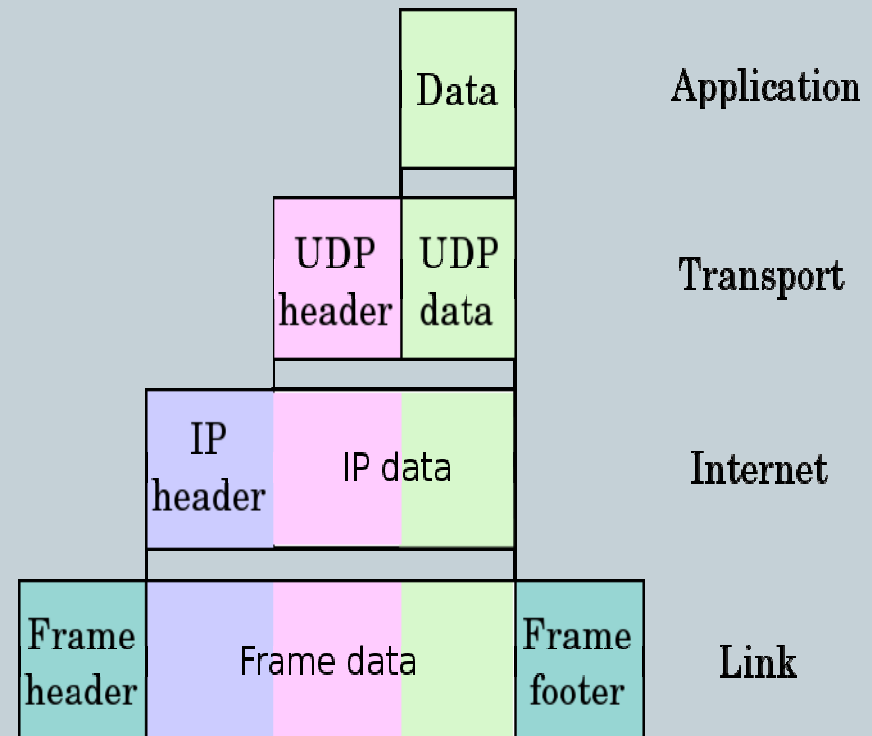
15

- Transmission control protocol
- It guaranties a *reliable service* over an *un-reliable network* (such as IP network)
- Connection oriented:
  - Two hosts exchange “port numbers” information
  - Each packet received is notified to the source with an ACK message (acknowledgement)
  - The connection is closed when the trasmission finish
  - TCP is used when the transmission has not a strict delay requirement but it requires high reliability

