



POLITECNICO
MILANO 1863



Fondamenti di Internet e Reti

Antonio Capone, Matteo Cesana,
Ilario Filippini, Guido Maier



POLITECNICO
MILANO 1863



1 – Introduzione e Architetture

**Antonio Capone, Matteo Cesana,
Ilario Filippini, Guido Maier**



POLITECNICO
MILANO 1863



1a – Introduzione al corso

**Scopo e finalità, testi e materiale,
esami, contatti**

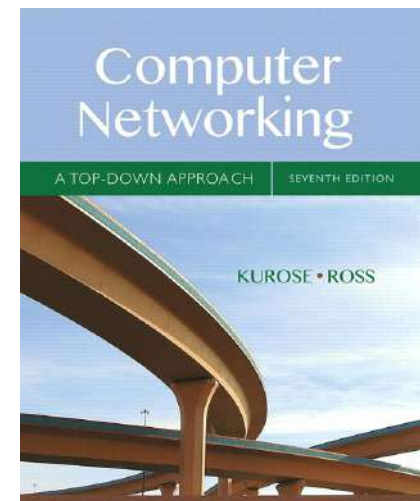
Docente

- **Prof. Antonio Capone**
- **Ufficio:**
 - DEIB, ed. 20, 3° piano
- **Tel: (02 2399) 3449**
- **E-mail: antonio.capone@polimi.it**
- **Web page: <http://home.deib.polimi.it/capone/>**
- **Orario di ricevimento:**
 - Martedì 14.00-15.30
- **Collaboratori per esercitazioni e laboratori:**
 - Ing. Alessandro Redondi,
 - Ing. Davide Sanvito



Materiale didattico

- **Testo di riferimento:**
James F. Kurose, Keith W. Ross
Reti di Calcolatori e Internet
Un approccio top-down
Settima Edizione (2017)
Pearson
- **Disponibile anche in inglese**
James F. Kurose, Keith W. Ross
Computer Networking
A top-down approach
Sixth Edition (2012)
Pearson



Materiale didattico

- **Altro materiale**
 - Slides delle lezioni
 - Appunti delle esercitazioni
 - Materiale a supporto delle attività di laboratorio
 - Video lezioni di supporto (canale YouTube)
 - Letture suggerite (link, articoli)
 - Tool per lezioni e laboratorio
- **Tutto il materiale è disponibile sulla pagina web del corso cui si accede dalla pagina personale del docente**



Home Page del corso

<http://www.antlab.polimi.it/teaching-capone/fondamenti-di-internet-e-reti>



Antonio Capone

Antonio's Home

Biography and Services

Publications

Teaching

- Wireless Networks
- Fondamenti di Internet e Reti**
- Lab Fond. di Internet e Reti
- Office hours
- Fundamentals of Communications Networks
- CISCO Networking Academy

Main Menu Login Form

ANTLab

Staff

Research

Projects

Antonio's Fondamenti di Internet e Reti



[Prof. Antonio Capone](#)

Ricevimento: Martedì 14.00-15.30 (controllare sempre [questa pagina](#) per aggiornamenti)

Anno accademico 2016-2017

1. [News](#)
2. [Programma ed orario](#)
3. [Materiale didattico per lezioni ed esercitazioni](#)
4. [Canale video del corso](#)
5. [Temi d'esame](#)
6. [Laboratorio e strumenti sw](#)
7. [Libri](#)
8. [Letture di approfondimento](#)

News:



Organizzazione del corso

- **Mix didattico:**
 - Lezione: 58 ore (circa)
 - Esercitazione: 24 ore (circa)
 - Laboratorio: 18 ore
- **Le esercitazioni sono inserite alla fine di ciascun argomento (non c'è una programmazione settimanale fissa)**
- **Il laboratorio si svolge in aula connessa senza PC, è necessario portare il proprio laptop (BYOD)**



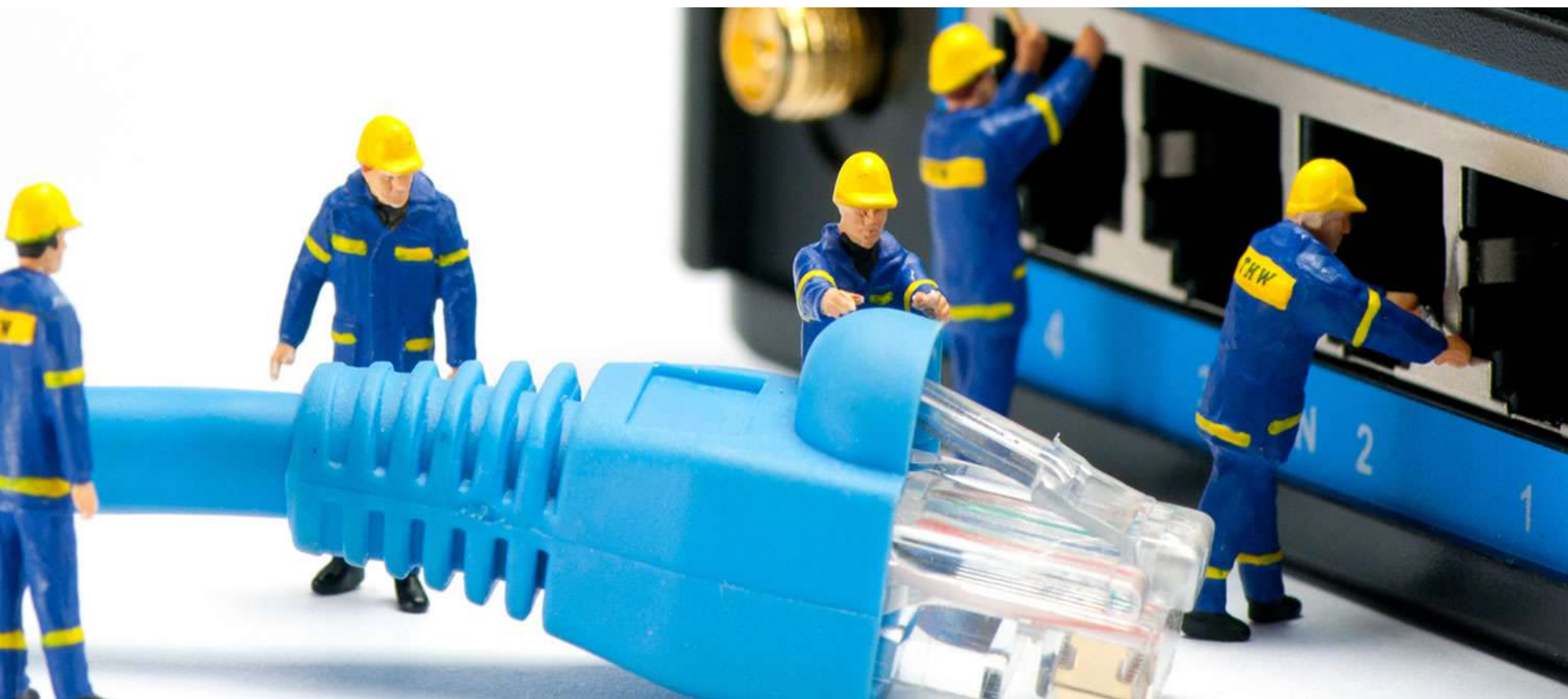
Modalità d'esame

- **Due prove in itinere**
 - Prova 1: 4 Maggio
 - Prova 2: fine corso (date da definire)
 - Durante prova 2, anche appello completo
- **Esame solo scritto**
 - 3 esercizi simili a quelli visti a lezione/esercitazione
 - 1 esercizio di laboratorio
 - Domande (risposta aperta e/o chiusa)
 - (di solito: 6 pt per esercizio, 8 pt domande, 6 pt laboratorio, totale 32 pt)
- **Test online durante il corso (kahoot.it)**
 - Punti extra per chi frequenta
 - Alla fine di ogni parte/capitolo in classe (usando PC, tablet, o smartphone)
 - 3 pt x frazione di risposte esatta su totale complessivo
 - 1 pt premio a chi vince la gara di ogni parte/capitolo
- **Orale solo a discrezione del docente (solo in caso estremi di comportamenti anomali)**



Scopo del corso

Iniziare a trasformarvi in esperti di Internet !



Scopo del corso

- **Fornire le conoscenze fondamentali sul funzionamento della rete tra calcolatori che oggi è alla base della quasi totalità dei servizi di comunicazione**
 - Audio e video telefonia
 - TV, video streaming, video on demand
 - Web, mobile apps, e accesso a DB in genere
 - Email
 - Messaging, chat
 - Social networks
 - Mappe e navigazione
 - Strumenti di collaborazione e condivisione
 - e molte altre ...
- 
- 



Scopo del corso

E in base a queste conoscenze imparare a:

Controllare l'interazione tra
le applicazioni e la rete

Configurare la rete e le sue
componenti

Prevedere il comportamento
della rete calcolandone le
prestazioni

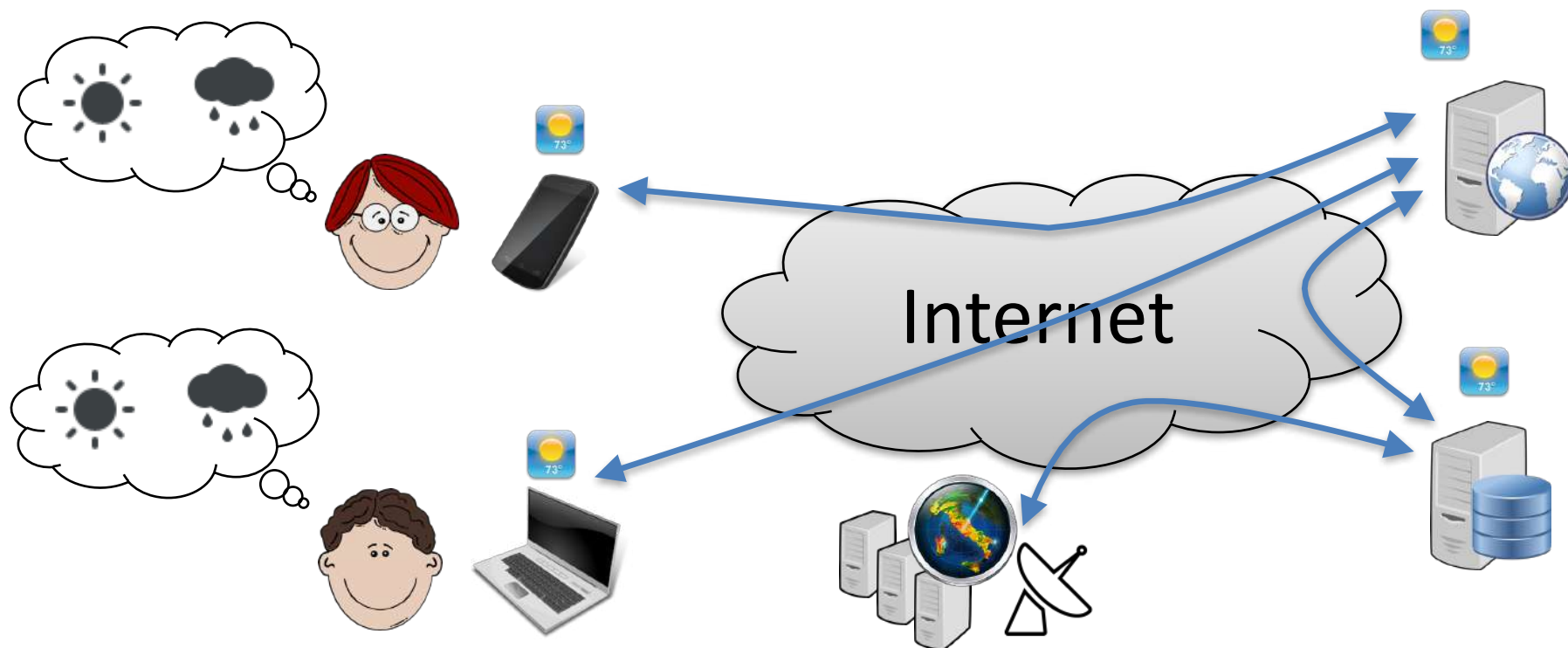
Monitorare il
comportamento del sistema
e risolvere i problemi

Essere pronti ai cambiamenti tecnologici della rete
(imparare ad imparare)



A cosa serve

- Tutti questi servizi del mondo della società dell'informazione sono **applicazioni distribuite**
- Le applicazioni distribuite sono programmi in esecuzione su più calcolatori che **scambiano dati tra loro usando Internet**



Vecchi mondi che non esistono più

Applicazioni isolate

- Elaborazioni isolate
- Scambio dati su rete

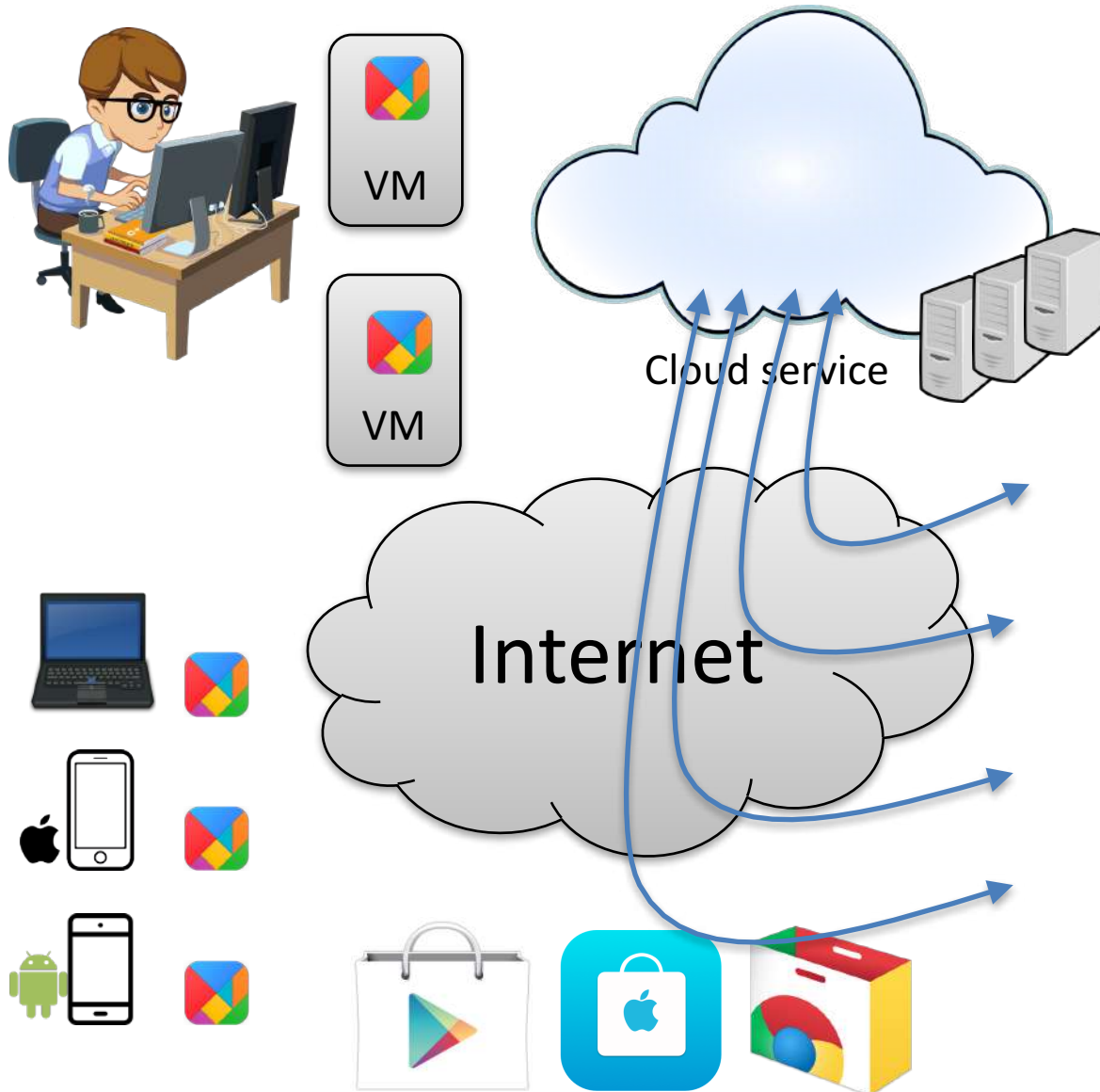


Reti di telecomunicazione

- Reti dedicate ai servizi
- Nessuna elaborazione



Cosa significa oggi sviluppare un'applicazione

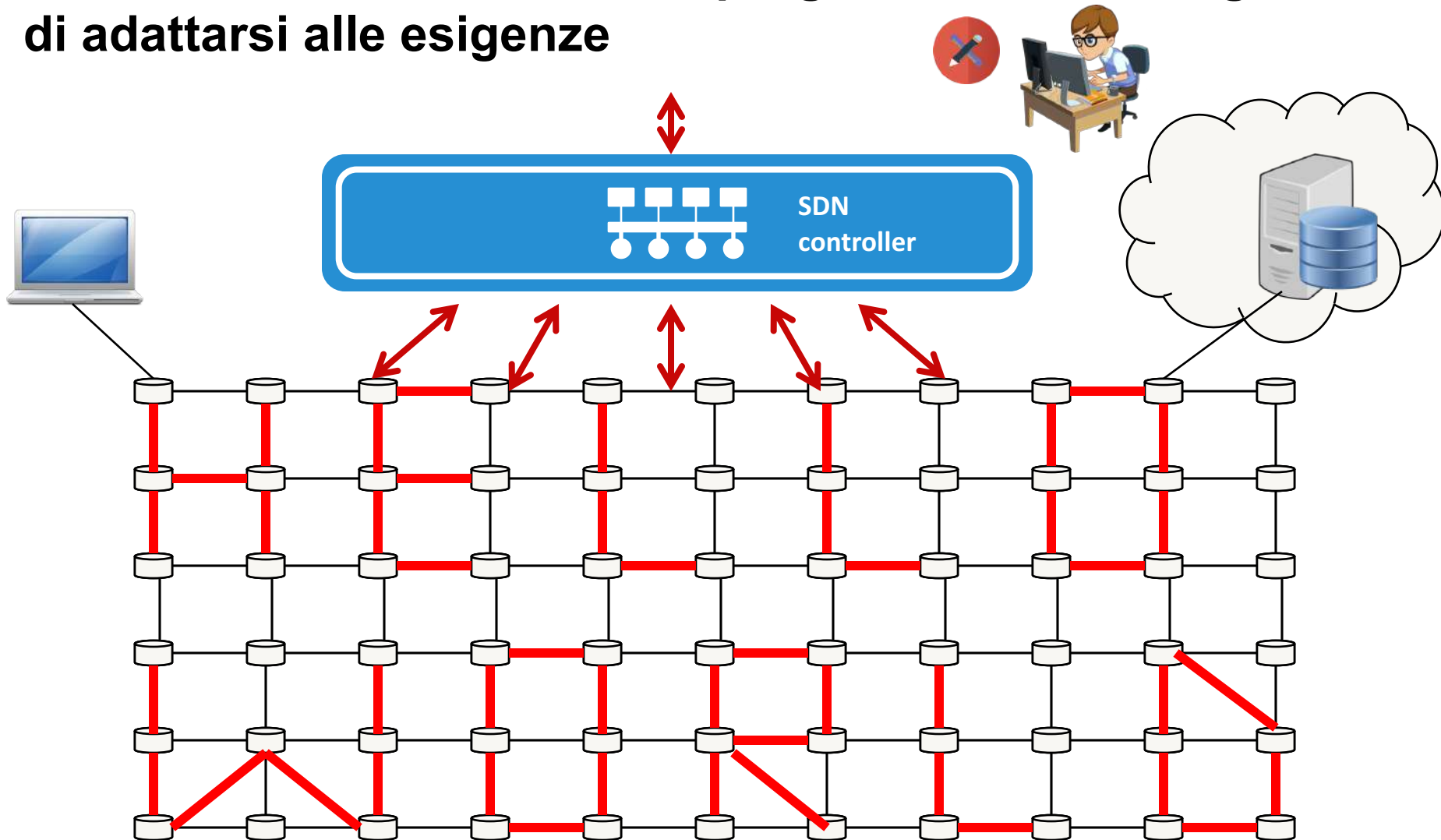


1. Scrivere le componenti server
2. Inserirle in macchine virtuali
3. Istanziarle su un servizio cloud
4. Scrivere le componenti client per diverse piattaforme
5. Inserirle nei relativi online store
6. Aspettare che gli utenti scarichino le applicazioni

Controllare la relazione con la rete è fondamentale

Applicazioni di rete: Software Defined Networking

Anche la rete sta diventando programmabile ed in grado di adattarsi alle esigenze



Le conoscenze per l'ingegneria dell'informazione

Le competenze richieste a chi:

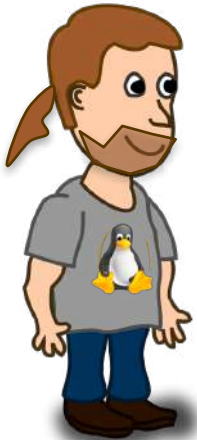


Sviluppa
applicazioni e
servizi

Costruisce la rete e
i suoi servizi di
comunicazione



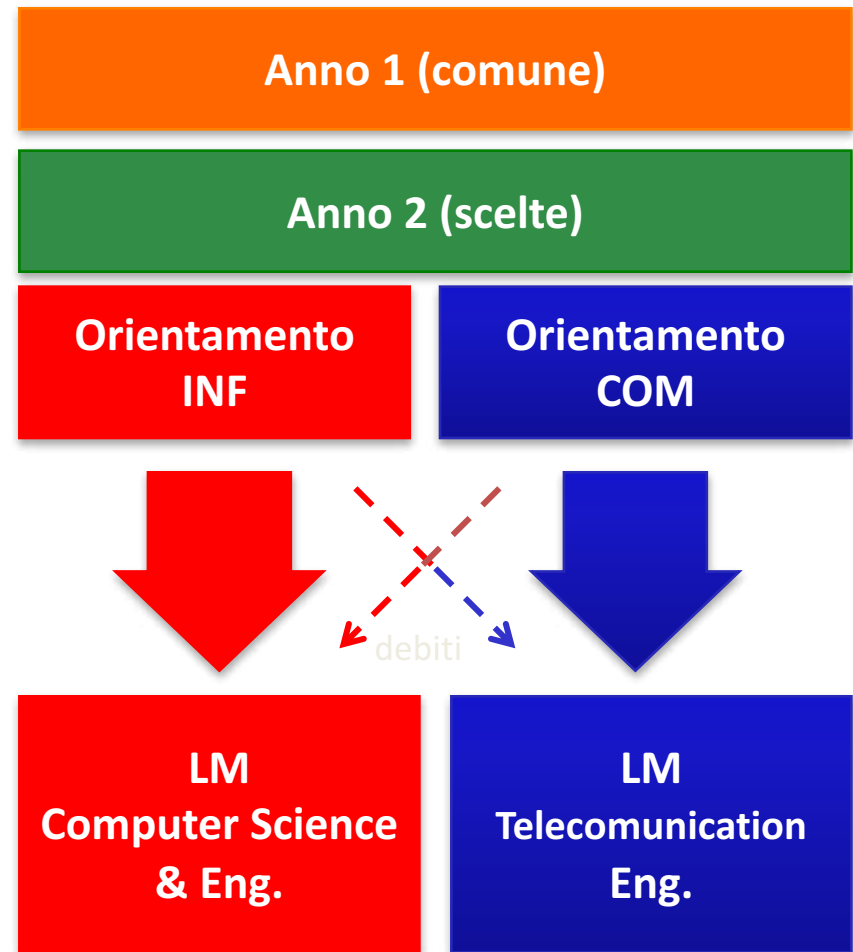
Sono sempre più integrate



Per qualunque ingegnere che opera nel settore dell'informazione è fondamentale sia saper sviluppare applicazioni e servizi su piattaforme diverse sia progettare e gestire la rete e i servizi di comunicazione

Percorso didattico in Informatica e Comunicazioni

- Come sapete i due percorsi didattici sono stati integrati
- E' possibile rimanere puri (**INF** o **COM**) ma sono state progettate delle ibridazioni
- E una serie di corsi che consentono una solida preparazione multidisciplinare
- Questo è il primo corso di area **COM** che vedete



Percorso didattico in Informatica e Comunicazioni

1° anno

INSEGNAMENTO	CREDITI
ANALISI MATEMATICA 1	10
GEOMETRIA E ALGEBRA LINEARE	8
FONDAMENTI DI INFORMATICA	10
FISICA	12
ELETTROTECNICA	9
FONDAMENTI DI INTERNET E RETI	10