# UDP protocol

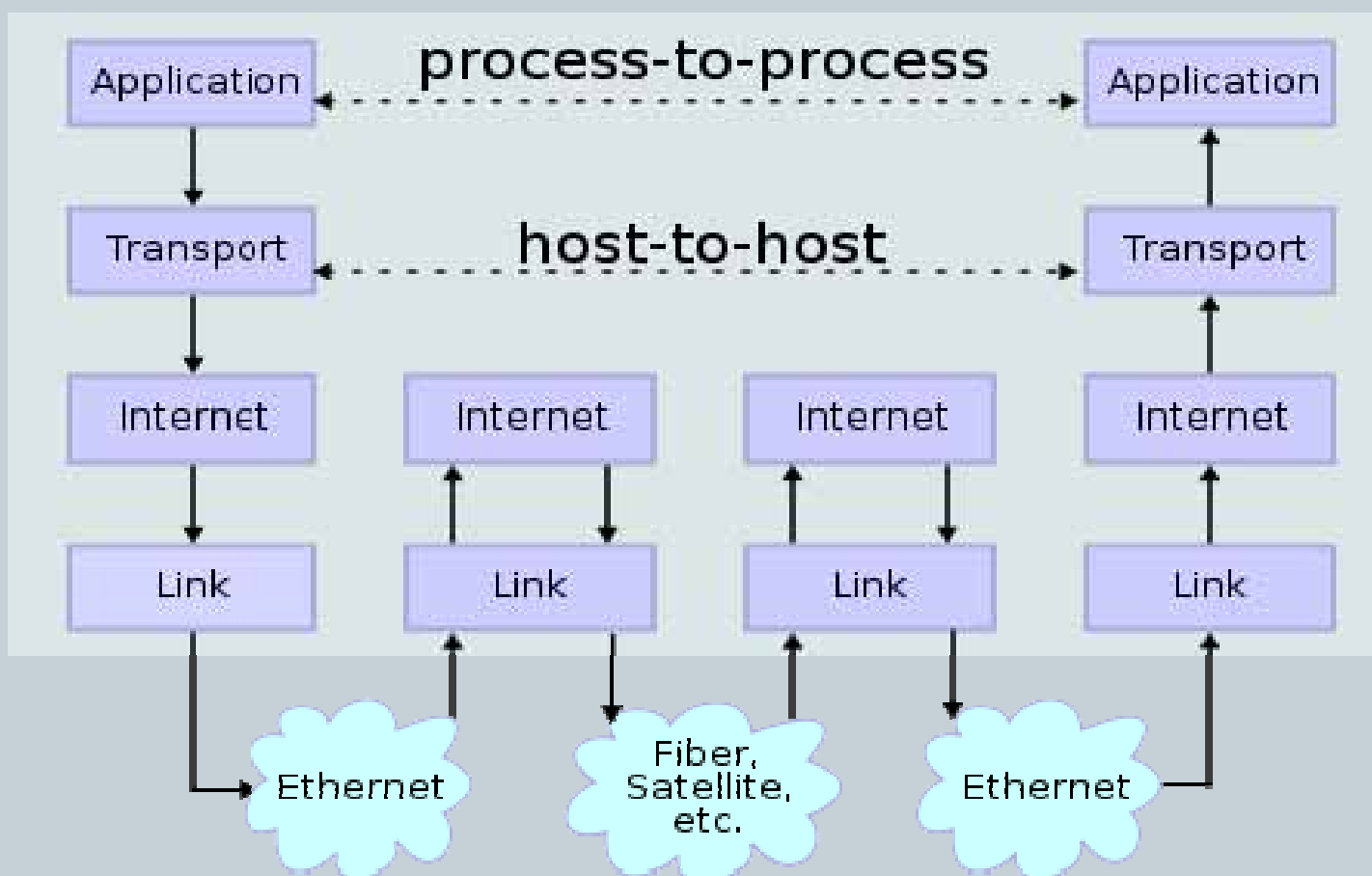
16

- It is used when you need to transfer packets as fast as possible and you can tolerate some levels of lost packets
- Packets might be lost and never reach the destination host
  - the source cannot be informed
- There is no flow and congestion control
- Examples: multimedia streaming