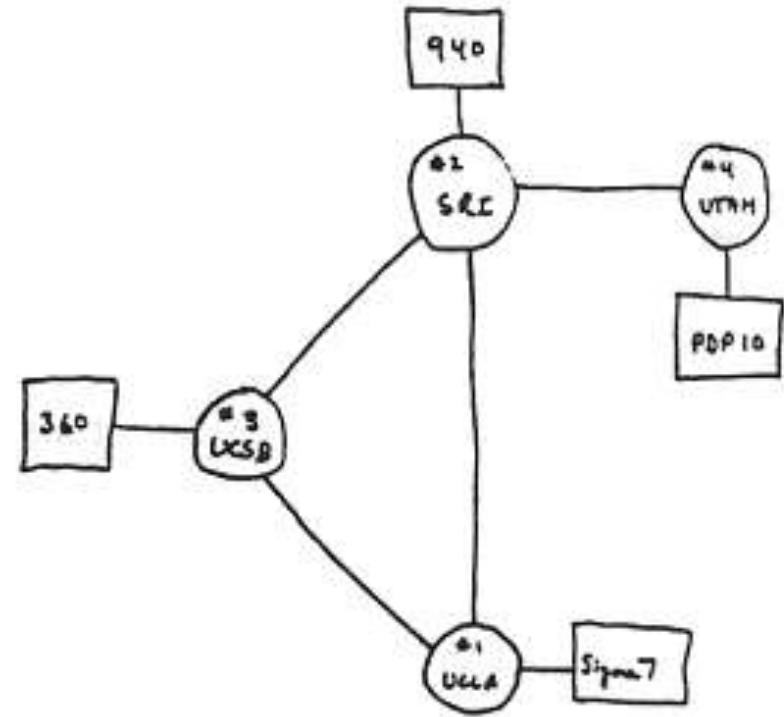
2° anno

INSEGNAMENTO	CREDITI
ANALISI MATEMATICA 2	10
ARCHITETTURA DEI CALCOLATORI E SISTEMI OPERATIVI	10
LOGICA E ALGEBRA	10
Insegnamento a scelta dal gruppo TABASE	
ELETTROMAGNETISMO E CAMPI	10
STATISTICA E CALCOLO DELLE PROBABILITA'	
TEORIA DEI FENOMENI ALEATORI E DELLA STIMA	11
ALGORITMI E PRINCIPI DELL'INFORMATICA	
LAB: ALGORITMI E PRINCIPI DELL'INFORMATICA	
SEGNALI PER LE COMUNICAZIONI	
LAB: SEGNALI PER LE COMUNICAZIONI	10
FONDAMENTI DI AUTOMATICA	

- Al 3° anno sono possibili dei recuperi per chi cambia idea
- Ci sono corsi che possono essere scelti nell'altra area; ad esempio se si sceglie INF in area comunicazioni ci sono:
 - SOFTWARE DEFINED NETWORKING (5 cr)
 - SICUREZZA DELLE RETI (5 cr)
 - OTTICA E IMMAGINI (5 cr)
 - ONDE ELETTROMAGNETICHE (5 cr)
 - SEGNALI PER LE COMUNICAZIONI (10 cr)



Brevissima storia di Internet



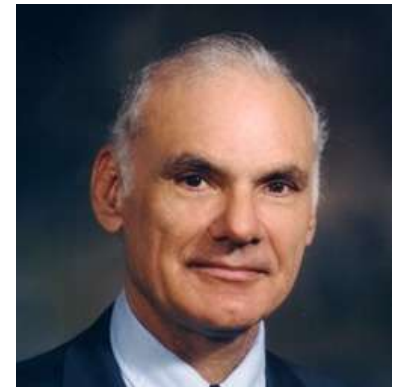
THE ARPA NETWORK

DEC 1969



Storia di Internet: anni '60

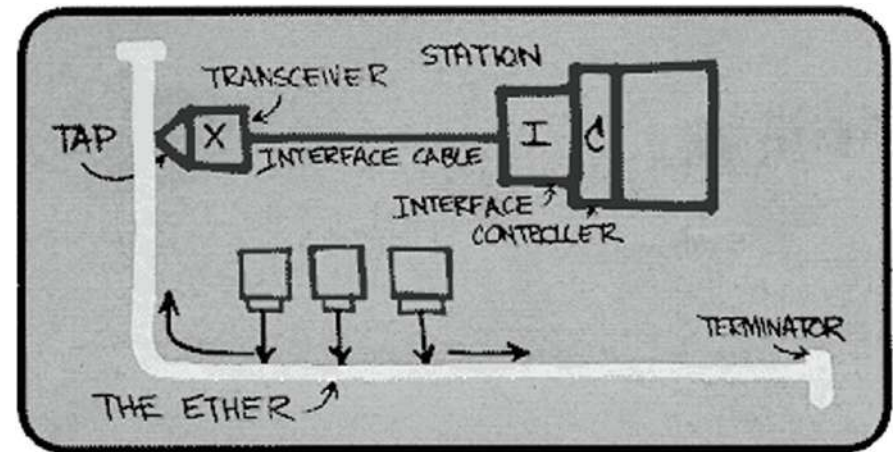
- **1961:** Kleinrock – dimostra l'efficacia della commutazione di pacchetto grazie alla teoria delle code
- **1967:** Lawrence Roberts progetta ARPAnet (Advanced Research Projects Agency)
- **1969:** primo nodo di IMP (Interface Message Processor) di ARPAnet a UCLA



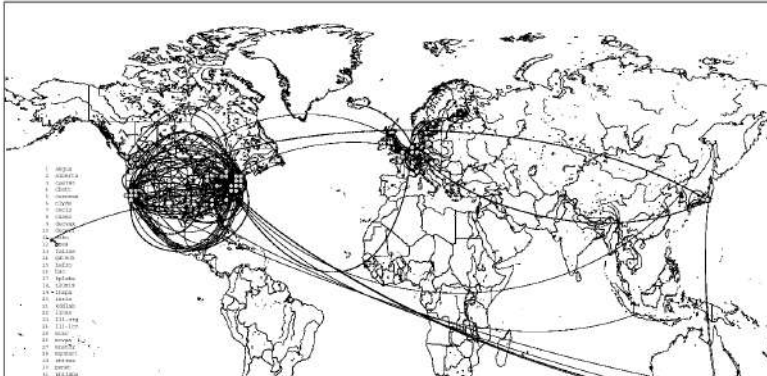
Storia di Internet: anni '70

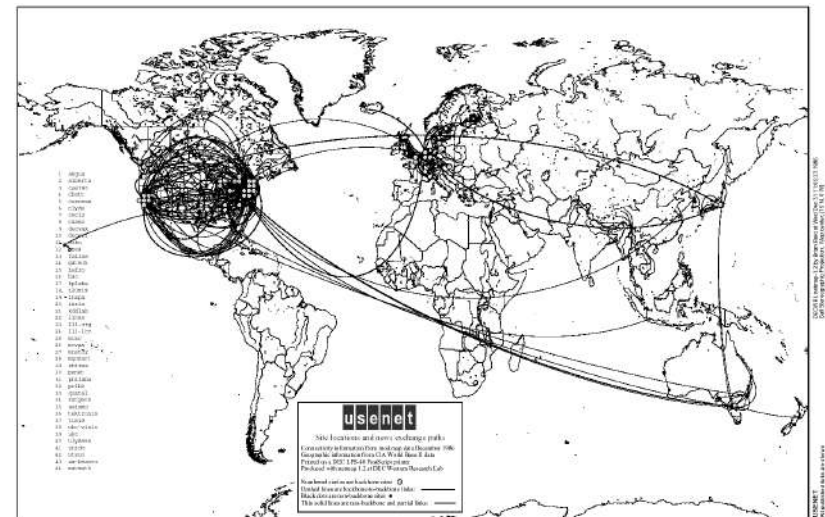
- **1972:**
 - Nasce NCP (Network Control Protocol) il primo protocollo di Internet
 - Primo programma per la posta elettronica
 - ARPAnet ha 15 nodi
- **1970:**
 - ALOHAnet rete radio a pacchetti al Univ. of Hawaii
- **1974:**
 - Cerf and Kahn – definiscono i principi dell'internetworking (rete di reti)

- **1976:**
 - Nasce Ethernet nei laboratori di Xerox
- **1979:**
 - ARPAnet ha 200 nodi



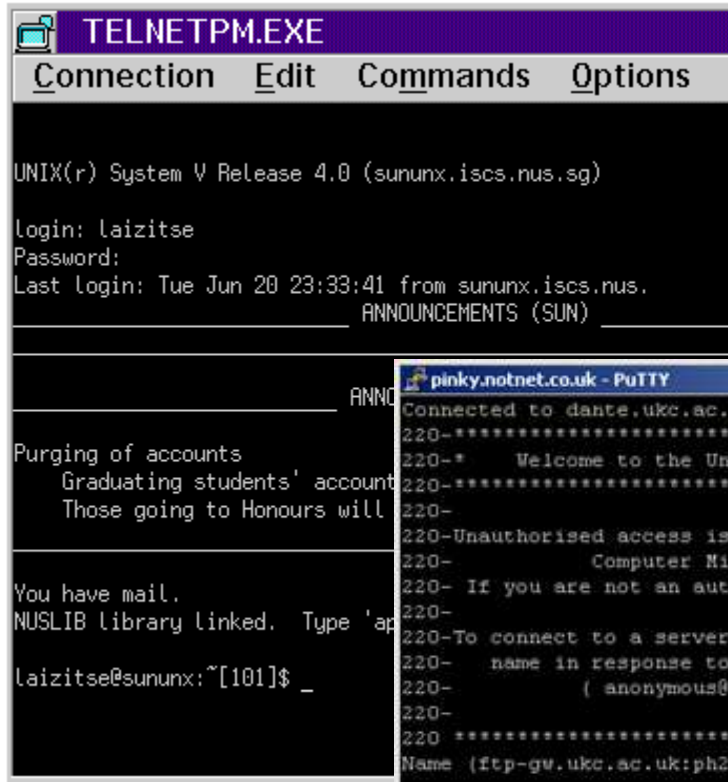
Storia di Internet: anni '80

- **1982:** definizione del protocollo SMTP per la posta elettronica
 - **1983:** rilascio di TCP/IP che sostituisce NCP
 - **1983:** definizione del DNS per la traduzione degli indirizzi IP
 - **1985:** definizione del protocollo FTP
 - **1988:** controllo della congestione TCP
- **Nuove reti nazionali: Csnet, BITnet, NSFnet, Minitel**
 - **100.000 host collegati**
- 
- The map illustrates the global reach of the early Internet. A dense web of connections is centered in North America, with lines extending across the Atlantic to Europe and across the Pacific to Asia. A legend on the left side of the map lists various network names and their corresponding numbers, such as 1. ARPANET, 2. NSFNET, 3. BITNET, 4. CSNET, 5. Minitel, 6. JANET, 7. EUnet, 8. GTE, 9. Telenor, 10. Telenor, 11. Telenor, 12. Telenor, 13. Telenor, 14. Telenor, 15. Telenor, 16. Telenor, 17. Telenor, 18. Telenor, 19. Telenor, 20. Telenor, 21. Telenor, 22. Telenor, 23. Telenor, 24. Telenor, 25. Telenor, 26. Telenor, 27. Telenor, 28. Telenor, 29. Telenor, 30. Telenor, 31. Telenor, 32. Telenor, 33. Telenor, 34. Telenor, 35. Telenor, 36. Telenor, 37. Telenor, 38. Telenor, 39. Telenor, 40. Telenor, 41. Telenor, 42. Telenor, 43. Telenor, 44. Telenor, 45. Telenor, 46. Telenor, 47. Telenor, 48. Telenor, 49. Telenor, 50. Telenor, 51. Telenor, 52. Telenor, 53. Telenor, 54. Telenor, 55. Telenor, 56. Telenor, 57. Telenor, 58. Telenor, 59. Telenor, 60. Telenor, 61. Telenor, 62. Telenor, 63. Telenor, 64. Telenor, 65. Telenor, 66. Telenor, 67. Telenor, 68. Telenor, 69. Telenor, 70. Telenor, 71. Telenor, 72. Telenor, 73. Telenor, 74. Telenor, 75. Telenor, 76. Telenor, 77. Telenor, 78. Telenor, 79. Telenor, 80. Telenor, 81. Telenor, 82. Telenor, 83. Telenor, 84. Telenor, 85. Telenor, 86. Telenor, 87. Telenor, 88. Telenor, 89. Telenor, 90. Telenor, 91. Telenor, 92. Telenor, 93. Telenor, 94. Telenor, 95. Telenor, 96. Telenor, 97. Telenor, 98. Telenor, 99. Telenor, 100. Telenor.



Storia di Internet: Le prime applicazioni

Telnet



```
TELNETPM.EXE
Connection Edit Commands Options

UNIX(r) System V Release 4.0 (sununx.iscs.nus.sg)

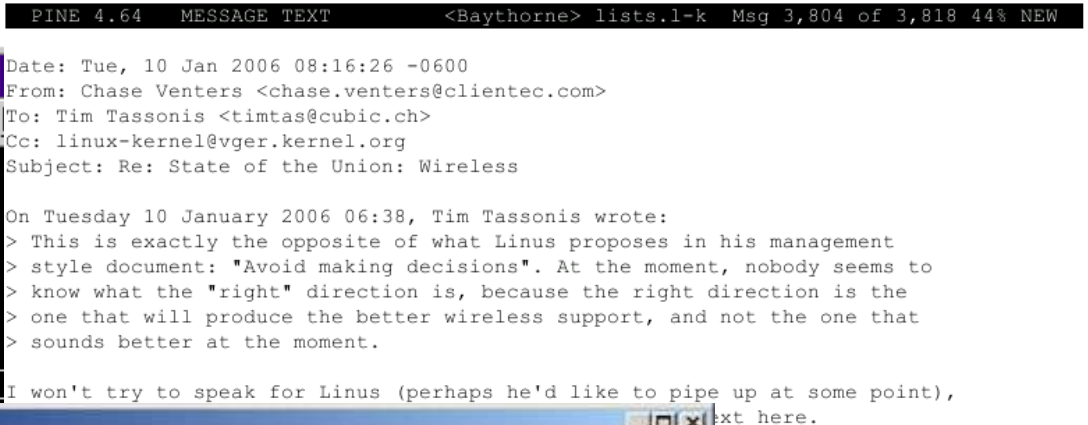
login: laizitse
Password:
Last login: Tue Jun 20 23:33:41 from sununx.iscs.nus.
ANNOUNCEMENTS (SUN)

Purging of accounts
  Graduating students' account
  Those going to Honours will

You have mail.
NUSLIB library linked. Type 'ap

laizitse@sununx:~[101]$ _
```

Email

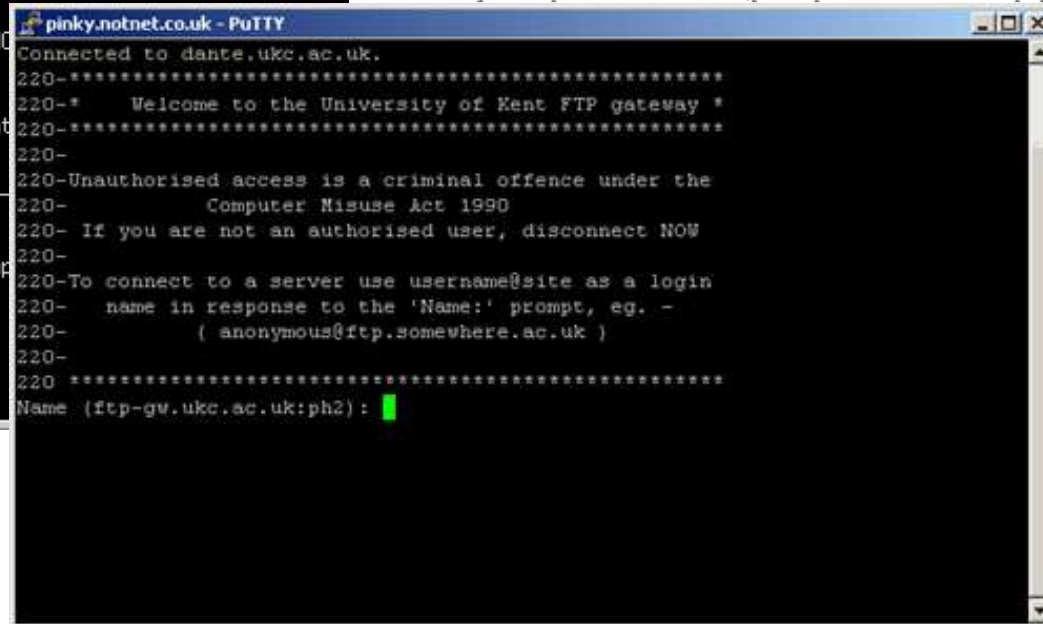


```
PINE 4.64 MESSAGE TEXT <Baythorne> lists.1-k Msg 3,804 of 3,818 44% NEW

Date: Tue, 10 Jan 2006 08:16:26 -0600
From: Chase Venters <chase.venters@clientec.com>
To: Tim Tassonis <timtas@cubic.ch>
Cc: linux-kernel@vger.kernel.org
Subject: Re: State of the Union: Wireless

On Tuesday 10 January 2006 06:38, Tim Tassonis wrote:
> This is exactly the opposite of what Linus proposes in his management
> style document: "Avoid making decisions". At the moment, nobody seems to
> know what the "right" direction is, because the right direction is the
> one that will produce the better wireless support, and not the one that
> sounds better at the moment.

I won't try to speak for Linus (perhaps he'd like to pipe up at some point),
text here.
```



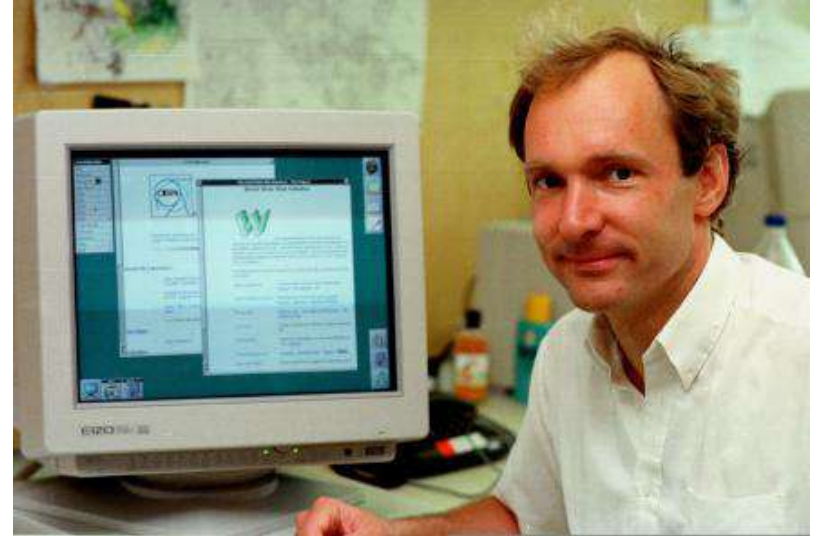
```
pinky.notnet.co.uk - PuTTY
Connected to dante.ukc.ac.uk.
220-*****
220-* Welcome to the University of Kent FTP gateway *
220-*****
220-Unauthorized access is a criminal offence under the
220- Computer Misuse Act 1990
220- If you are not an authorised user, disconnect NOW
220-
220-To connect to a server use username@site as a login
220- name in response to the 'Name:' prompt, eg. -
220- ( anonymous@ftp.somewhere.ac.uk )
220-
220 *****
Name (ftp-gw.ukc.ac.uk:ph2):
```

FTP



Storia di Internet: anni '90

- **1990:** ARPAnet viene dismessa
- **1991:** NSF lascia decadere le restrizioni sull'uso commerciale di NSFnet
- **Primi anni '90:** Tim Berners-Lee inventa il web al Cern di Ginevra
- **1994:** Mosaic, poi Netscape
- **Fine '90 :** commercializzazione del Web



Storia di Internet: anni '00

2000 – 2009:

- Arrivano le “killer applications”:
messaggistica istantanea, condivisione di file P2P, IP Telephony, social networks
- La sicurezza di rete diventa un problema
- Centinaia di milioni di host, un miliardo di utenti
- Velocità nelle dorsali dell'ordine dei Gbps

You Tube

last.fm

flickr

skype



Diffie-Hellman-Merkle



Shawn Fanning



KoZaA

BitTorrent



Zuckerberg

twitter



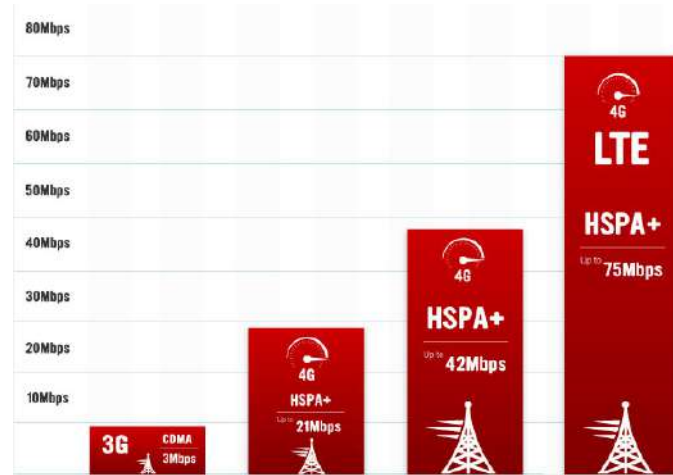
myspace.com



Storia di Internet: anni '10

2010 – oggi:

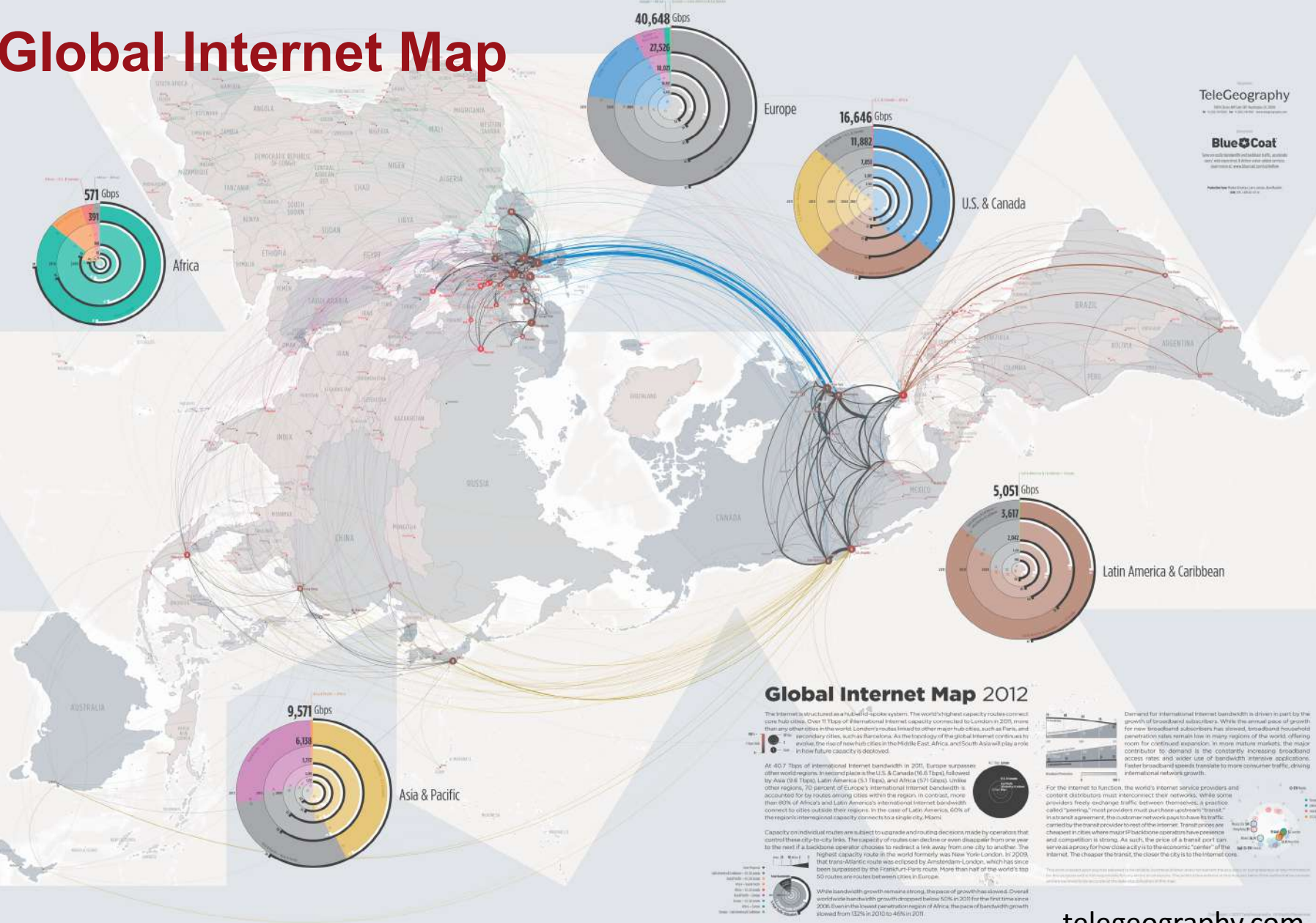
- Esplosione della *Mobile Internet*
- Arrivano gli smart-phone
- La telefonia si trasferisce definitivamente su Internet
- I contenuti video diventano il traffico predominante sulla rete



NETFLIX



Global Internet Map



TeleGeography

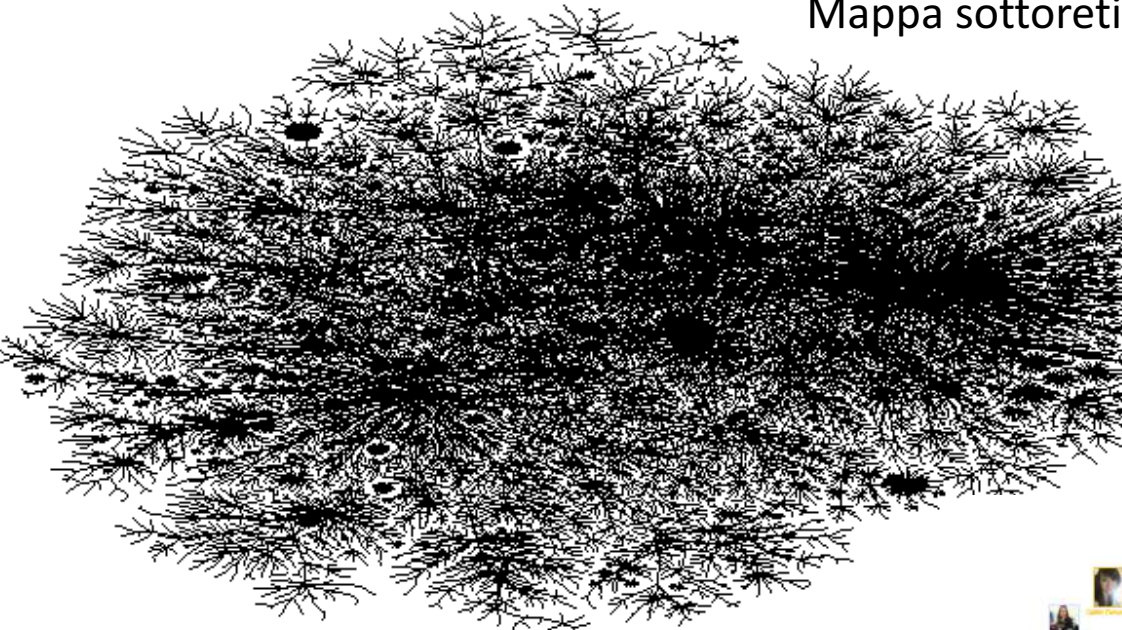
Blue Coat

telegeography.com

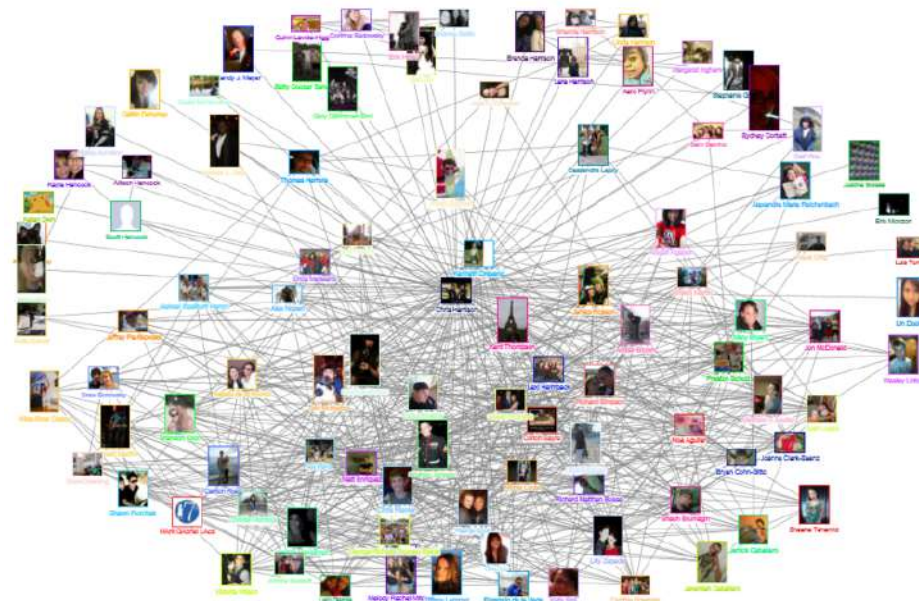


Il mondo è piccolo

Mappa sottoreti IP (2008)

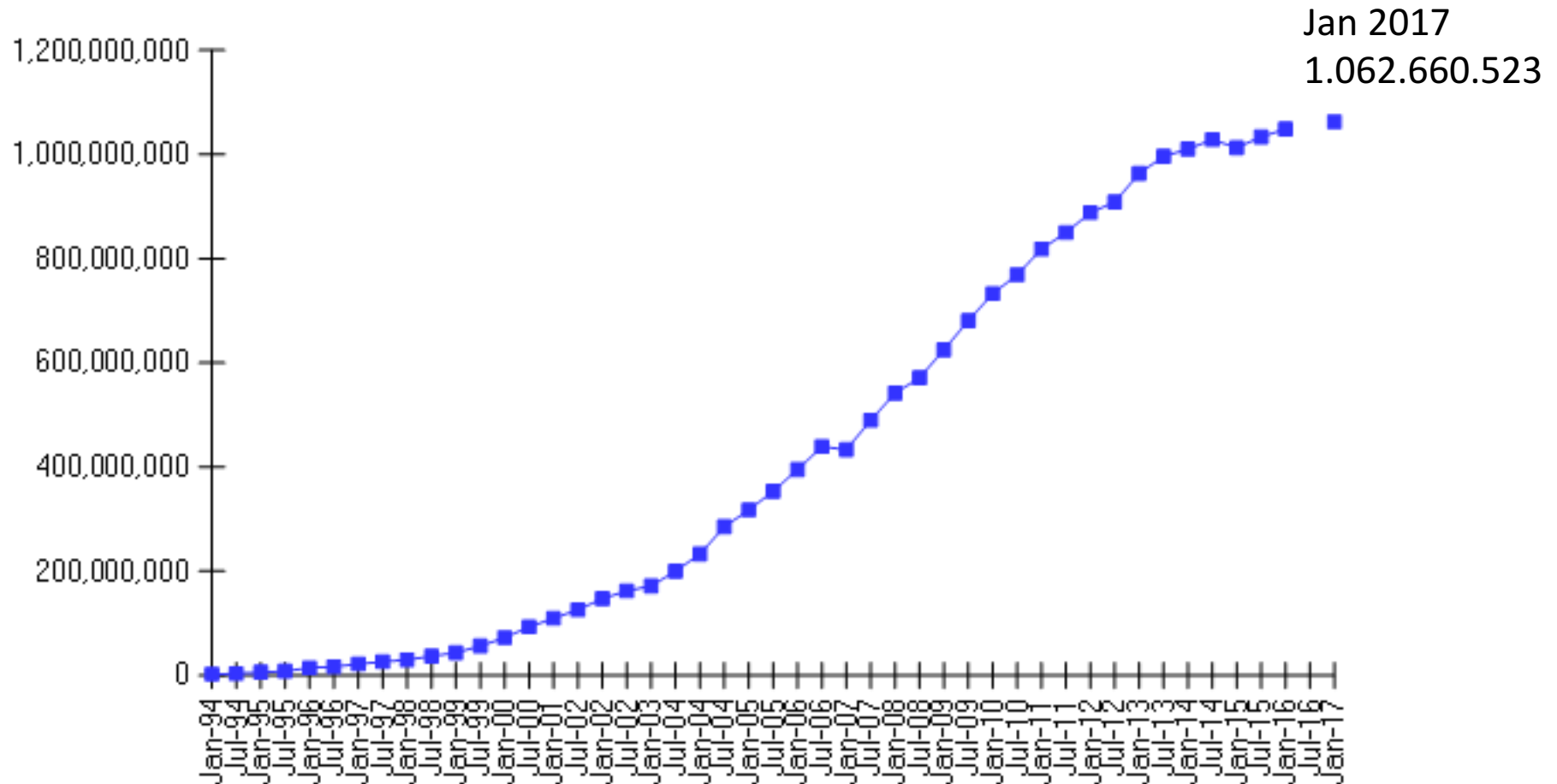


Esempio di topologia di
social network



La crescita di internet

Internet Domain Survey Host Count

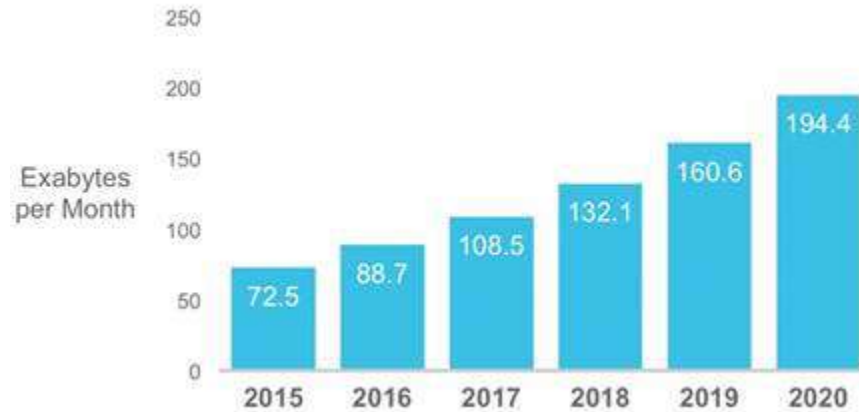


isc.org

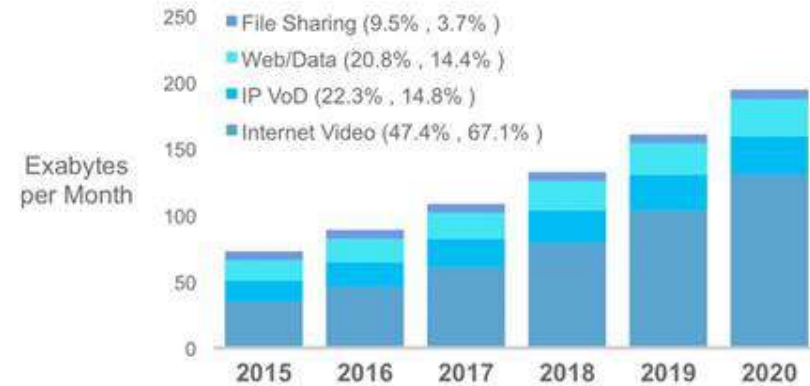
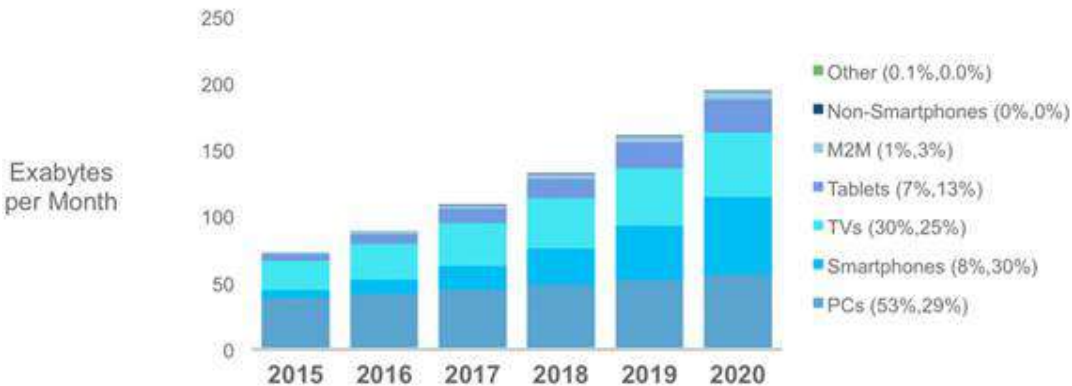
Source: Internet Systems Consortium (www.isc.org)



Il traffico di internet



(exa=10¹⁸)



CISCO VNI
(2016)



Internet è nel ~~CLOUD~~ MARE



99% del
traffico
internazionale
passa
attraverso cavi
sottomarini

telegeography.com



Internet NON è un posto sicuro



Programma del corso

1. INTRODUZIONE E ARCHITETTURE

- Principi generali, architettura e componenti, meccanismi di base

2. SISTEMI DI COMUNICAZIONE

- Come misuro le prestazioni di una rete: il concetto di *throughput*, i ritardi nelle reti di telecomunicazioni (tempo di trasmissione, ritardo di trasferimento, tempo di processing, tempo di accodamento).

3. MODELLI FUNZIONALI

- Come è gestita la comunicazione in rete: il concetto di protocollo di comunicazione, modelli architetturali a livelli, commutazione di pacchetto e commutazione di circuito



Programma del corso

4. PROTOCOLLI APPLICATIVI

- architetture delle applicazioni di rete: approccio *client-server* ed approccio *peer-to-peer*;
- esempi di protocolli applicativi *client-server*: HTTP, FTP, SMTP;
- architetture *peer-to-peer*: la rete Gnutella, BitTorrent

5. IL LIVELLO DI TRASPORTO

- caratterizzazione del servizio di comunicazione tra processi applicativi; trasporto non affidabile: il protocollo UDP (formato dei segmenti);
- trasporto affidabile: il protocollo TCP (formato dei segmenti, apertura della connessione, controllo di flusso, controllo di congestione e controllo d'errore).



Programma del corso

6. IL LIVELLO DI NETWORKING:

- *l'Internet Protocol (IP)*: servizi offerti da IP, formato dei pacchetti IPv4
- Gestione di indirizzi IP: formati e notazioni degli indirizzi IPv4, le classi e gli indirizzi speciali, pianificazione di uno spazio di indirizzamento IPv4, tecniche di *subnetting* e *supernetting*, assegnamento automatico di indirizzi IP: il Dynamic Host Control Protocol (DHCP)
- Corrispondenza tra indirizzi IP ed indirizzi simbolici (Il *Domain Name System*);
- *l'Internet Control Message Protocol (ICMP)*



Programma del corso

7. INOLTRO ED INSTRADAMENTO IN INTERNET:

- inoltro diretto ed indiretto
- uso delle tabelle di *routing*;
- instradamento su cammini minimi, la costruzione dell'albero dei cammini minimi,
- Instradamento link state ed instradamento distance vector
- Esempi di protocolli: RIP, OSPF, BGP,



Programma del corso

8. RETI LOCALI E LIVELLO DI LINEA:

- problema dell'accesso multiplo,
- indirizzamento nelle reti locali,
- *l'Address Resolution Protocol* (ARP),
- interconnessione di reti locali con bridge/switch,
- lo standard Ethernet/802.3: principi e funzionamento,
- lo standard IEEE 802.11 (WiFi): principi e funzionamento.

9. INTRANET

- Indirizzamento privato e traduzioni di indirizzi IP (NAT, NAPT)
- Interconnessione di reti private (*IP Tunneling*)
- IPv6 (cenni)

10. LIVELLO FISICO

- Come viaggia in rete l'informazione: cenni sulla caratterizzazione dei segnali (il concetto di banda del segnale, i segnali numerici)
- Dove viaggia l'informazione in rete: cenni sulla caratterizzazione dei mezzi trasmissivi (la banda di canale, ritardo di trasferimento, il concetto di capacità di canale)



Programma del corso

LABORATORIO:

- Attività di base svolta a lezione ed esercitazione:
 - Sniffer di rete (Wireshark)
 - Ping (PingPlotter), Traceroute, Dig, Strumenti del browser (chrome)
 - Protocolli applicativi (con utilizzo di server e client email e web)
- Attività di laboratorio
 - Lab 1: Python e scripting per analisi di rete
 - Lab 2: Programmazione socket in Python, parte I
 - Lab 3: Programmazione socket in Python, parte II
 - Lab 4: Configurazione e simulazione di rete (Packet Tracer), parte I
 - Lab 5: Configurazione e simulazione di rete (Packet Tracer), parte II
 - Lab 6: Attività sperimentali su reti wireless



Programma CISCO Networking Academy

- Il Politecnico offre agli studenti la possibilità di seguire online e gratuitamente i corsi di formazione per la certificazione CISCO CCNA
- Il programma CISCO NA è uno strumento aggiuntivo e facoltativo offerto per integrare la preparazione e acquisire competenze anche di tipo implementativo che non vengono normalmente fornite nei corsi
- Il programma CCNA si articola in 4 moduli/semestri
 - CCNA 1, CCNA 2, CCNA 3, CCNA 4



Come si segue il programma CCNA ?

- Il programma CCNA si segue mediante un piattaforma di E-learning
- Ogni studente studia autonomamente usando la piattaforma
- I docenti dei corsi di reti e il responsabile del programma CISCO (Prof. Antonio Capone) sono a disposizione per chiarimenti durante l'orario di ricevimento
- Seguire il programma CCNA richiede dell'impegno aggiuntivo rispetto al normale impegno per i corsi universitari
- A fronte di questo impegno si riceve una preparazione complementare a quella universitaria molto apprezzata nel mercato del lavoro



Piattaforma di E-Learning

- **Esempi della piattaforma:**
 - Portale d'ingresso: NetSpace

The screenshot displays the Cisco Networking Academy NetSpace portal. At the top, the Cisco logo and 'Cisco Networking Academy' text are on the left, while a user greeting 'Welcome, Antonio Capone, Sign Out | Help' and a search bar are on the right. A navigation bar below contains links: 'NetSpace Home', 'About Us', 'Program', 'Offerings', and 'Communities'. The main content area features a 'Scheduled Site Maintenance' alert, a 'My Tasks' section, and a 'Learn' tab. Under the 'Learn' tab, there's a 'Teach' and 'Manage' section. A large image of a smiling man at a computer is accompanied by the text 'Access the Courses You Are Taking' and a description. To the right, there's a 'Security in the Cloud Webinar' section with a 'Register Here' link, and a 'Self-Enroll Made Easy' section with a 'More Details' link and a 'Learn' button.

Welcome, Antonio Capone, Sign Out | Help

cisco Cisco Networking Academy Mind Wide Open

NetSpace Home About Us Program Offerings Communities

Scheduled Site Maintenance [Show](#)

My Tasks
Your classroom related tasks will display here.

Security in the Cloud Webinar
Learn about the security risks and threats in cloud, and other topics such as Identity and Access Management & data security
[Register Here](#)

Self-Enroll Made Easy
Explore the revamped Course Catalog & Self-Paced Courses, under Offerings menu, choose "Enroll Now" and start learning.
[More Details](#)

Teach Manage Learn

Access the Courses You Are Taking
When you are enrolled in courses you will be able to access them from this tab.

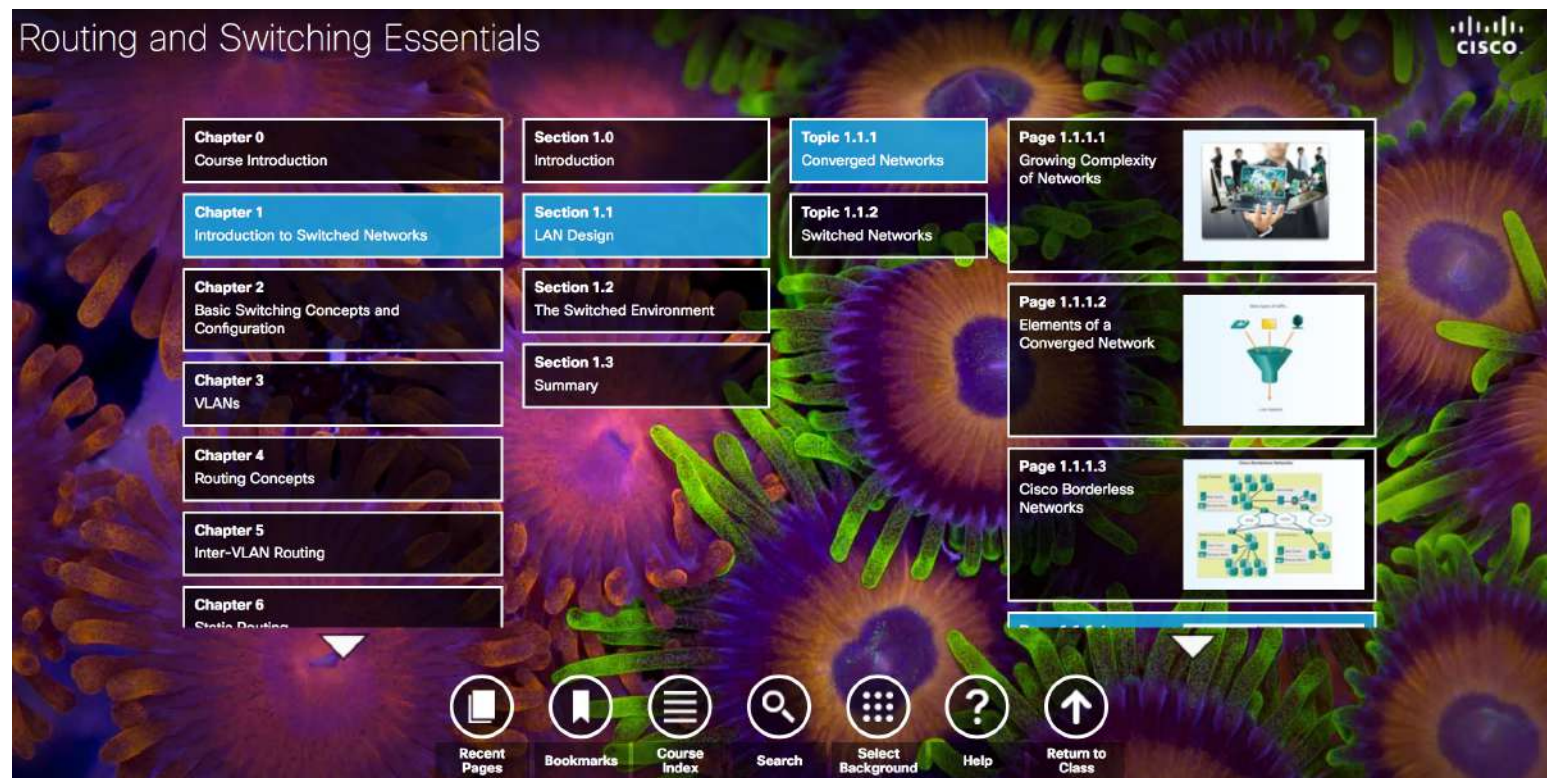
Actions
[View Course History](#)



Piattaforma di E-Learning

- Esempi della piattaforma:
 - Esempio (1)

NOTA: il materiale è in inglese



Come si fanno gli esami CCNA ?

- Gli esami si fanno sempre mediante un piattaforma di E-learning
- Gli esami consistono in quiz e attività interattive
- Ogni semestre prevede un esame finale
- Il giorno dell'esame lo studente ha tempo dalle 9 alle 17 per completare i quiz
- Normalmente alla fine di ogni semestre sono fissate 4-5 date per l'esame
- Ogni "assessment" ha un tempo massimo dopo l'attivazione
- Copiare o barare agli esami non conviene, conta solo quello che si impara



Come ci si iscrive al CCNA?

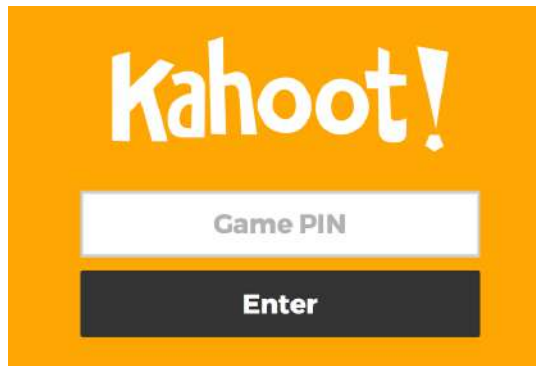
- ▶ Per iscrizioni e maggiori informazioni consultare la pagina del POLIMI: <http://home.deib.polimi.it/capone>
 - > teaching
 - > CISCO NA
- ▶ ISCRIZIONI:
 - ▶ 2 semestre (mar-apr)
- ▶ Dopo l'iscrizione (siete pazienti viene fatta a mano) riceverete un email con login e password
- ▶ Usateli per effettuare il primo accesso al portale
- ▶ Dopo il primo accesso cambiate login (preferibilmente in nome.cognome) e password e inserite tutte le informazioni personali



Quanto ne sapete di Internet

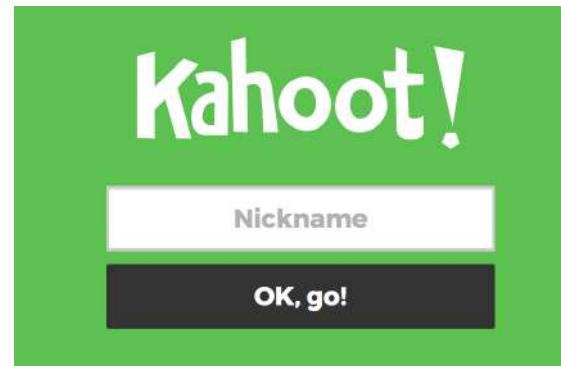
- **Test d'ingresso: (no punti premio)**
 - Quiz gara
 - Usate il vostro smartphone/tablet/PC collegato a Internet
 - Collegativi con il browser a:

Inserite il PIN che vi darò tra un attimo

The image shows the Kahoot! login interface for entering a Game PIN. It features the Kahoot! logo in white on an orange background. Below the logo is a white input field labeled "Game PIN" and a black button labeled "Enter".

kahoot.it

Inserite come nickname
Cognome_Nome

The image shows the Kahoot! login interface for entering a nickname. It features the Kahoot! logo in white on a green background. Below the logo is a white input field labeled "Nickname" and a black button labeled "OK, go!".