# Data Flow

17



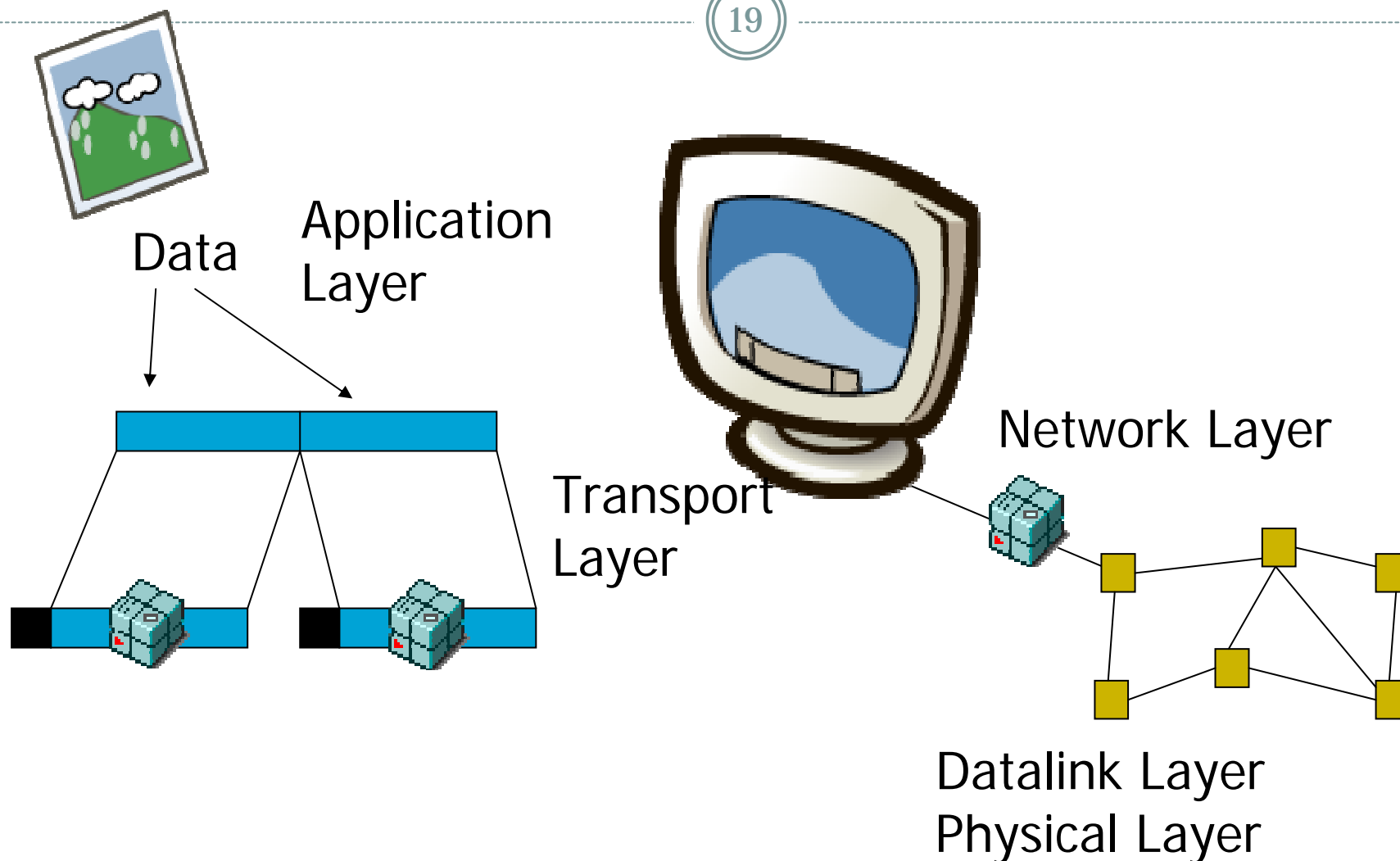
# Application Layer

18

- It is the highest layer in the Internet stack
- It deals with data processing made by the source and the destination hosts
- It knows only transport layer . It don't know how network is built.
- Example :
  - Email (SMTP,POP,IMAP)
  - Web browsing (HTTP)
  - File transfer (FTP)
  - streaming server/client