POLITECNICO
MILANO 1863

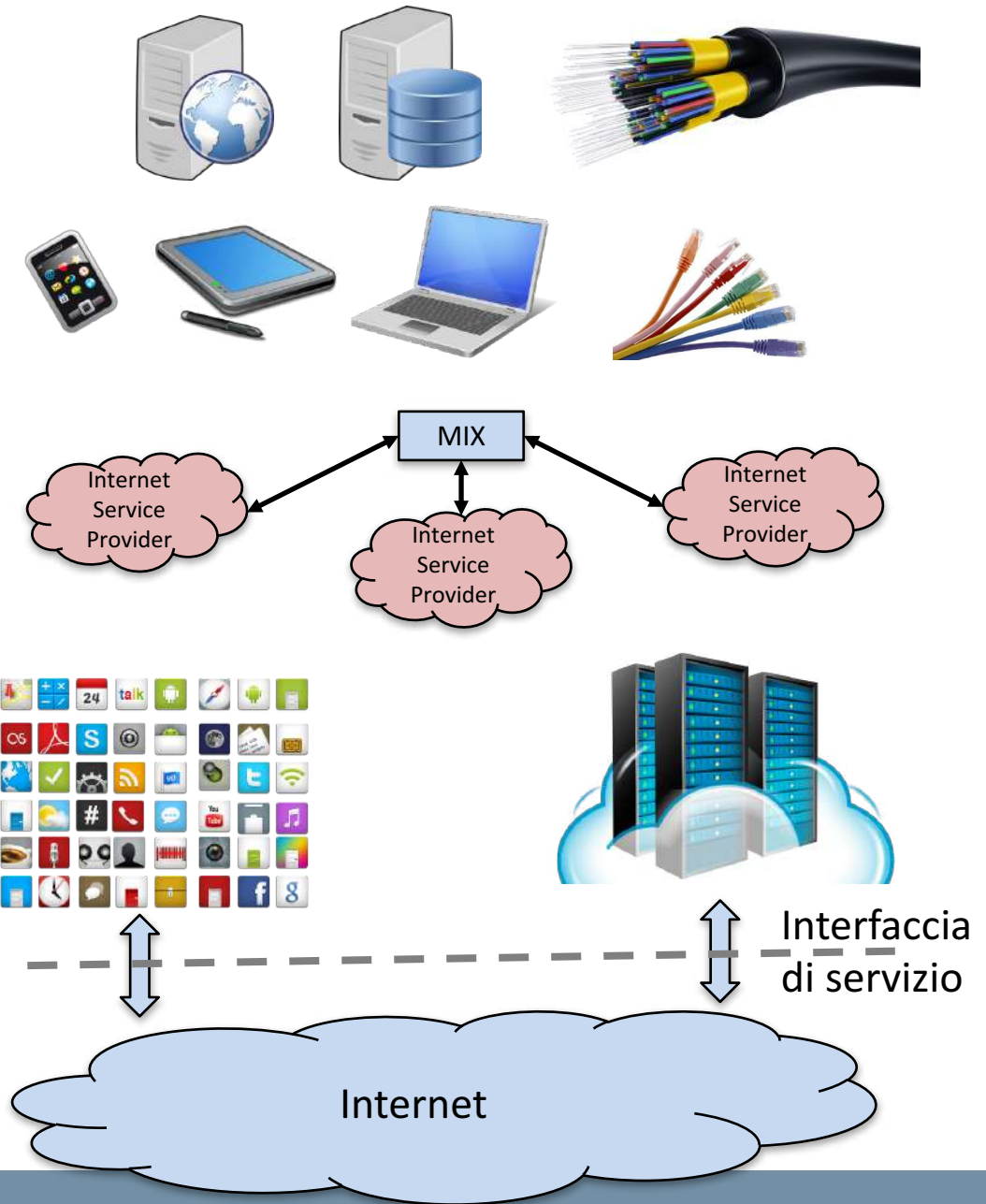


1b – Concetti base

**Cos'è Internet, Architettura e
componenti, Meccanismi base**

Cos'è Internet?

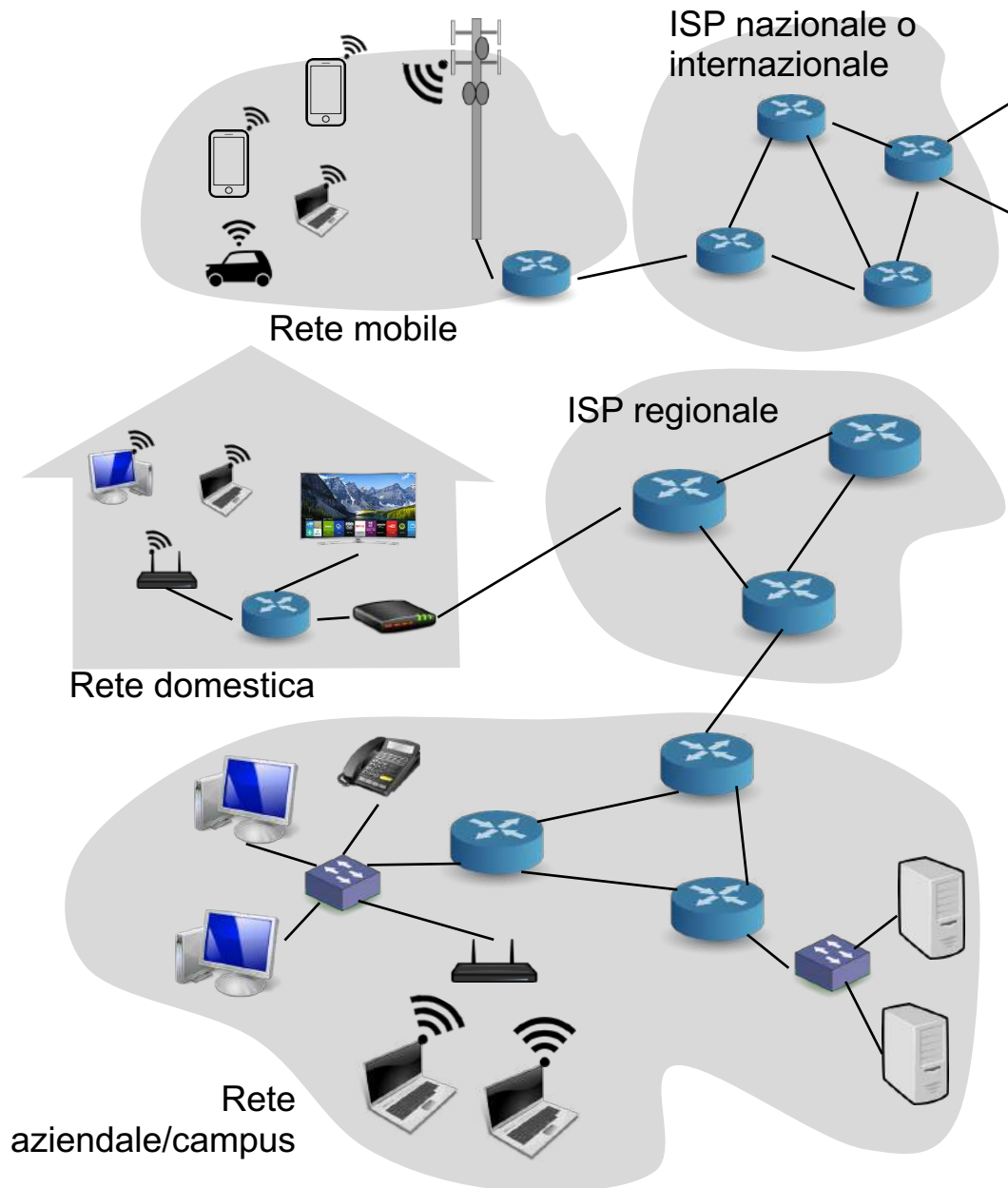
- 1) Una **infrastruttura fisica** fatta di componenti
- 2) Un'**architettura di rete**
- 3) Un **servizio di comunicazione** usato dalle applicazioni e i **protocolli di comunicazione** tra le componenti del sistema



Cos'è Internet?

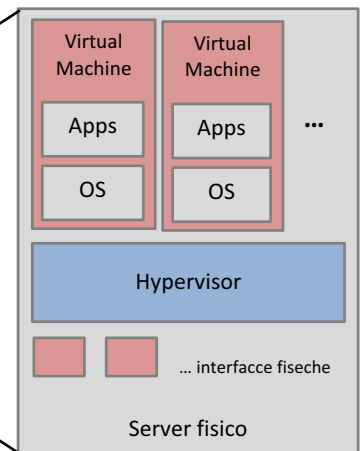
Componenti fisiche

- Milioni di computer connessi alla rete chiamati **host** = **terminali**
- Canali di comunicazione di diversi tipi (fibra, cavo, radio, satellite, ...) **link** = **collegamenti**
- Nodi di rete chiamati **router** = **nodi**
- Altri nodi di rete locali (switch, access point, modem, ...)



Componenti fisiche: host (terminali)

- Tutti gli **host** per la rete sono sistemi in grado di **inviare e ricevere informazioni** per le loro applicazioni finali
- Ma in realtà hanno caratteristiche molto diverse



Server fisici e virtuali per data center di servizi cloud



Dispositivi personali



Oggetti intelligenti



Componenti fisiche: link (collegamenti)

- I collegamenti possono essere di natura fisica molto diversa (fibra ottica, cavi coassiali, doppini, radio, ecc.)
- Differiscono anche per tecnologia di trasmissione dell'informazione
- E ovviamente per la velocità di trasmissione (rate) misurato in bit al secondo (b/s, Kb/s, Mb/s, Gb/s, Tb/s)



Fibra ottica



Cavi coassiali



Doppini



Antenne radio

Componenti fisiche: nodi di rete

- I nodi di rete di internet sono i **router** che operano su unità di informazione (sequenze di bit) finite dette pacchetti
- Esistono altri nodi di rete che a livello locale svolgono altre funzioni di collegamento
- Vedremo che il “livello” a cui opera un nodo di rete è un aspetto importante della tecnologia



router



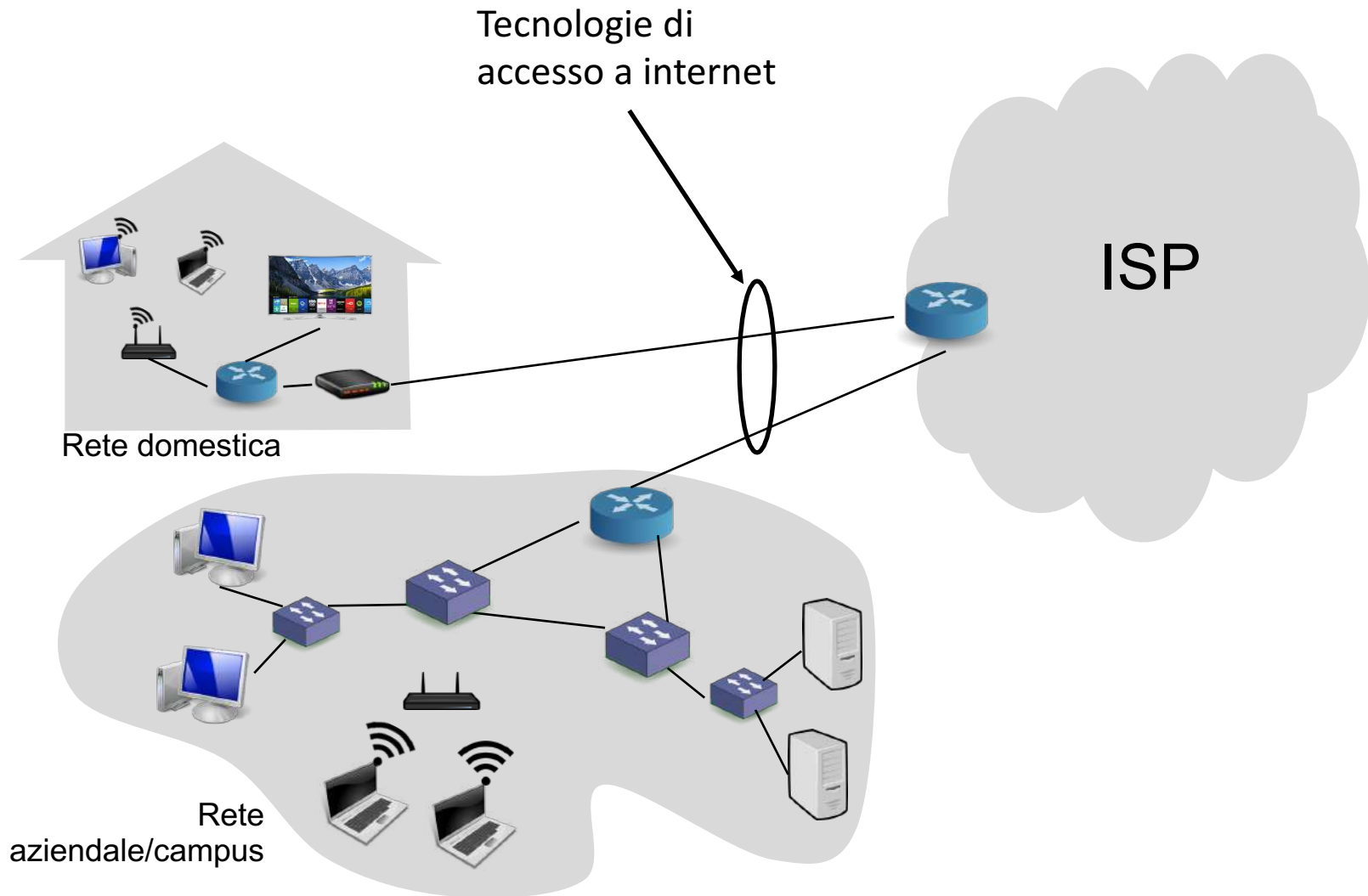
switch



access point



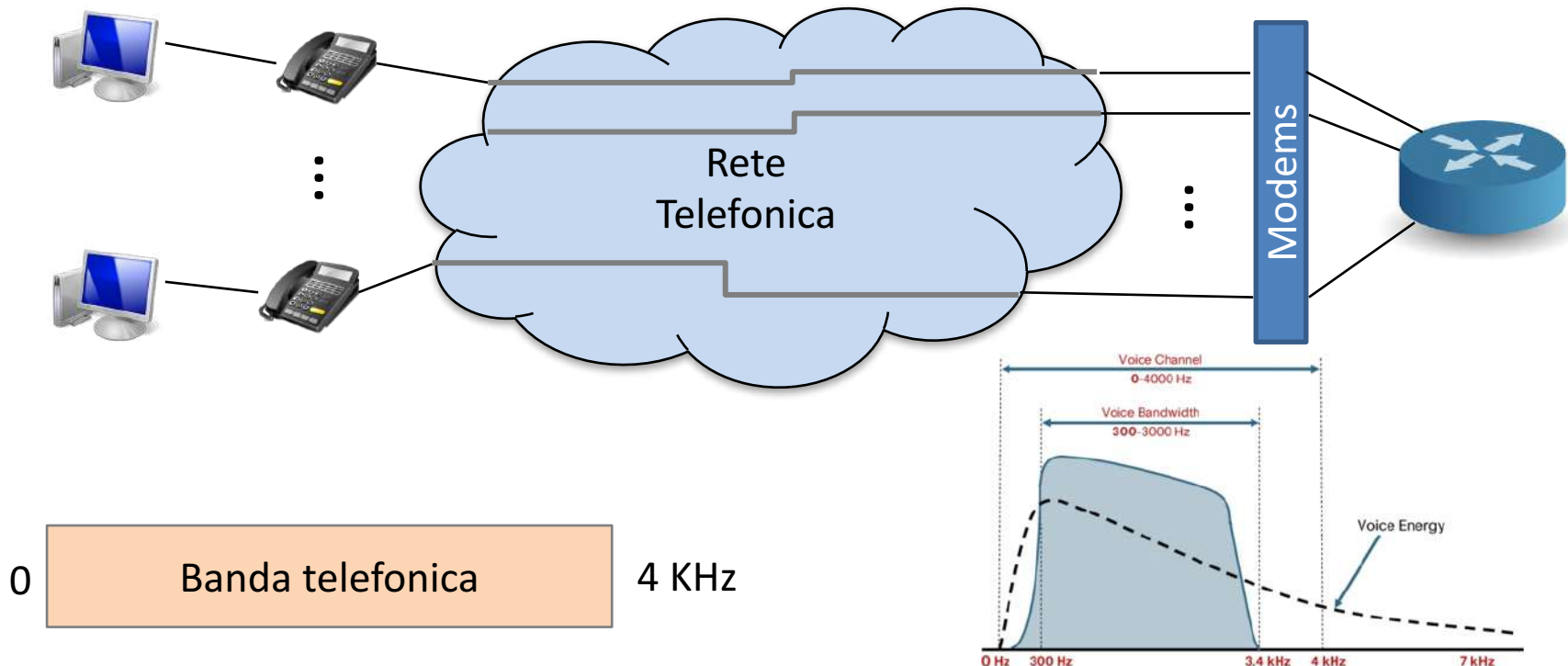
Architettura fisica: Accesso a Internet



Accesso a Internet: Dialup

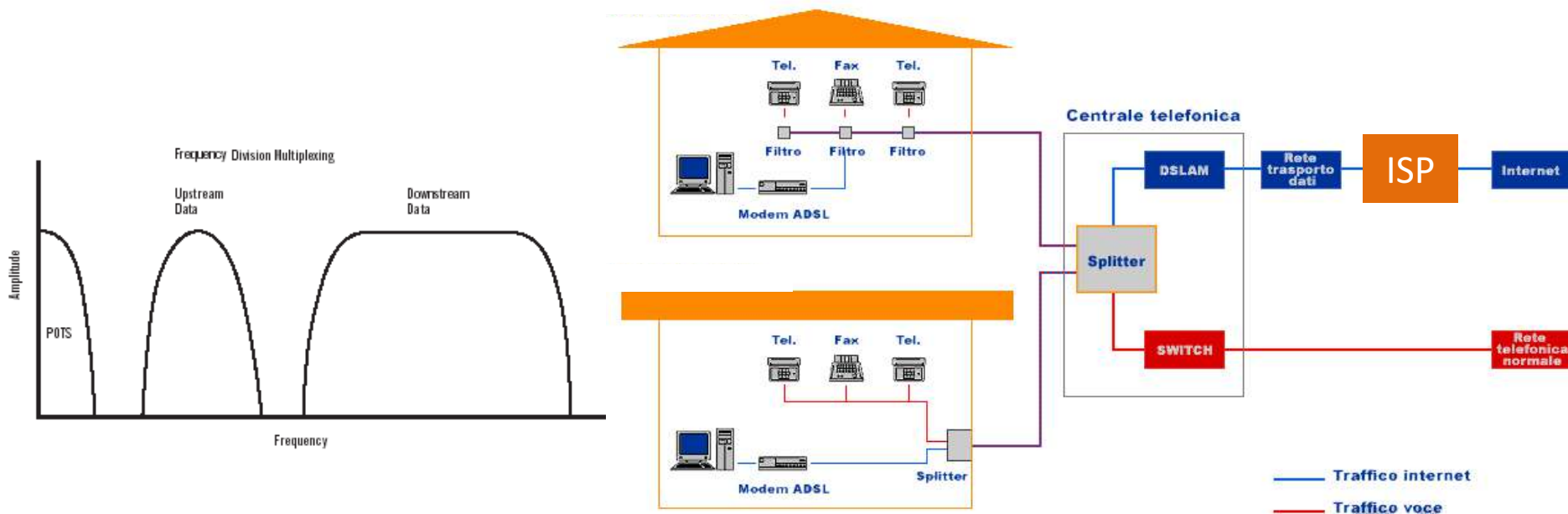
- **Dialup via modem**

- Fino a 56Kbps
- Accesso diretto al router del ISP mediante circuito telefonico
- Trasmissione del segnale in banda fonica



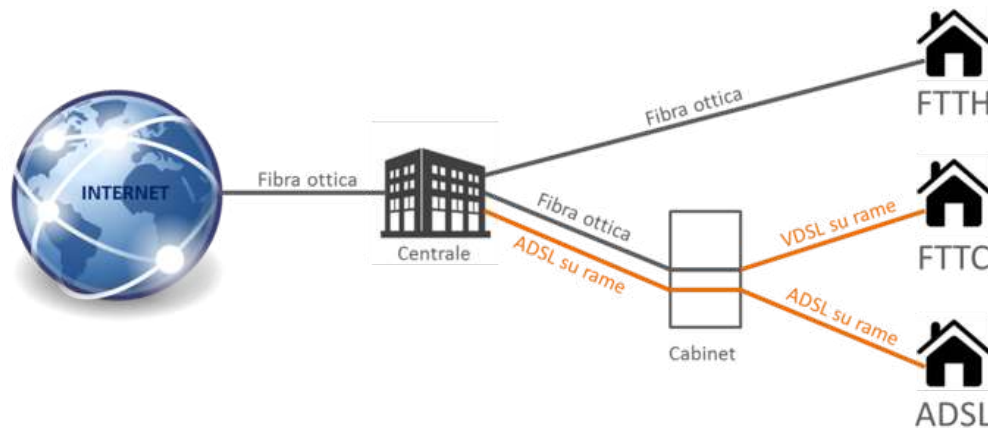
Accesso a Internet: ADSL

- **ADSL: asymmetric digital subscriber line**
 - Fino a 1 Mbps upstream, Fino a 20 Mbps downstream
 - Condivisione del doppino con la rete telefonica fino alla centrale (divisione di frequenza)
 - Accesso al router del provider mediante rete dati ad alta velocità

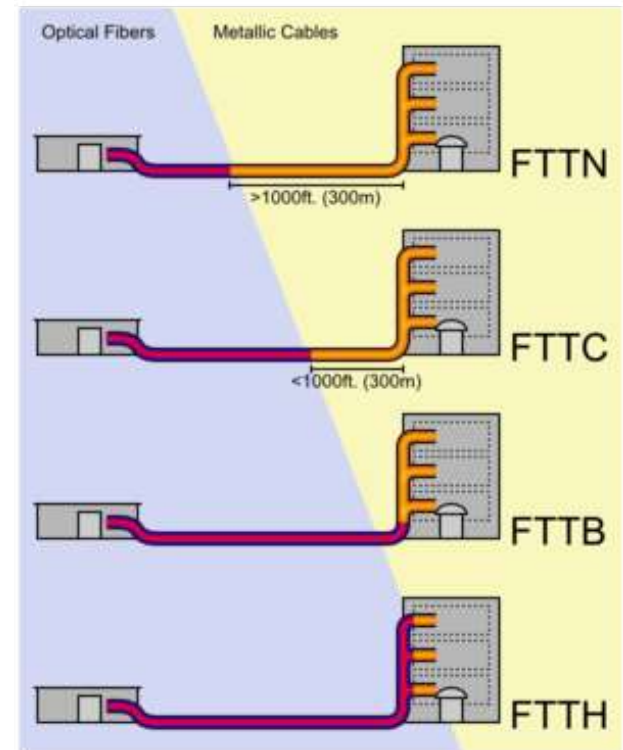


Accesso a Internet: Fibra

- Rete d'accesso di nuova generazione ad alta velocità
 - Sostituzione parziale o totale del doppino telefonico con fibra ottica



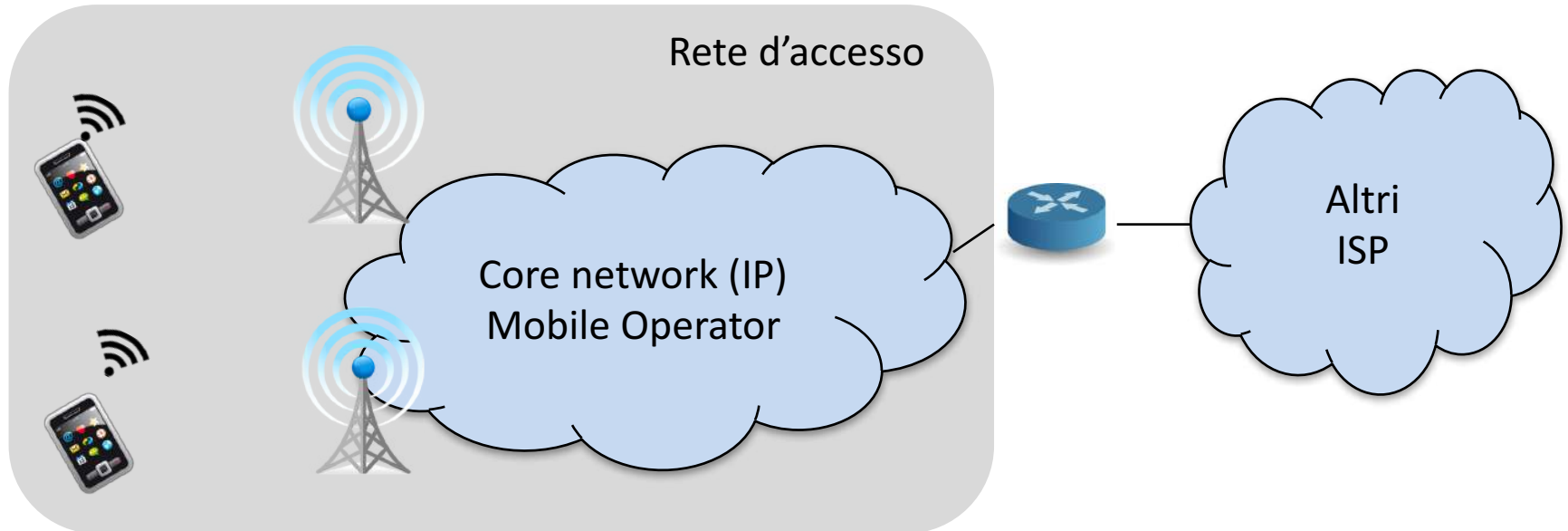
FTTH - Fiber To The Home
FTTB - Fiber To The Basement
FTTC - Fiber To The Curb
FTTN - Fiber To The Neighborhood



Accesso a Internet: Rete cellulare

- **Reti cellulari**

- GPRS/EDGE ~ 200 kbps
- HSPA ~ 14,5(down)/5,7(up) Mbps
- LTE ~ 300(down)/85(up) Mbps



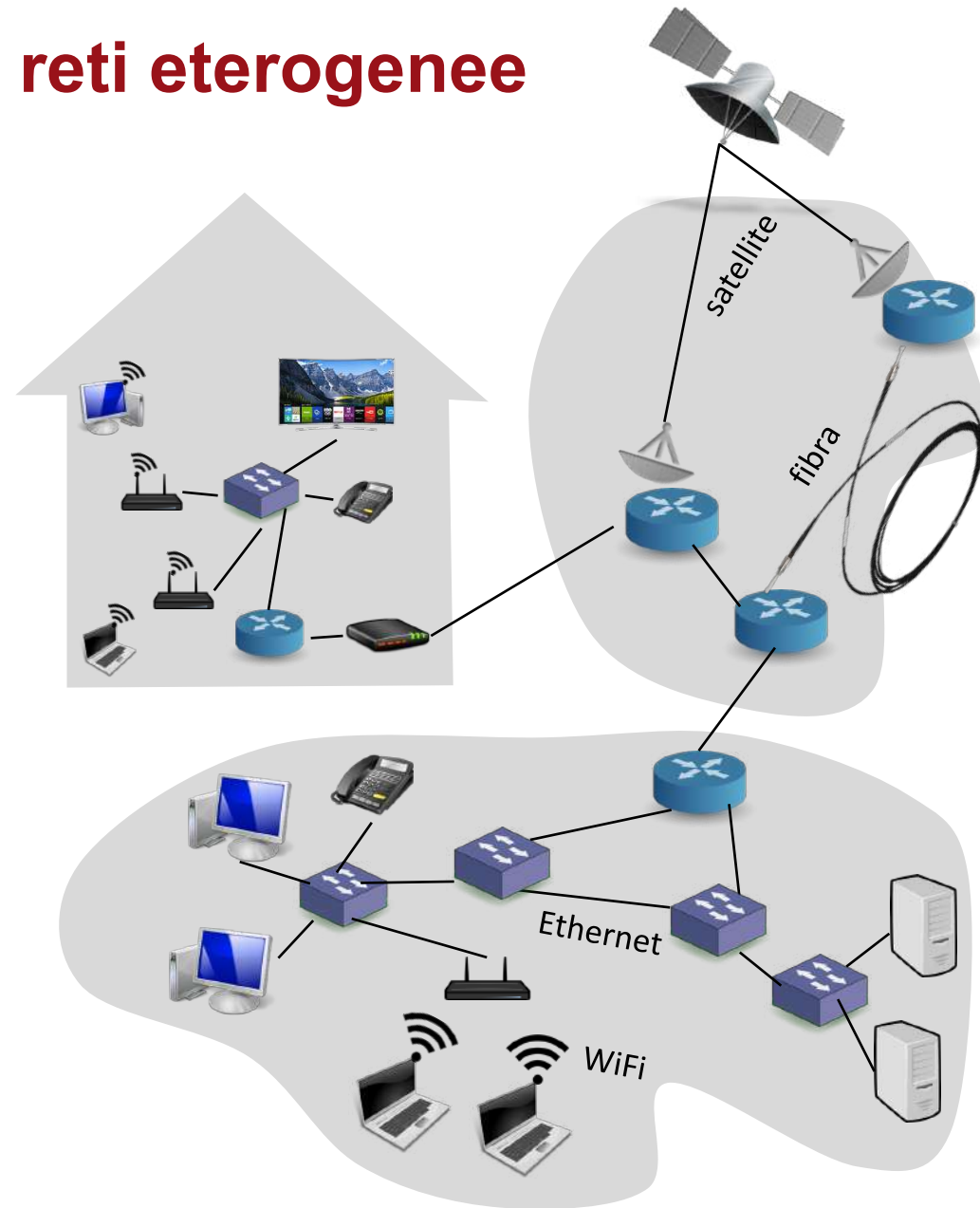
Reti di reti

- Nella descrizione di cosa sia Internet da punto di vista fisico e di servizio abbiamo trascurato un aspetto architettonico fondamentale: **Internet in realtà è un puzzle di tante reti interconnesse**
- **Questo ha due risvolti importanti:**
 - 1) La tecnologia di Internet (IP - Internet Protocol) può essere usata per interconnettere sotto-reti di tipo eterogeneo
 - 2) L'intera rete Internet mondiale è composta da tante reti gestite da operatori indipendenti (ISP – Internet Service Provider) che si accordano per collegarle insieme



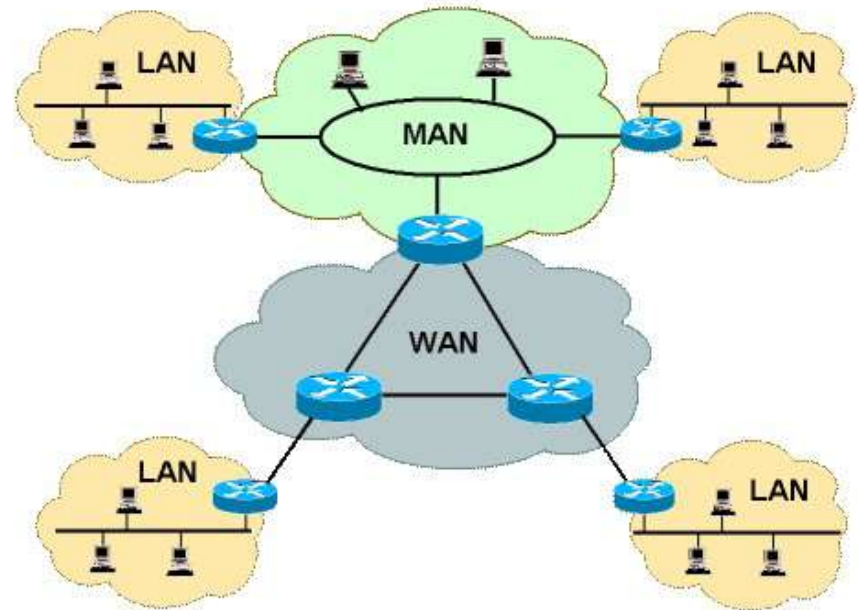
Reti di reti: Insieme di reti eterogenee

- Le diverse porzioni di rete sono composte da **tecnologie diverse**
- I router possono essere interconnessi da **link di vario tipo**
- Ma anche da “**sotto-reti**” che gestiscono internamente propri nodi e link
- Esempio: reti locali Ethernet e/o WiFi

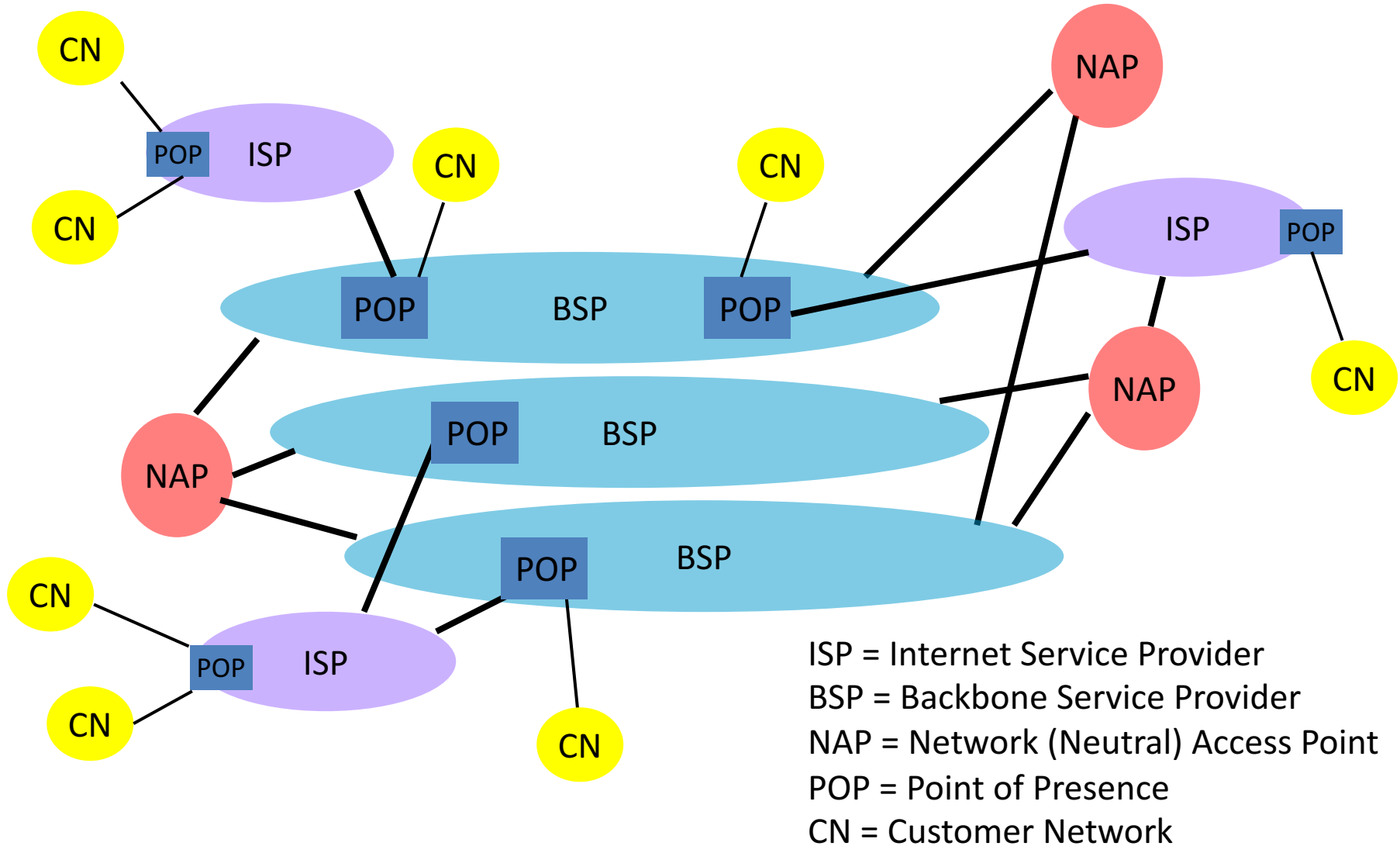


Rete di reti: insieme di reti eterogenee

- Così come Ethernet e WiFi, si possono usare tipi di reti diversi non solo in ambito locale
- **Tipi di rete in base all'estensione geografica:**
 - **LAN:** Local Area Network
 - Impiegate in aree limitate (tipicamente edifici, campus)
 - **MAN:** Metropolitan Area Network
 - Coprono estensioni fino ad alcune decine di km
 - **WAN:** Wide Area Network
 - Hanno copertura ampia a piacere



Rete di reti: architettura di interconnessione



Reti di reti: Internet Exchange Map

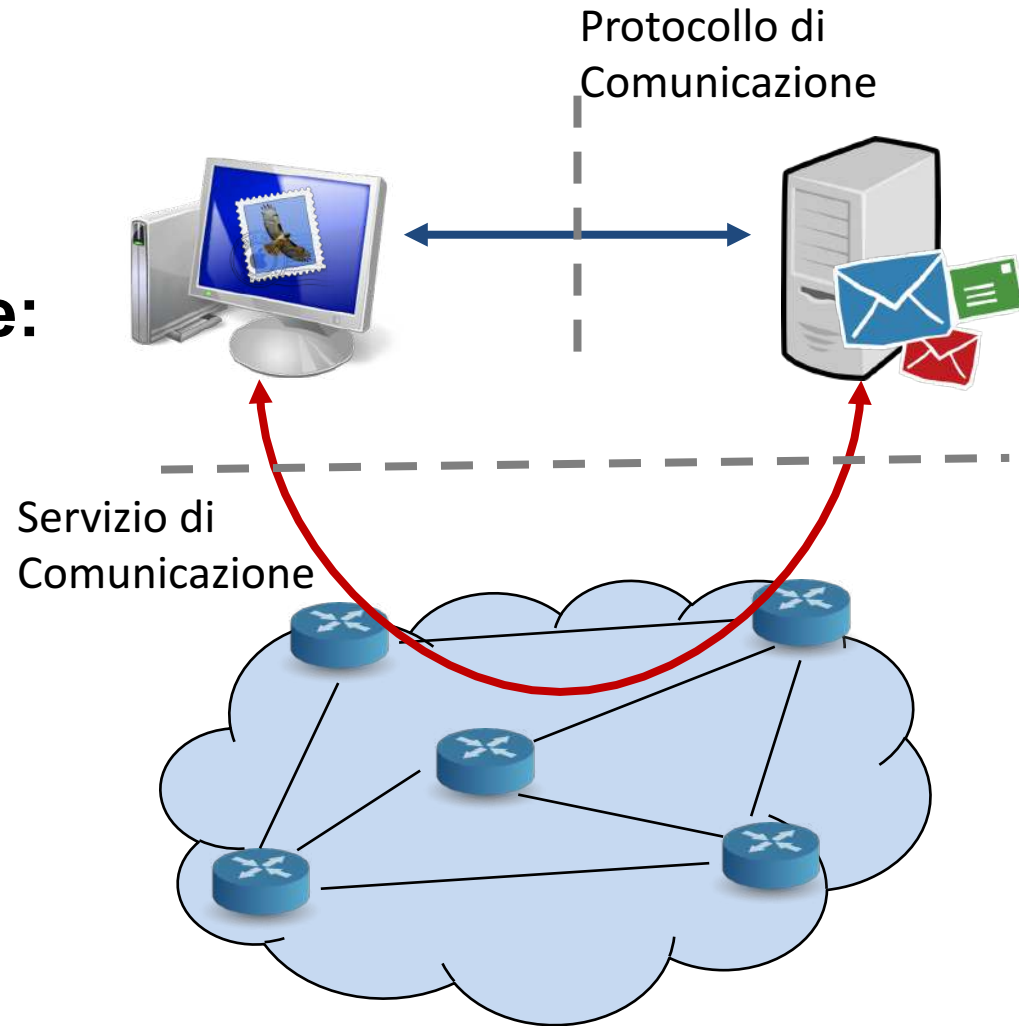


telegeography.com

Cos'è Internet?

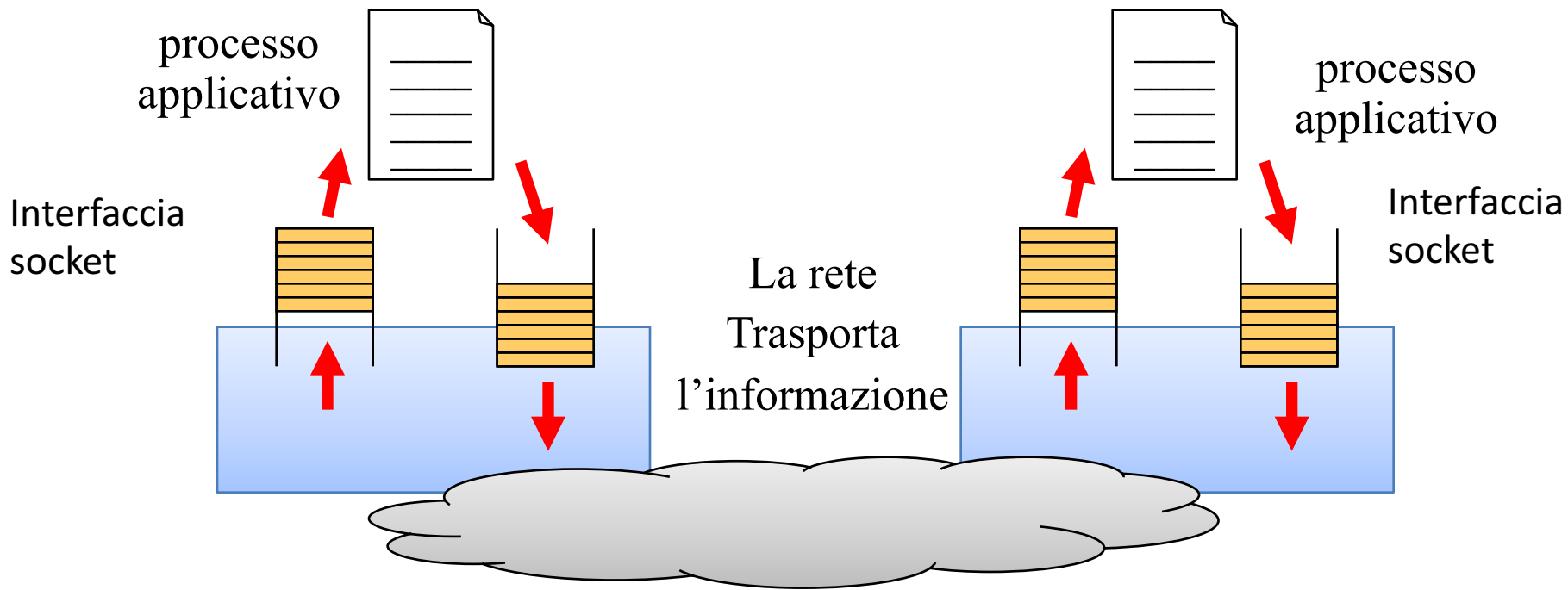
Il servizio e i protocolli di comunicazione

- **Infrastruttura di comunicazione** consente le **applicazioni distribuite**:
 - Web, email, games, e-commerce, file sharing
- **Protocolli di comunicazione** per **inviare e ricevere messaggi**



Servizio di Comunicazione

- La rete fornisce un **servizio di comunicazione** alle applicazioni per il **trasporto** delle informazioni tra i processi remoti
- Il servizio di trasporto offerto dalla rete alle applicazioni può essere di vari tipi

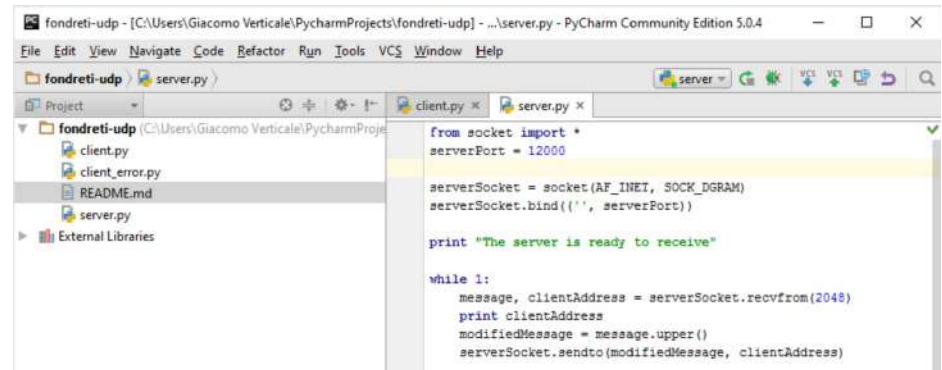


Servizio di Comunicazione

- Possono essere trasportati brevi messaggi in modo non affidabile (esempi: DNS, segnalazione, ecc.)
- Possono essere trasportate sequenze anche lunghe di byte in modo affidabile (web, email, file transfer, ecc.)

Laboratorio: Socket programming

Vedrete in laboratorio come usare Python per accedere all'interfaccia software sia lato client che lato server per trasporto affidabile e non



The screenshot shows the PyCharm Community Edition 5.0.4 interface. The project is named 'fondreti-udp'. The file explorer on the left shows a directory structure with 'client.py', 'client_error.py', 'README.md', and 'server.py'. The main editor window displays the code for 'server.py'. The code is as follows:

```
from socket import *
serverPort = 12000

serverSocket = socket(AF_INET, SOCK_DGRAM)
serverSocket.bind(('', serverPort))

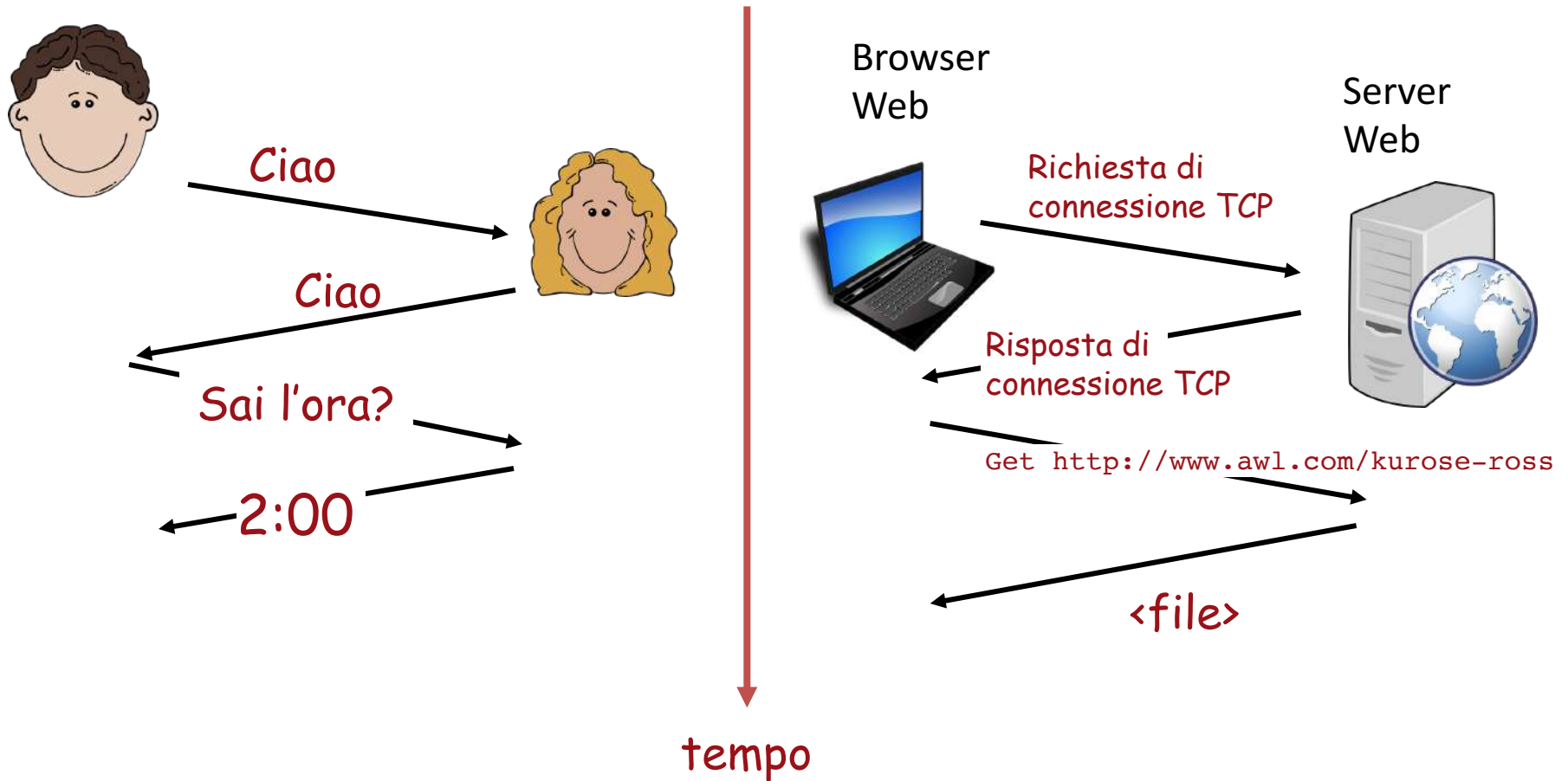
print "The server is ready to receive"

while 1:
    message, clientAddress = serverSocket.recvfrom(2048)
    print clientAddress
    modifiedMessage = message.upper()
    serverSocket.sendto(modifiedMessage, clientAddress)
```



Protocolli di comunicazione

Protocollo umano e protocollo di rete



Protocolli di comunicazione:

Esempio posta elettronica

S: 220 hamburger.edu

C: HELO crepes.fr

S: 250 Hello crepes.fr, pleased to meet you

C: MAIL FROM: <alice@crepes.fr>

S: 250 alice@crepes.fr... Sender ok

C: RCPT TO: <bob@hamburger.edu>

S: 250 bob@hamburger.edu ... Recipient ok

C: DATA

S: 354 Enter mail, end with "." on a line by itself

C: Do you like ketchup?

C: How about pickles?

C: .