# What happens to data ?

19



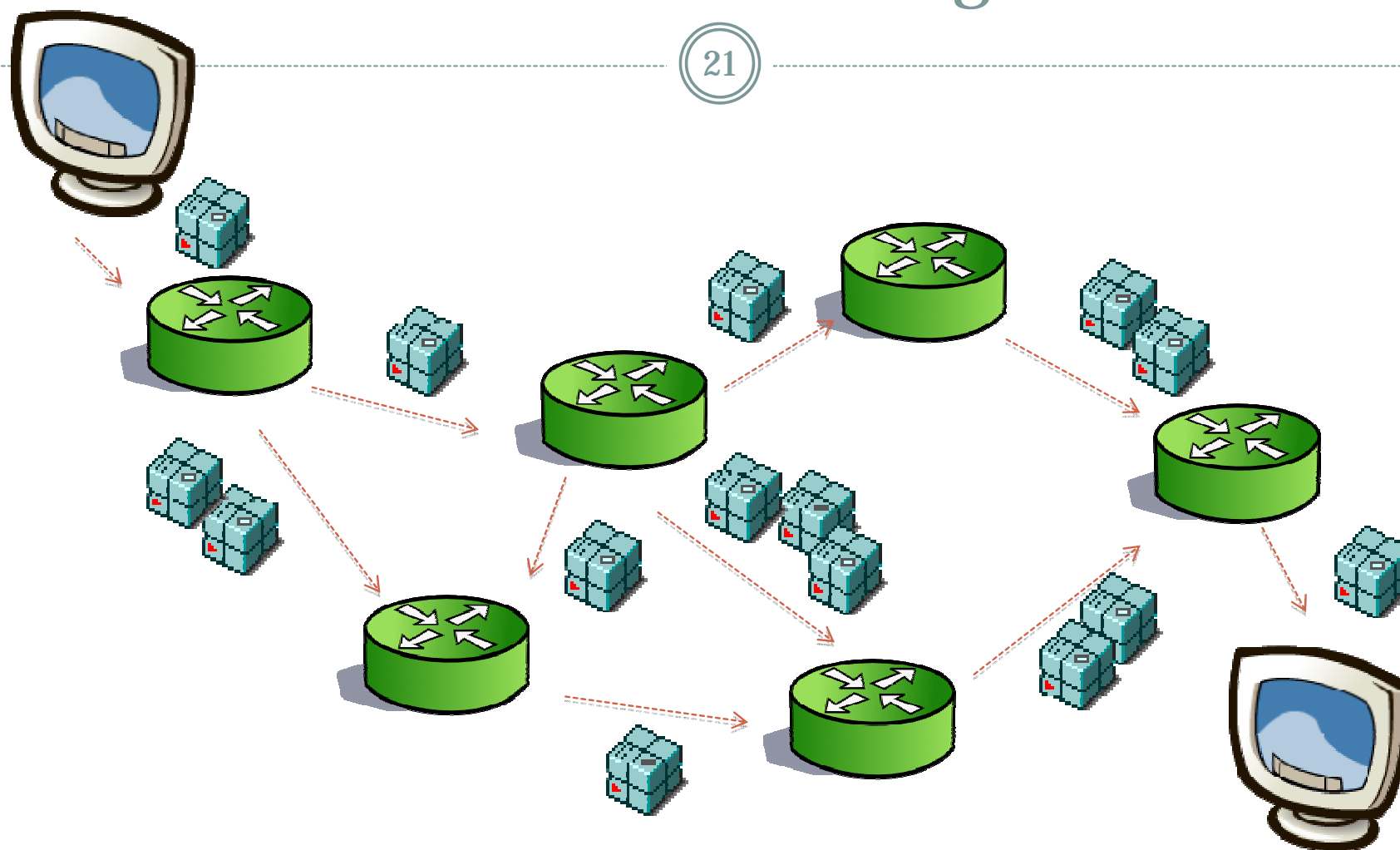
# Packet switching

20

- Mechanism used for sending data with Internet
- Each message is divided in many numbered packets
- Packets have a fixed length
- Each packet contains destination host computer address and source host computer address
- Each packet is sent separately and it can be forwarded following a completely different path one from the other