S: 250 Message accepted for delivery

C: QUIT

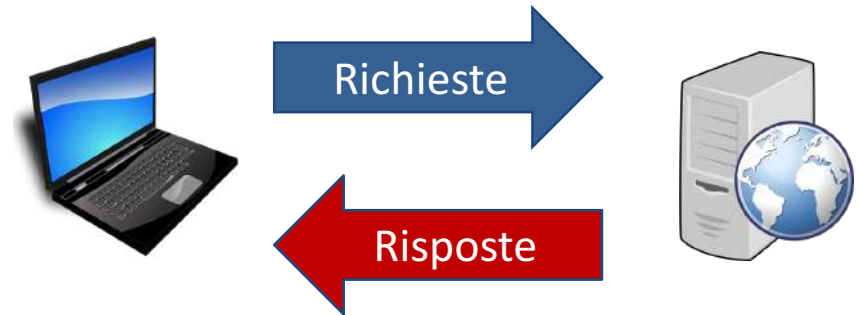
S: 221 hamburger.edu closing connection



Protocolli di comunicazione: modelli

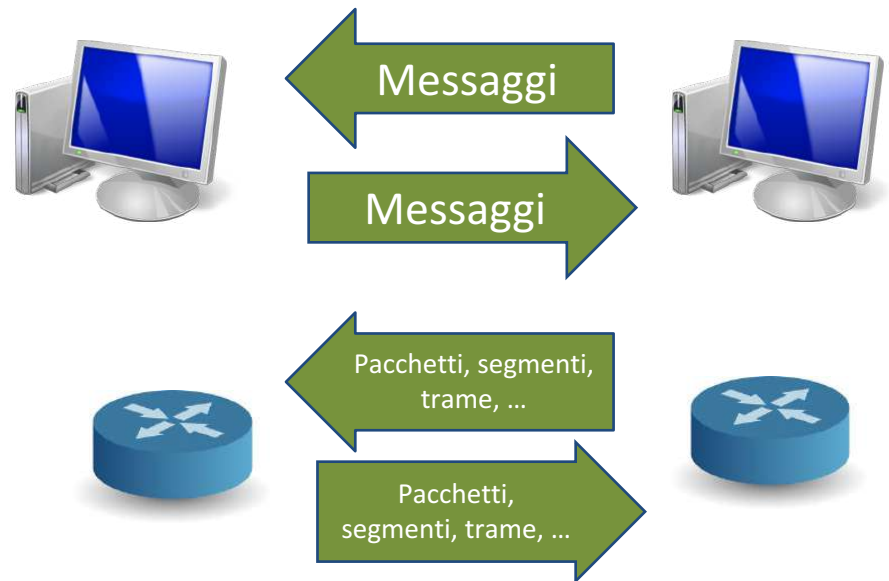
- **Modello client/server**

- client chiedono il servizio, i server lo forniscono
- I client fanno domande, i server rispondono



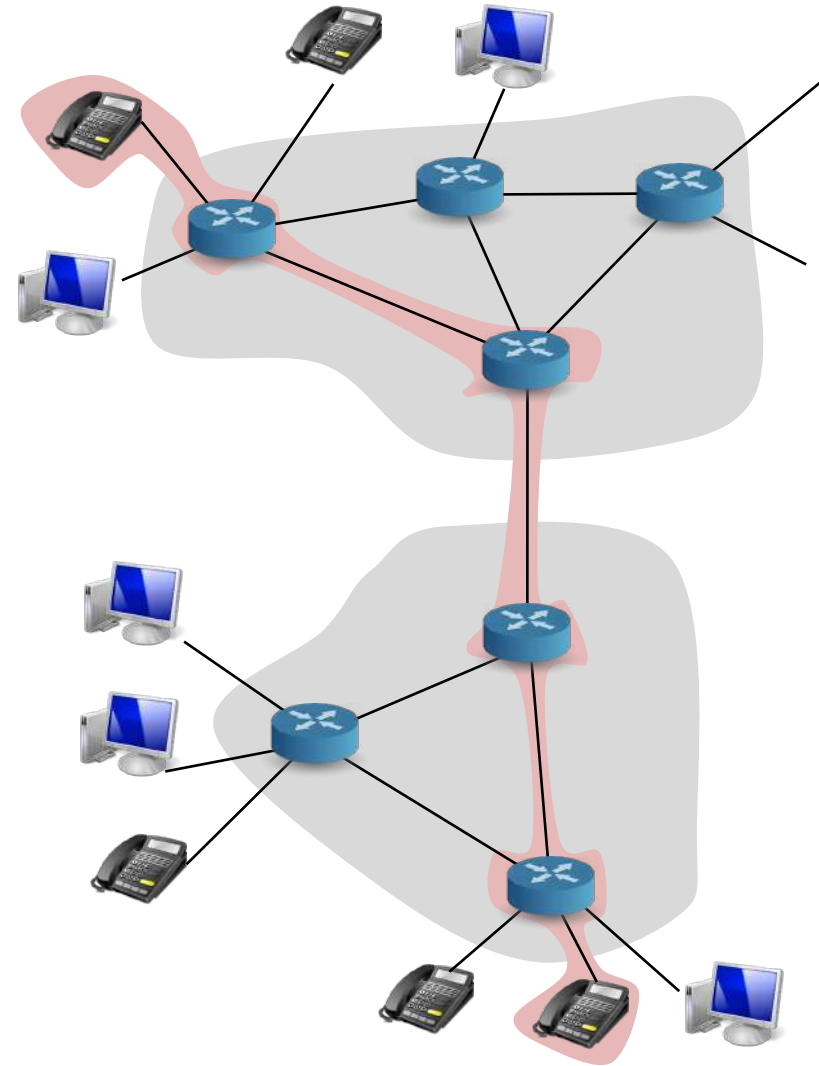
- **Modello peer-to-peer:**

- Tutti i terminali collaborano senza distinzione di ruoli (o quasi)



Come funziona Internet?

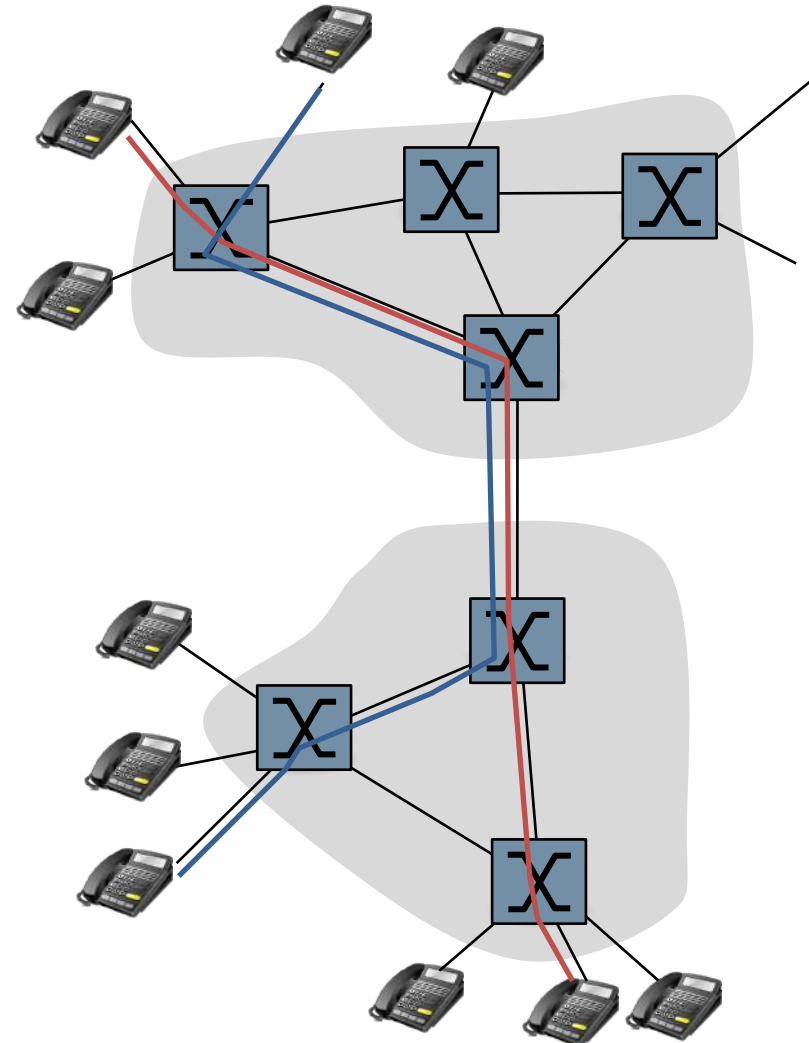
- E' la domanda che ci occuperà l'intero corso
- Ma partiamo da meccanismo di base
- **Come può essere trasferita l'informazione in rete?**
 - **Commutazione di circuito:** circuito dedicato per chiamata
 - **Commutazione di pacchetto:** dati inviati in rete con messaggi



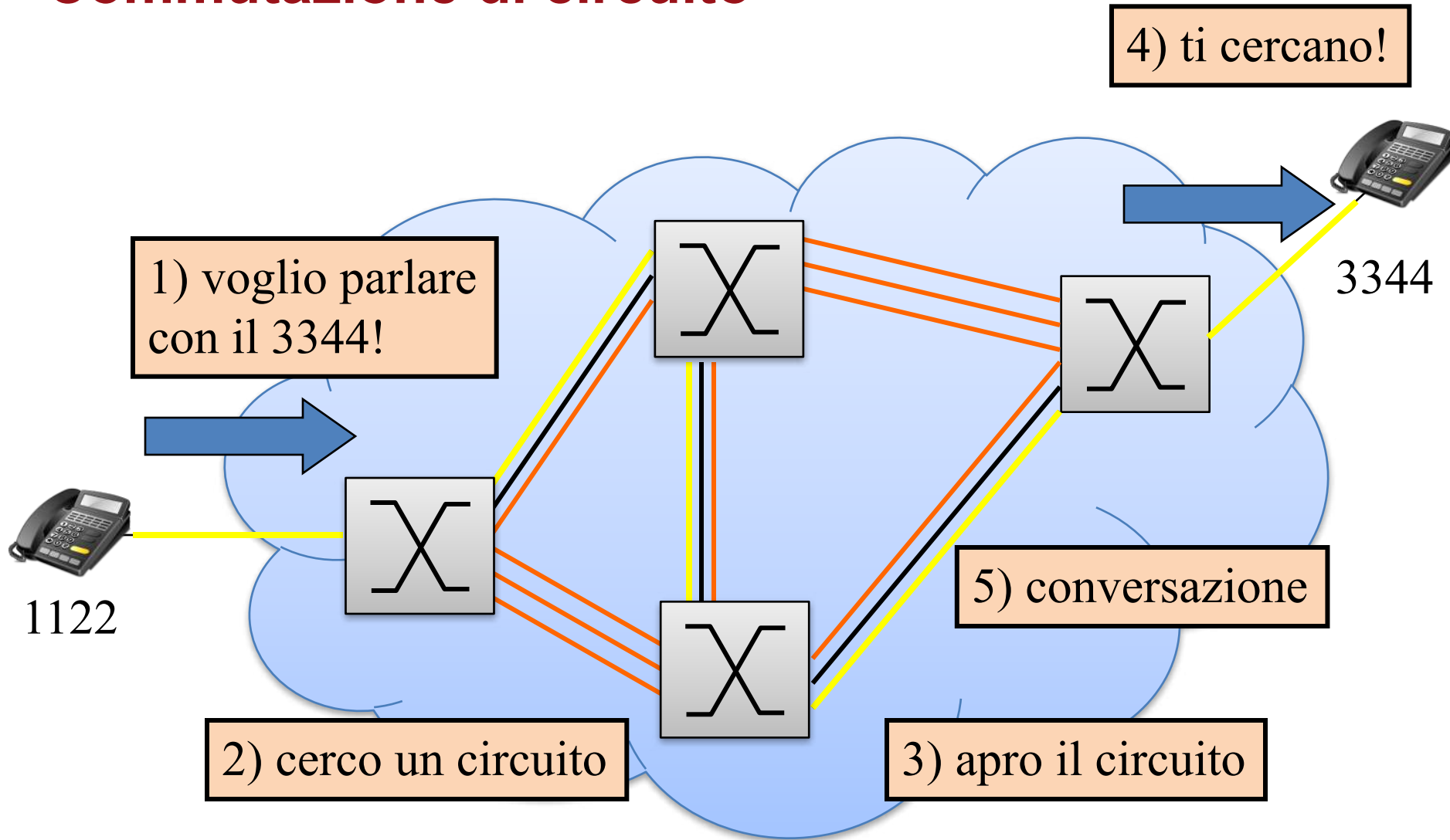
Commutazione di circuito

Le risorse per la comunicazione sono riservate per la chiamata

- Esempio rete telefonica



Commutazione di circuito



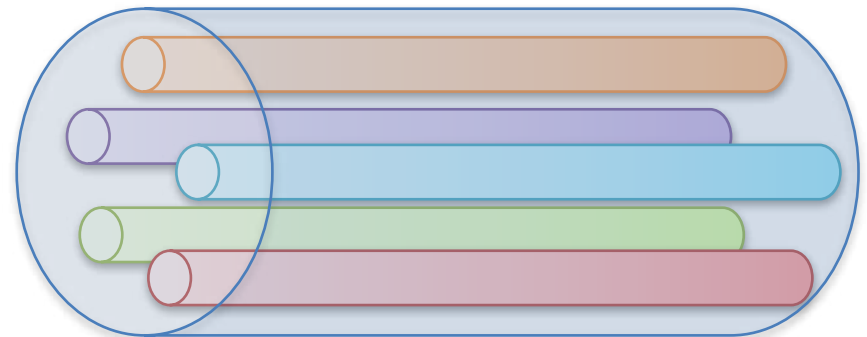
Commutazione di circuito

- Risorse di rete
(**collegamenti**) suddivise
in “pezzi”
- ciascun “pezzo” (= circuito)
viene allocato ai vari
collegamenti
- le risorse rimangono *inattive*
se non utilizzate (*non c'è
condivisione*)



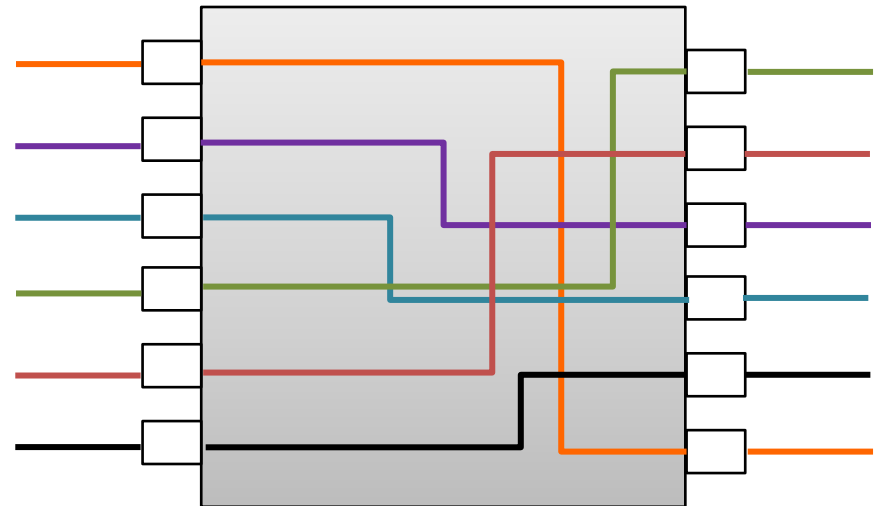
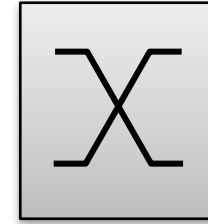
suddivisione della banda
in “pezzi”

- divisione di
frequenza
- divisione di tempo



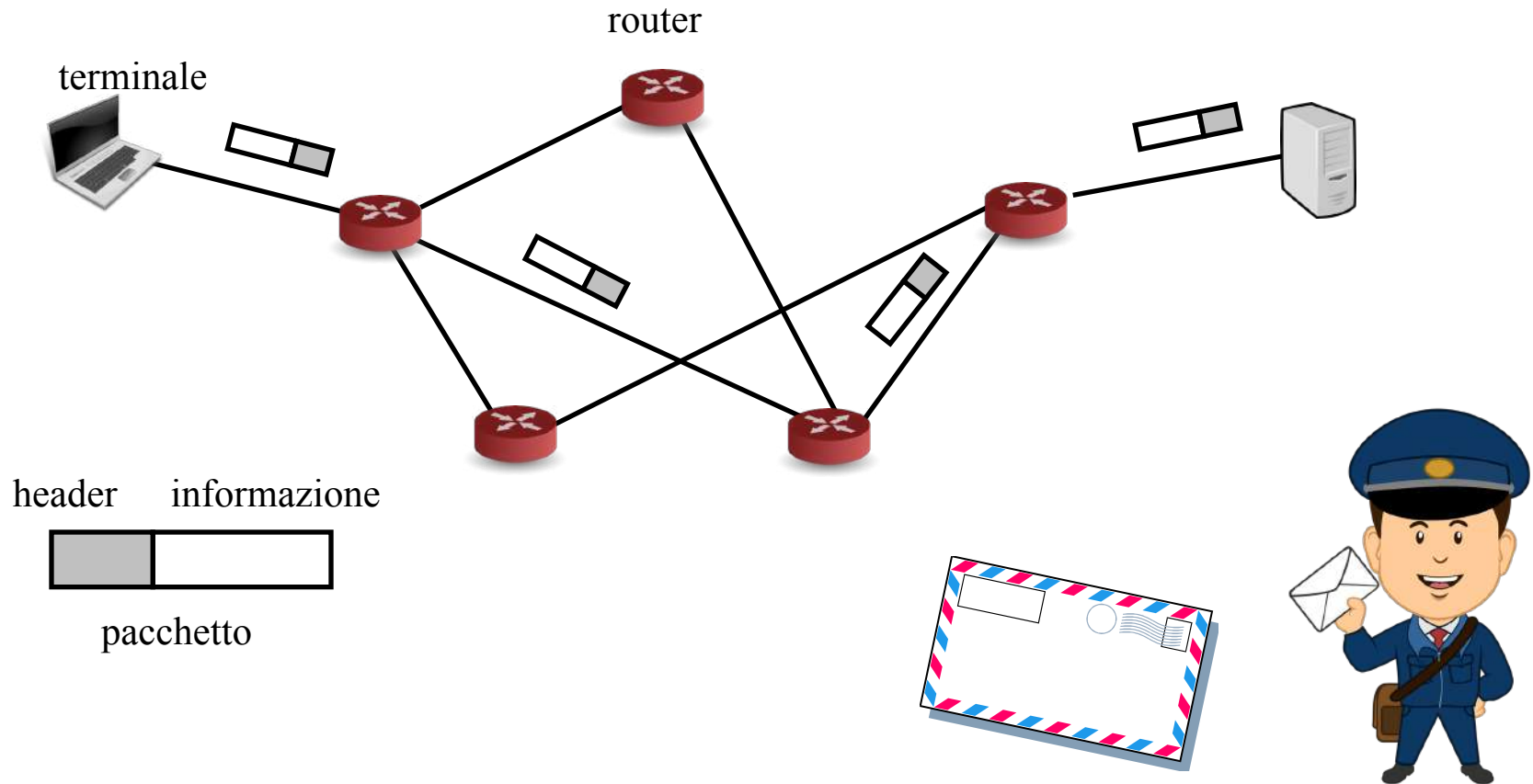
Commutazione di circuito

- **Modello di nodo
(commutatore a circuito)**
 - La capacità dei canali in ingresso è pari alla capacità (in bit al secondo) di quelli in uscita
 - Non serve memorizzare temporaneamente l'informazione

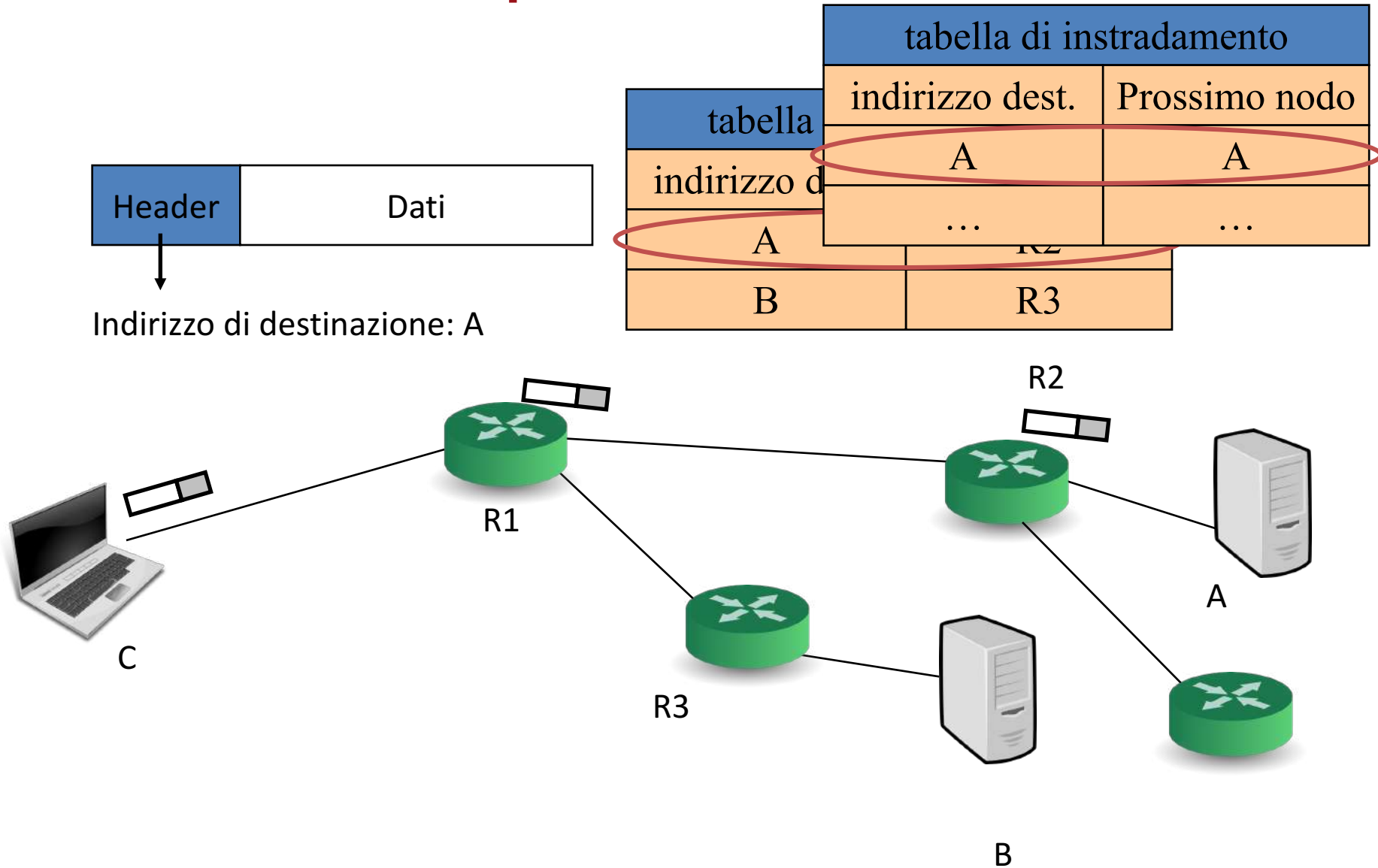


Commutazione di pacchetto

- **Informazione suddivisa in pezzi**
- **Collegamenti non suddivisi**



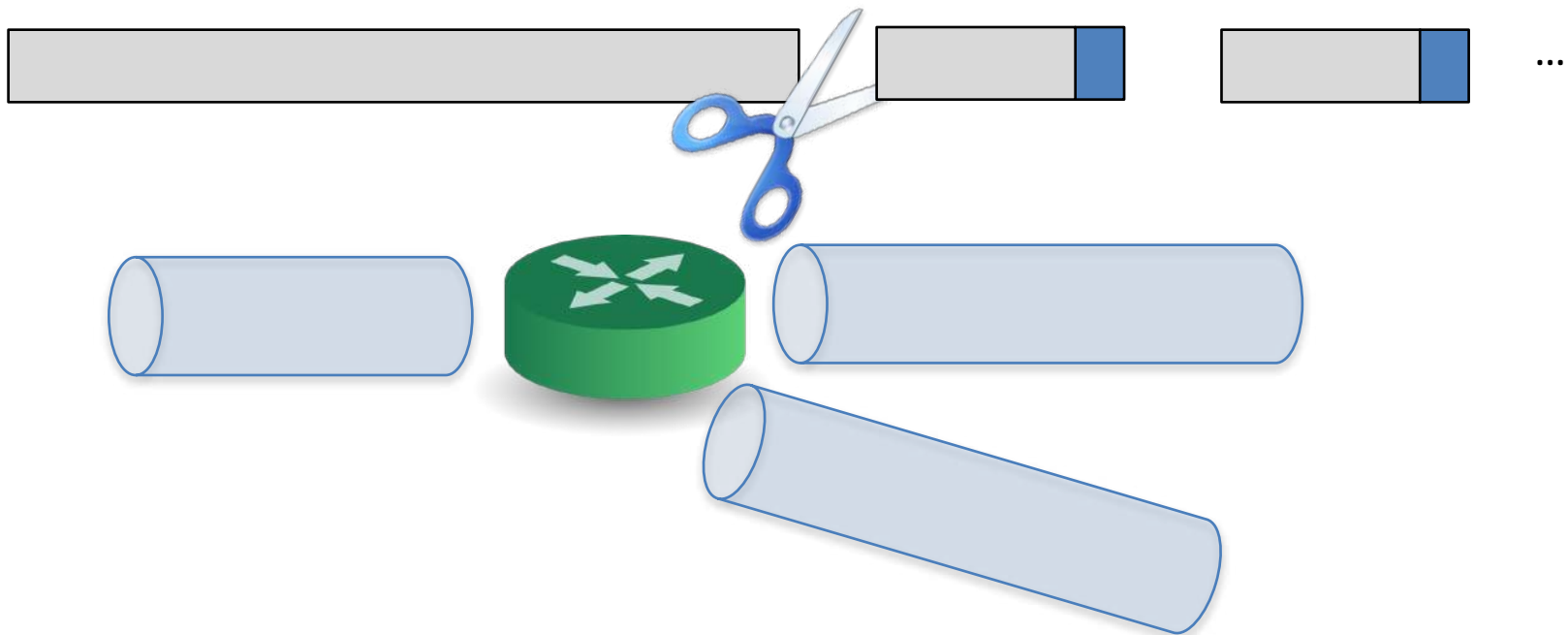
Commutazione di pacchetto



Commutazione di pacchetto

Il flusso di dati viene suddiviso in *pacchetti*

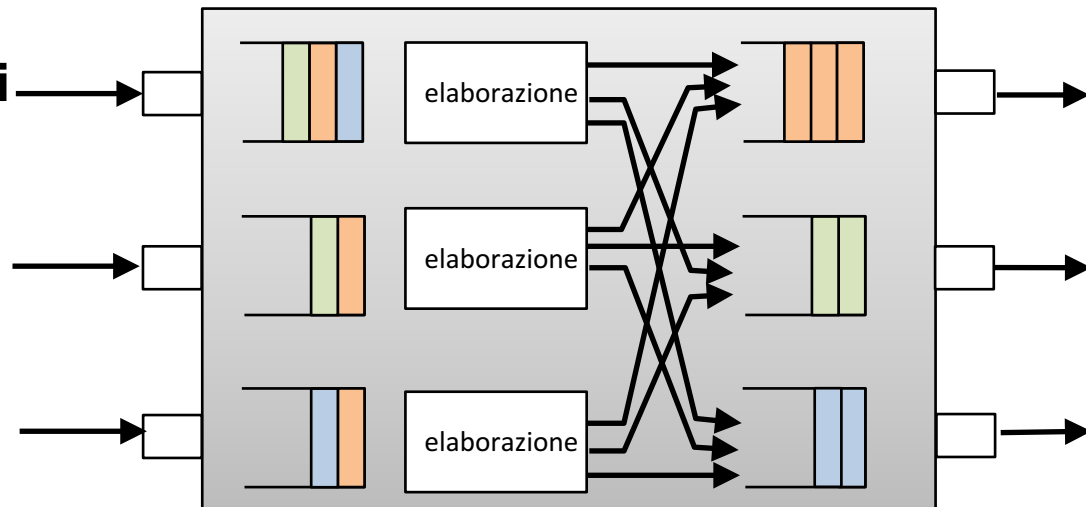
- I pacchetti di tutti gli utenti *condividono* le risorse di rete
- Ciascun pacchetto utilizza completamente il canale
- Le risorse vengono usate a seconda delle necessità



Commutazione di pacchetto

- **Modello di nodo
(packet switch/router)**

- L'arrivo dei pacchetti è **asincrono**
- La capacità dei collegamenti **arbitraria**
- Possono esserci **conflitti temporali** per la trasmissione
- Serve **memorizzare temporaneamente (coda)**
 - **All'ingresso** per analizzare indirizzo destinazione
 - **All'uscita** per gestire conflitti

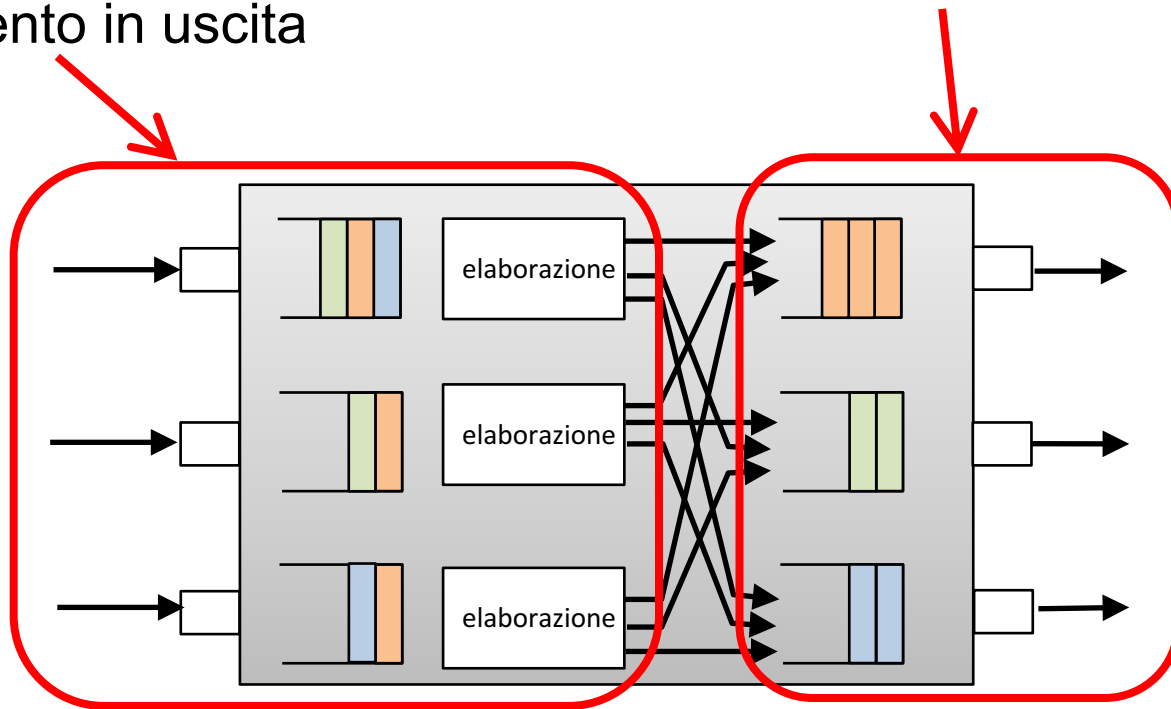


Commutazione di pacchetto

Contesa per le risorse:

store and forward: il commutatore deve ricevere l'intero pacchetto prima di poter cominciare a trasmettere sul collegamento in uscita

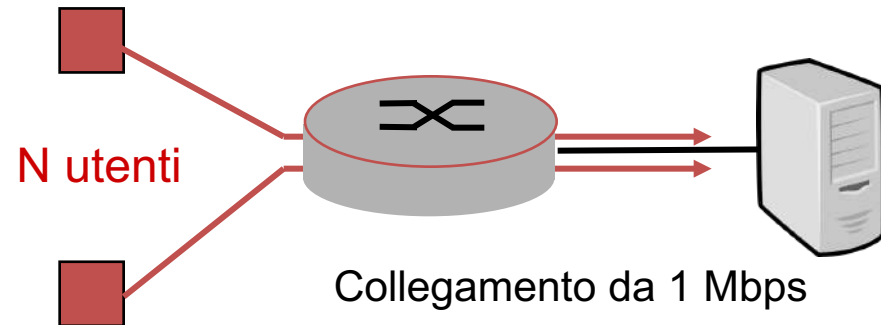
Multiplazione statistica: accodamento dei pacchetti, attesa per l'utilizzo del collegamento



Confronto tra pacchetto e circuito

Esempio:

- 1 collegamento da 1 Mbps
- Ciascun utente:
 - Genera 100 kbps quando è “attivo”
 - E’ attivo per il 10% del tempo
- **commutazione di circuito:**
 - 10 utenti ($1 \text{ Mbps} / 100 \text{ kbps} = 10$)



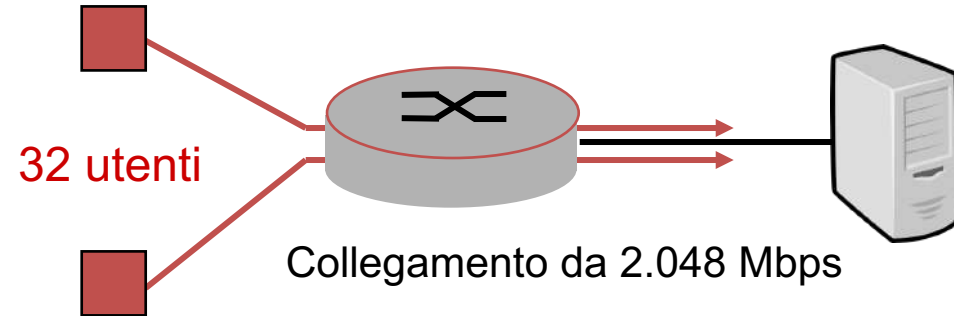
- **commutazione di pacchetto:**
 - con 35 utenti, la probabilità di averne > 10 attivi è inferiore a 0,0004 (risultato di teoria della prob.)

La commutazione di pacchetto consente a più utenti di usare la rete!

Confronto tra pacchetto e circuito

Esempio:

- 1 collegamento da 2.048 Mbps
- 32 utenti, ciascun utente:
 - Chiede pagine web di 50KB ogni 62.5s in media
- **commutazione di circuito:**
 - 1 canale 64 kbps per utente
 - Ritardo di trasferimento pagina web: 6.25s
($400 \text{ kbit} / 64 \text{ kbps} = 6,25\text{s}$)

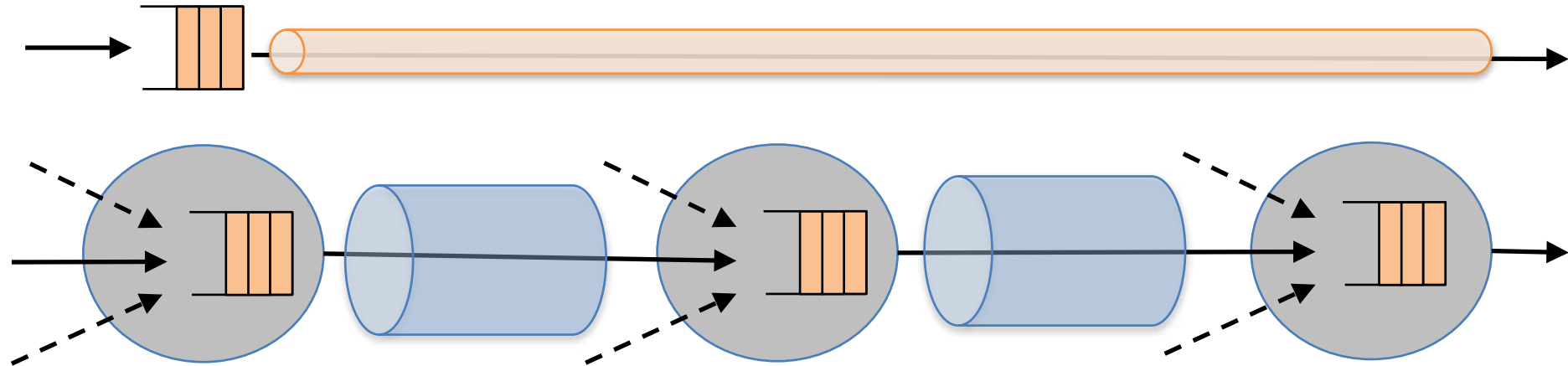


- **commutazione di pacchetto:**
 - Ritardo di trasferimento medio pagina web: 0.22s
(risultato di teoria delle code)

La commutazione di pacchetto consente di scaricare le informazioni più velocemente!



Confronto tra pacchetto e circuito



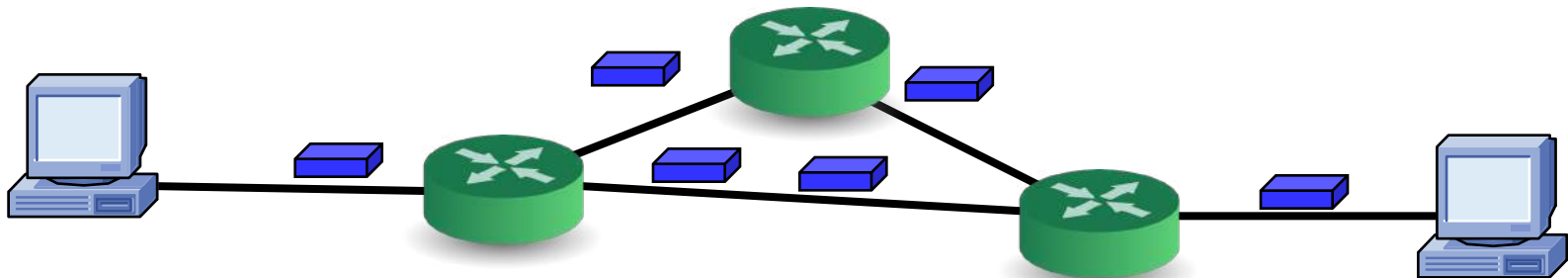
La commutazione di pacchetto è la scelta di Internet

- **Il problema delle coda: ritardo e perdita di pacchetti**

- Sono necessari protocolli per il trasferimento affidabile dei dati e per il controllo della congestione

Commutazione di pacchetto: datagram e circuito virtuale

- **Esistono due tipi di commutazione di pacchetto:**
 - Datagram
 - Circuito virtuale

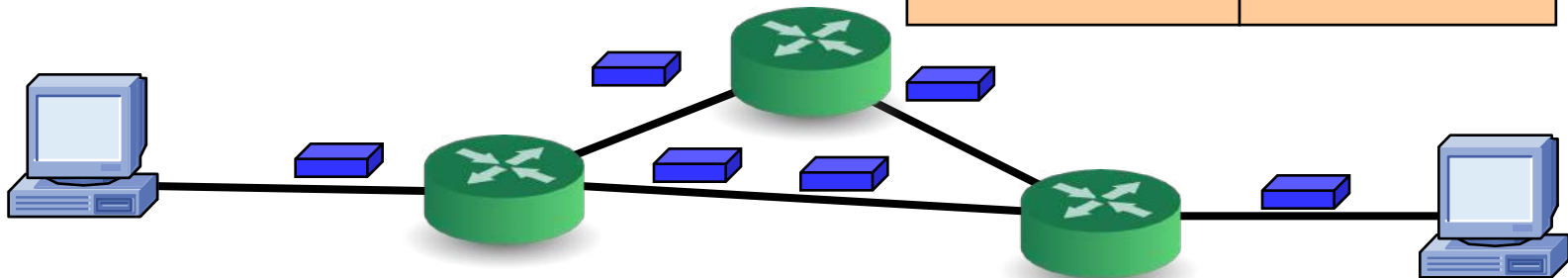


Commutazione di pacchetto: Datagram

- Nella commutazione datagram la scelta della porta d'uscita viene fatta sulla base del solo indirizzo di destinazione

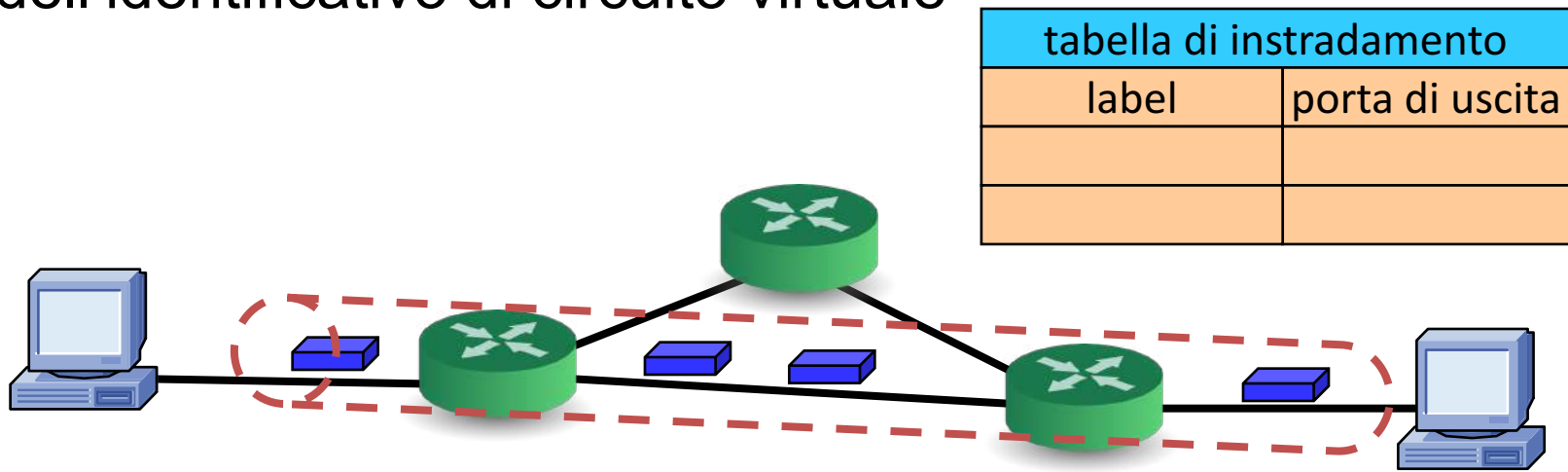
- ❑ I pacchetti dello stesso flusso sono inoltrati indipendentemente

tabella di instradamento	
destinazione	porta di uscita



Commutazione di pacchetto: Circuito virtuale

- I nodi identificano i **pacchetti di un flusso informativo sulla base di un identificativo di circuito virtuale (CVI o label)**
- Il circuito virtuale viene instaurato in una fase di setup prima della fase dati
- Dopo la fase di setup i pacchetti seguono tutti lo stesso percorso in rete perché sono instradati sulla base dell'identificativo di circuito virtuale





1c – Ritardi e Throughput

**Tempo di trasmissione, Propagazione,
Altri ritardi, Throughput**

Velocità di trasmissione

- E' la velocità (rate) R con la quale l'informazione digitale viene trasmessa su una linea = numero di bit trasmessi nell'unità di tempo
- E' misurata in bit/s (bps)



- Unità di misura:

$$1 \text{ kbps (kb/s)} = 10^3 \text{ bps}$$

$$1 \text{ Mbps (kb/s)} = 10^6 \text{ bps}$$

$$1 \text{ Gbps (kb/s)} = 10^9 \text{ bps}$$

$$1 \text{ B} = 8 \text{ bit}$$

$$1 \text{ kB} = 10^3 \text{ B}$$

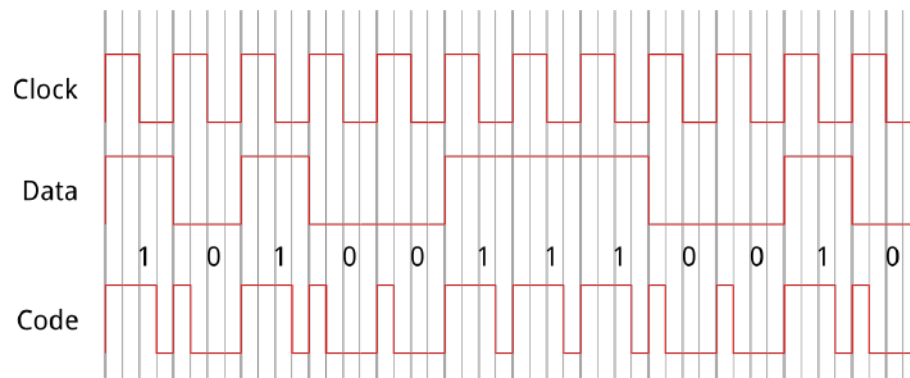
$$1 \text{ MB} = 10^6 \text{ B}$$

$$1 \text{ GB} = 10^9 \text{ B}$$



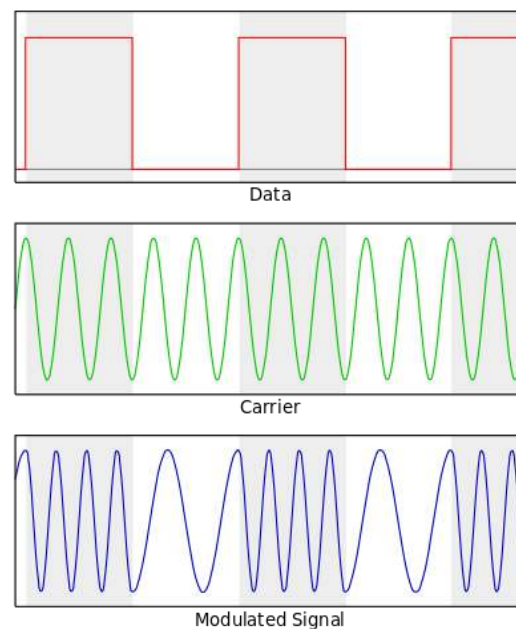
Velocità di trasmissione: esempi

- **Segnale nel tempo di Ethernet con codifica Manchester:**



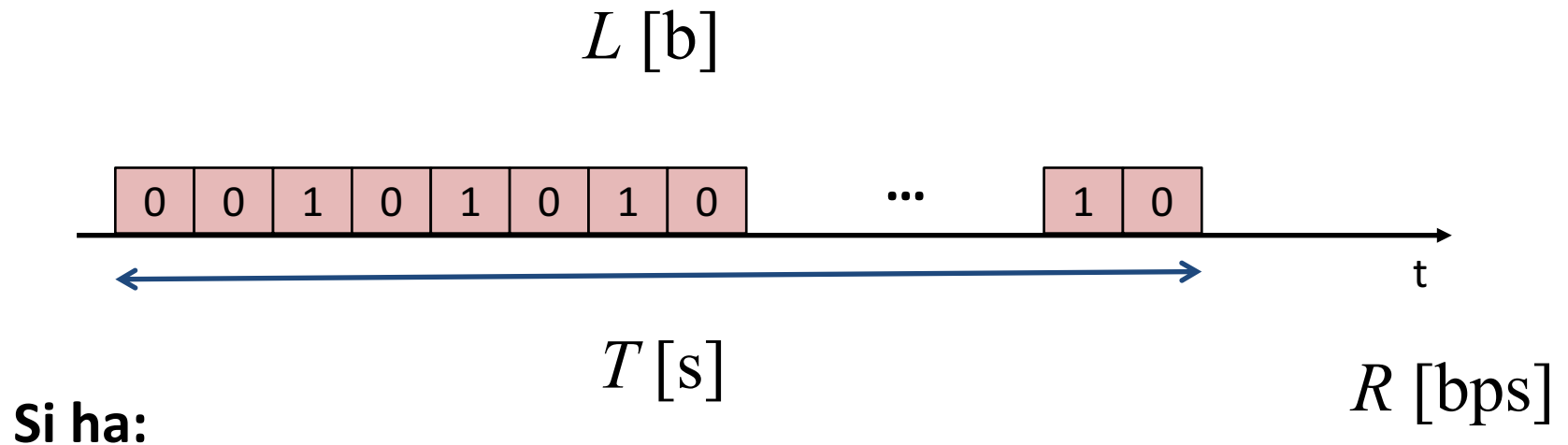
- **Segnale modulato in frequenza del GSM:**

NOTA: Il motivo per il quale la durata del singolo bit non può essere piccolissima (e quindi la velocità grandissima) deriva a limiti dei canali trasmissivi che saranno trattati nell'ultima parte del corso.



Tempo di trasmissione

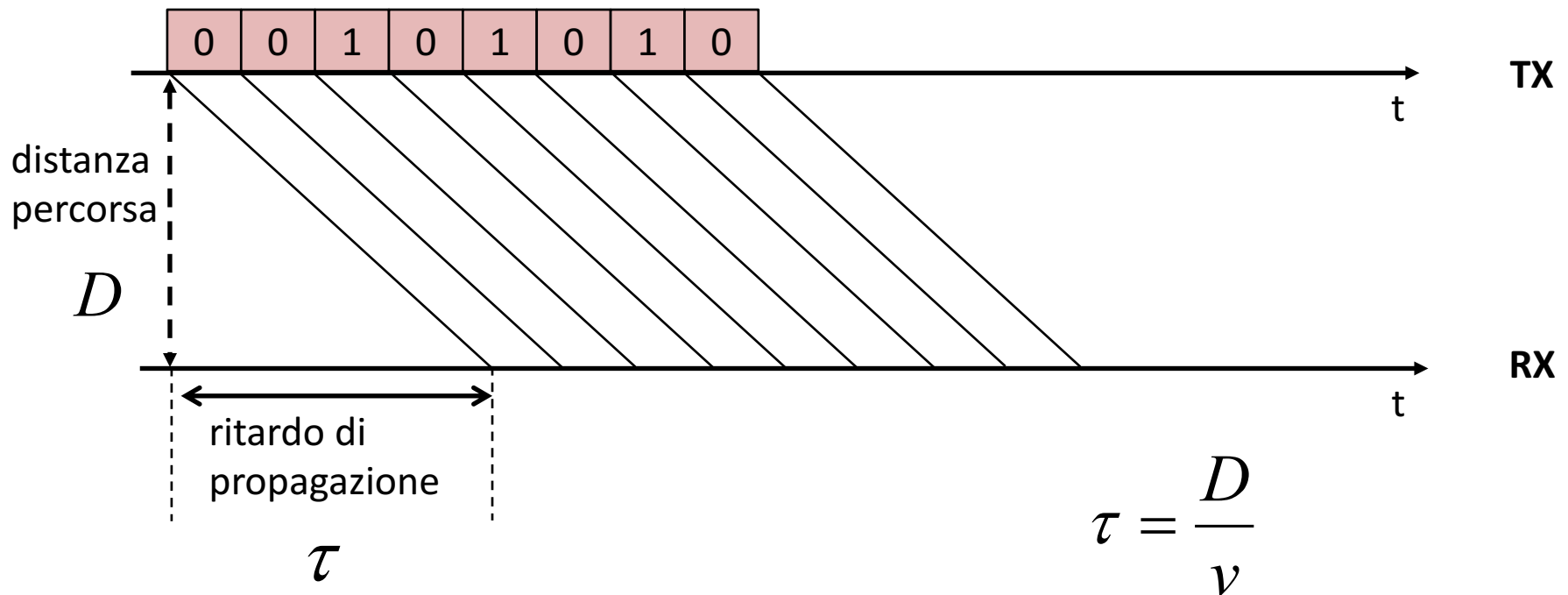
- Il tempo T per trasmettere L bits dipende dalla velocità di trasmissione R