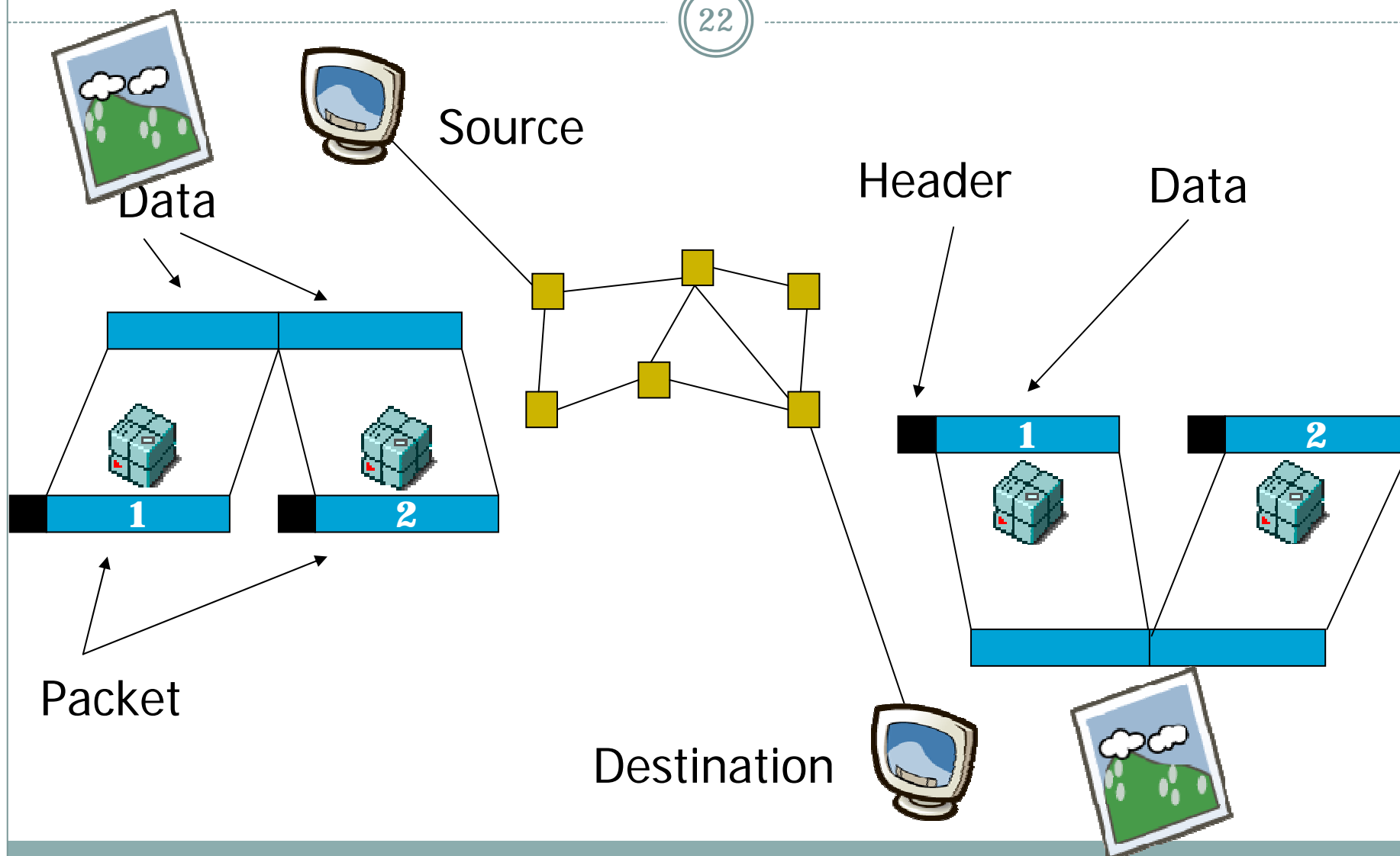
# Packet switching

21



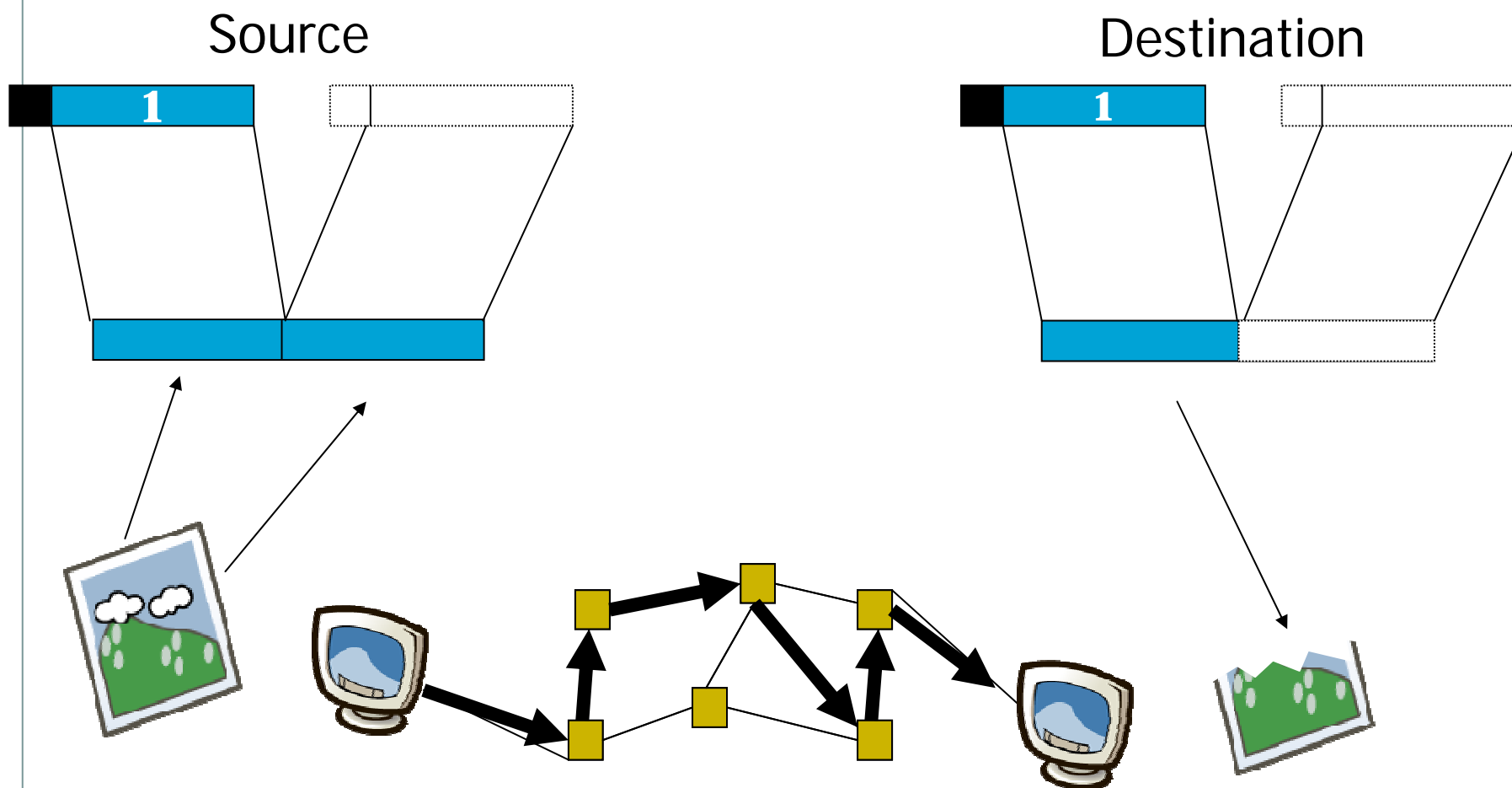
# Packet switching

22



# Packet switching

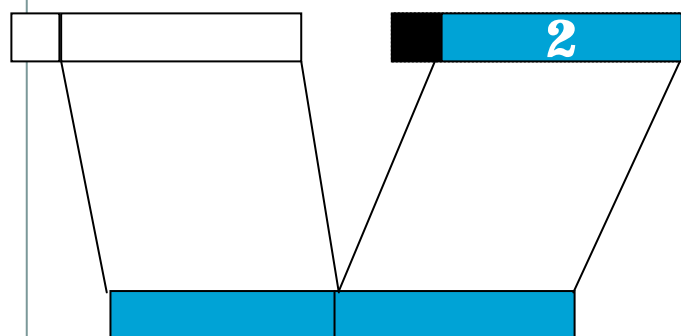
23



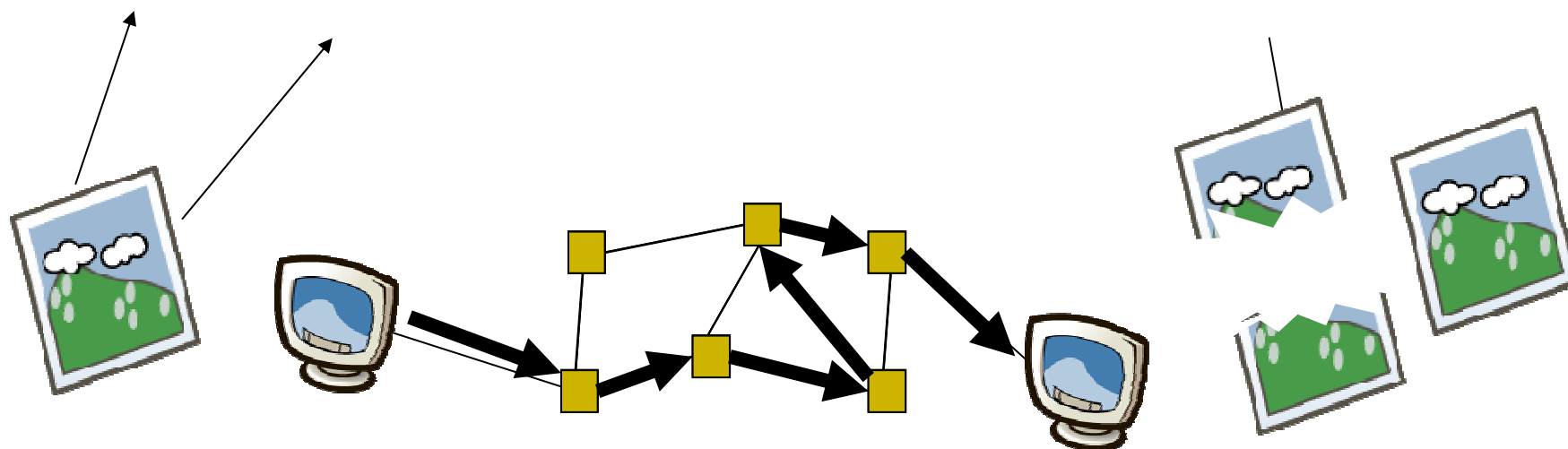
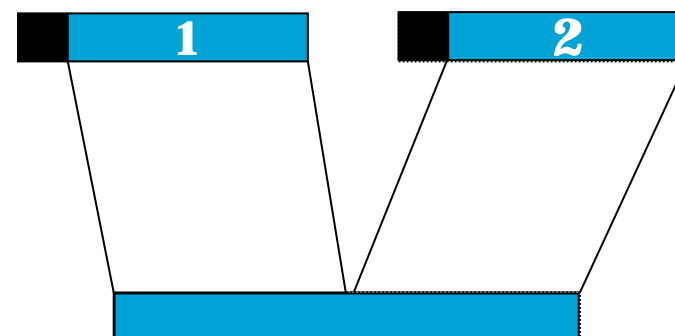
# Packet switching

24

Source



Destination



# Internet: a complex packet network

25

city-to-city connections