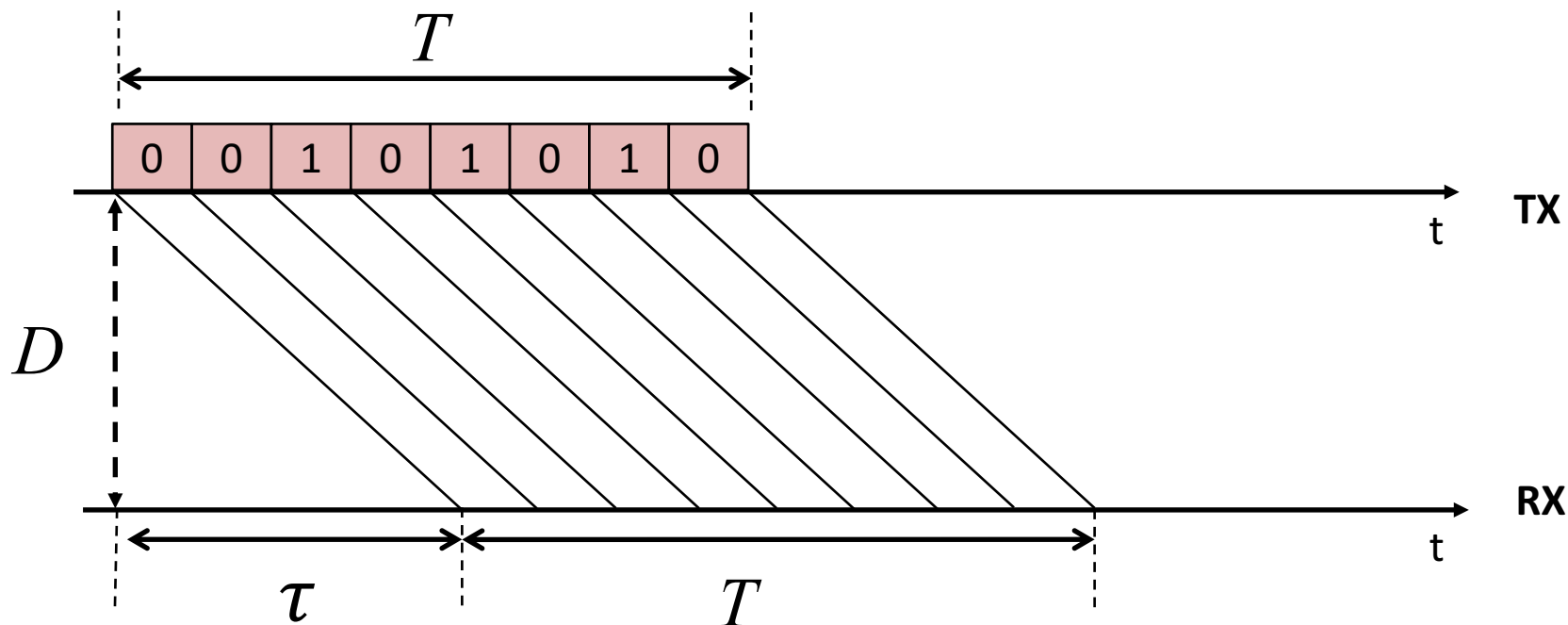
$$T = \frac{L}{R}$$

Ritardo di propagazione

- Il tempo τ affinché un impulso trasmesso dal trasmettitore TX raggiunga il ricevitore RX dipende dalla distanza D (in m) e dalla velocità di propagazione v (in m/s , prossima alla velocità della luce)



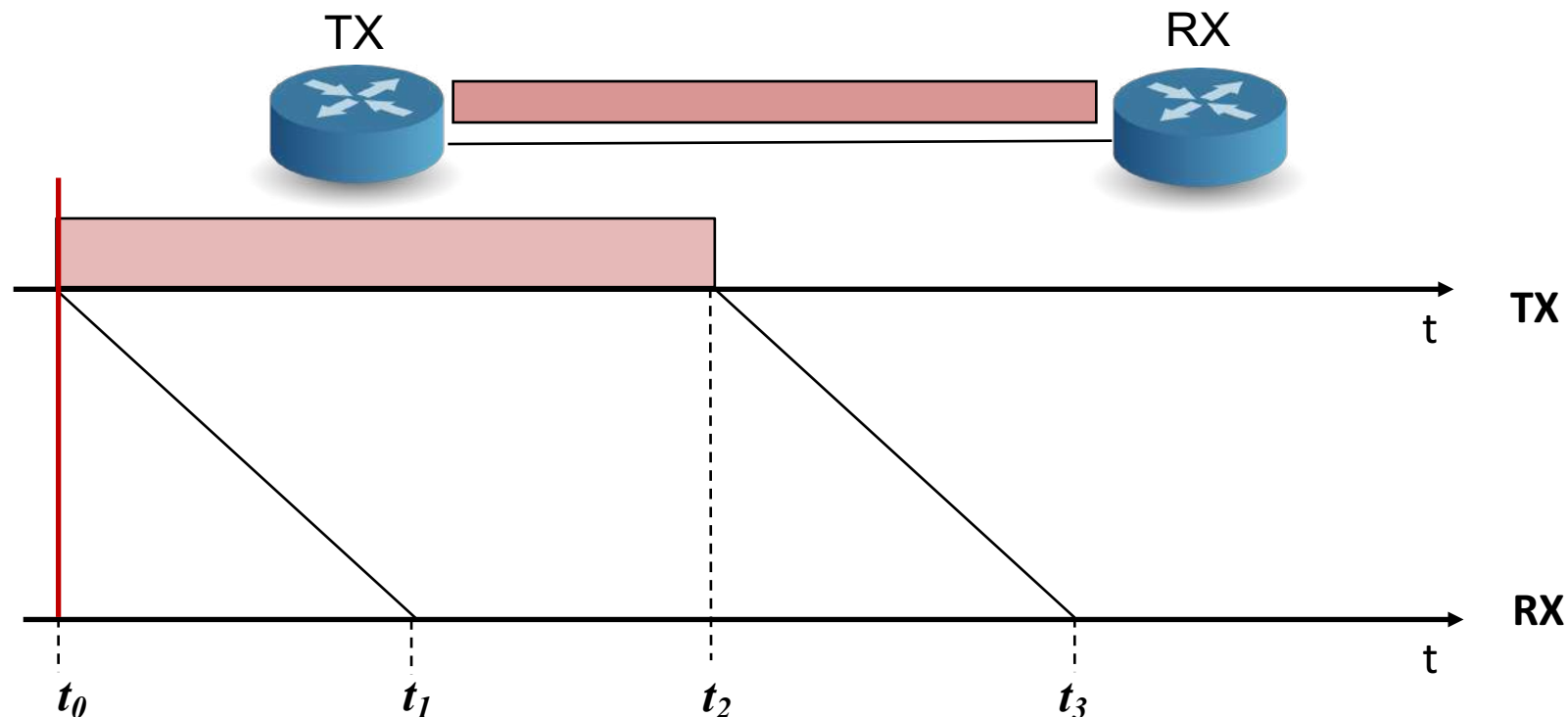
Tempi di attraversamento del canale



Ritardo fra la trasmissione del primo bit e la ricezione dell'ultimo $T_{tot} = T + \tau$



Tempi di attraversamento del canale



t_0 =inizio trasmissione

t_1 =arrivo primo bit

t_2 =fine trasmissione

t_3 =arrivo ultimo bit

Tempo di trasmissione:

$$T = t_2 - t_0 = t_3 - t_1 = L/R \quad L = \text{lunghezza del pacchetto [bit]}$$

R =velocità di trasm. [bit/s]

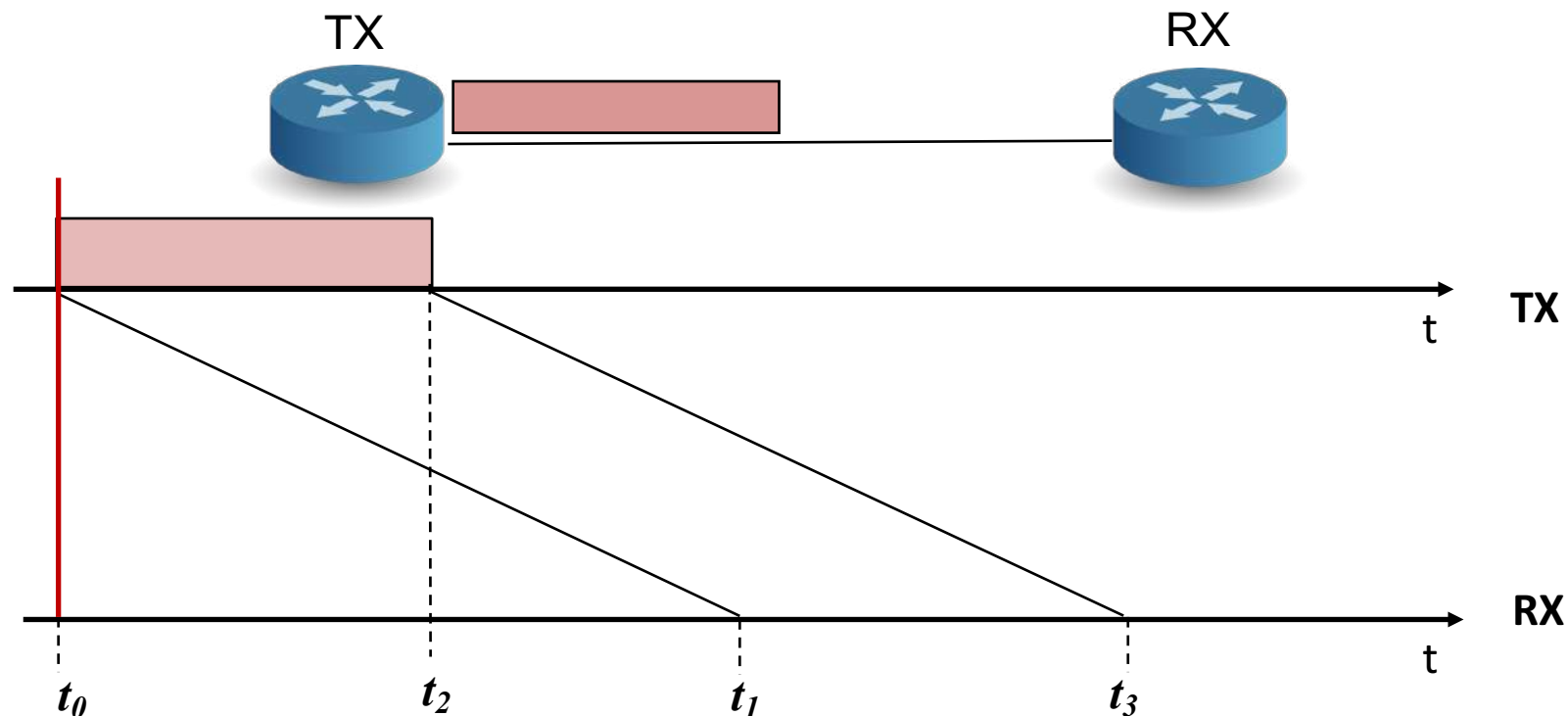
Ritardo di propagazione:

$$\tau = t_1 - t_0 = t_3 - t_2 = D/v \quad D = \text{lunghezza del coll. [m]}$$

v =velocità di prop. [m/s]



Tempi di attraversamento del canale



t_0 =inizio trasmissione

t_1 =arrivo primo bit

t_2 =fine trasmissione

t_3 =arrivo ultimo bit

Tempo di trasmissione:

$$T = t_2 - t_0 = t_3 - t_1 = L/R \quad L = \text{lunghezza del pacchetto [bit]}$$

R =velocità di trasm. [bit/s]

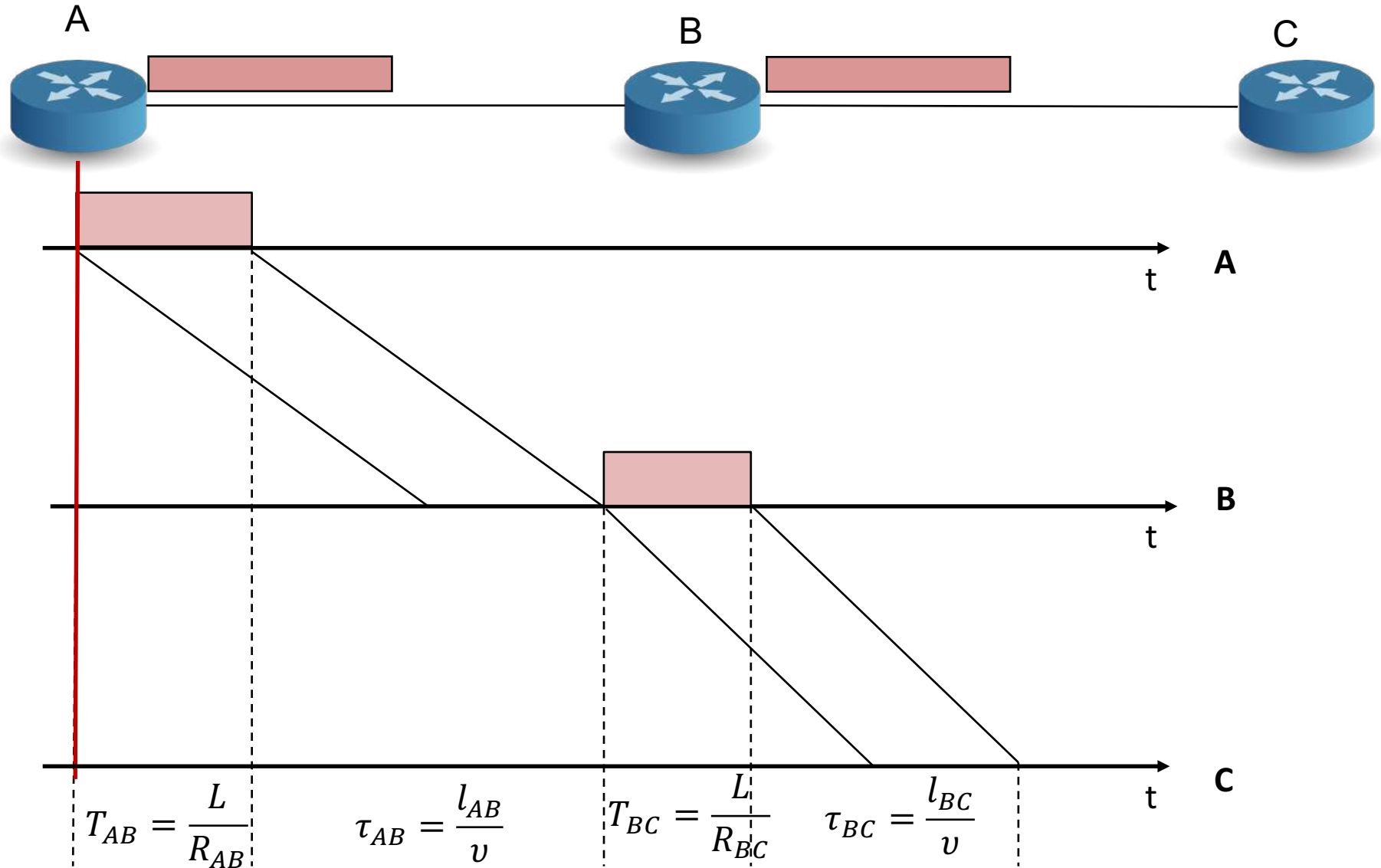
Ritardo di propagazione:

$$\tau = t_1 - t_0 = t_3 - t_2 = D/v \quad D = \text{lunghezza del coll. [m]}$$

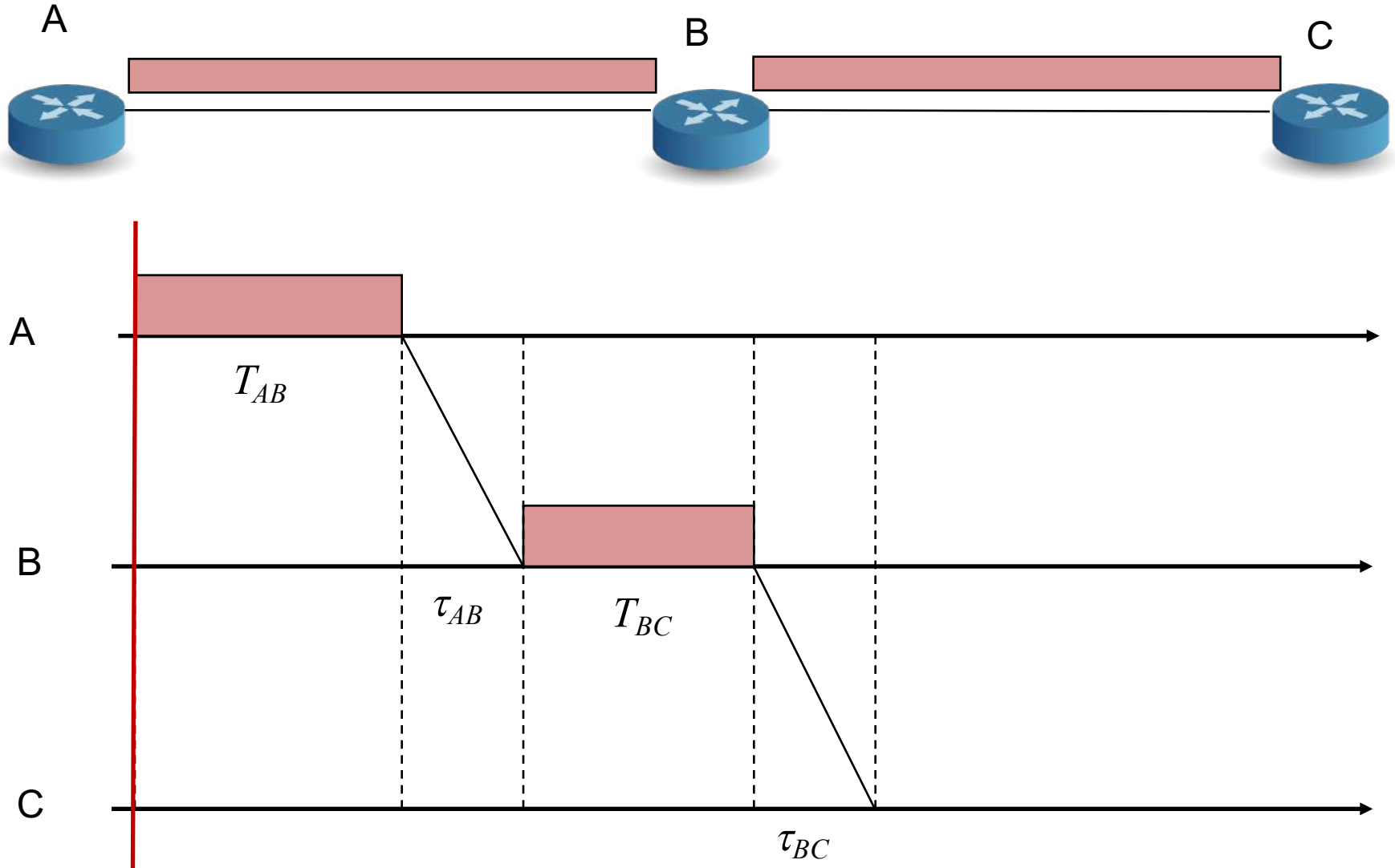
v =velocità di prop. [m/s]



Store and forward

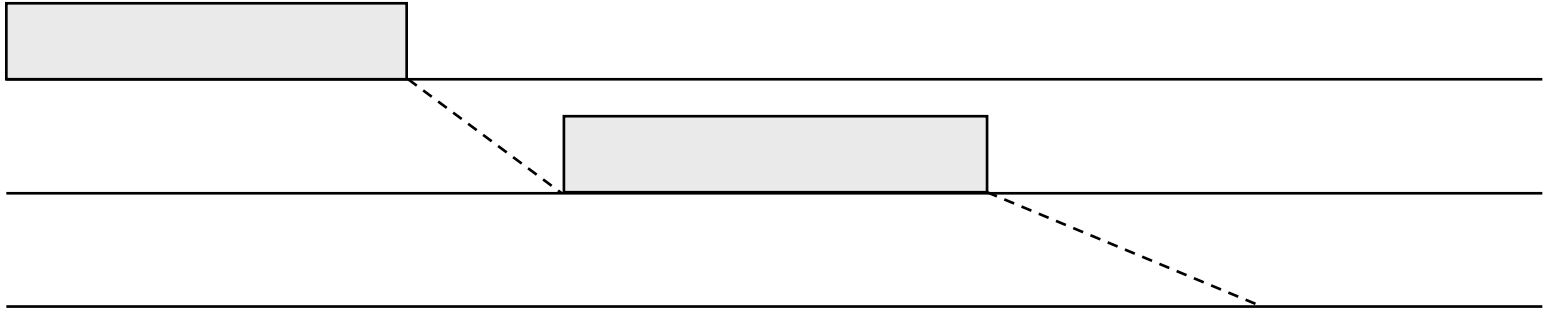


Store and forward

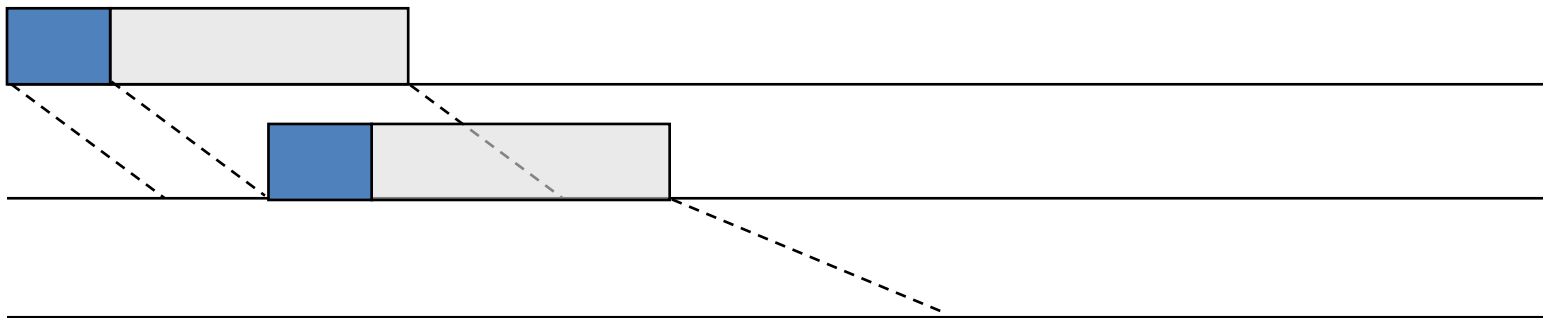


Commutazione a Pacchetto Cut-Through

- ***Store-and-forward***: il pacchetto deve essere completamente ricevuto prima di essere ritrasmesso

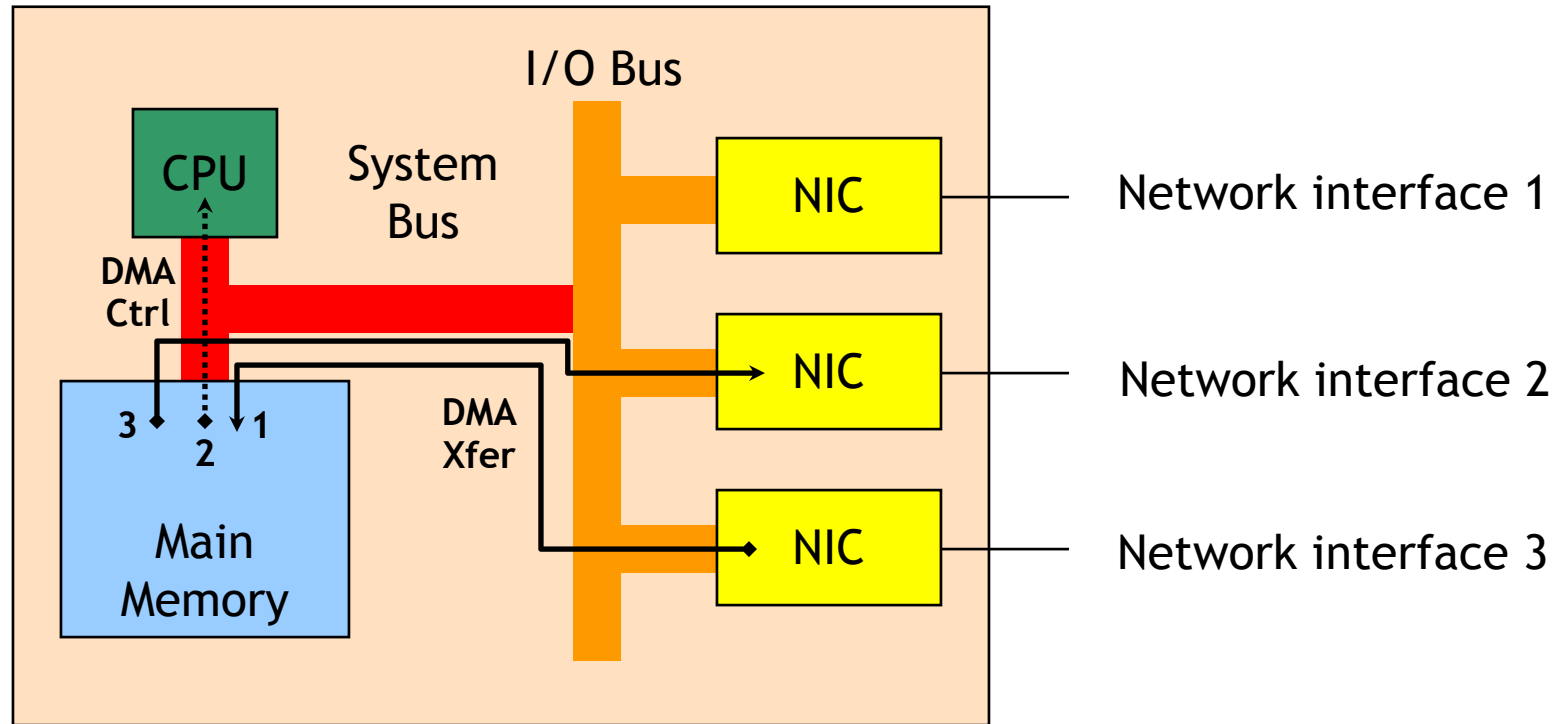


- ***Cut-Through***: il pacchetto viene ritrasmesso alla completa ricezione dell'header



Architettura semplificata di un nodo

Packet switch su architettura hardware general purpose (soft-switch)

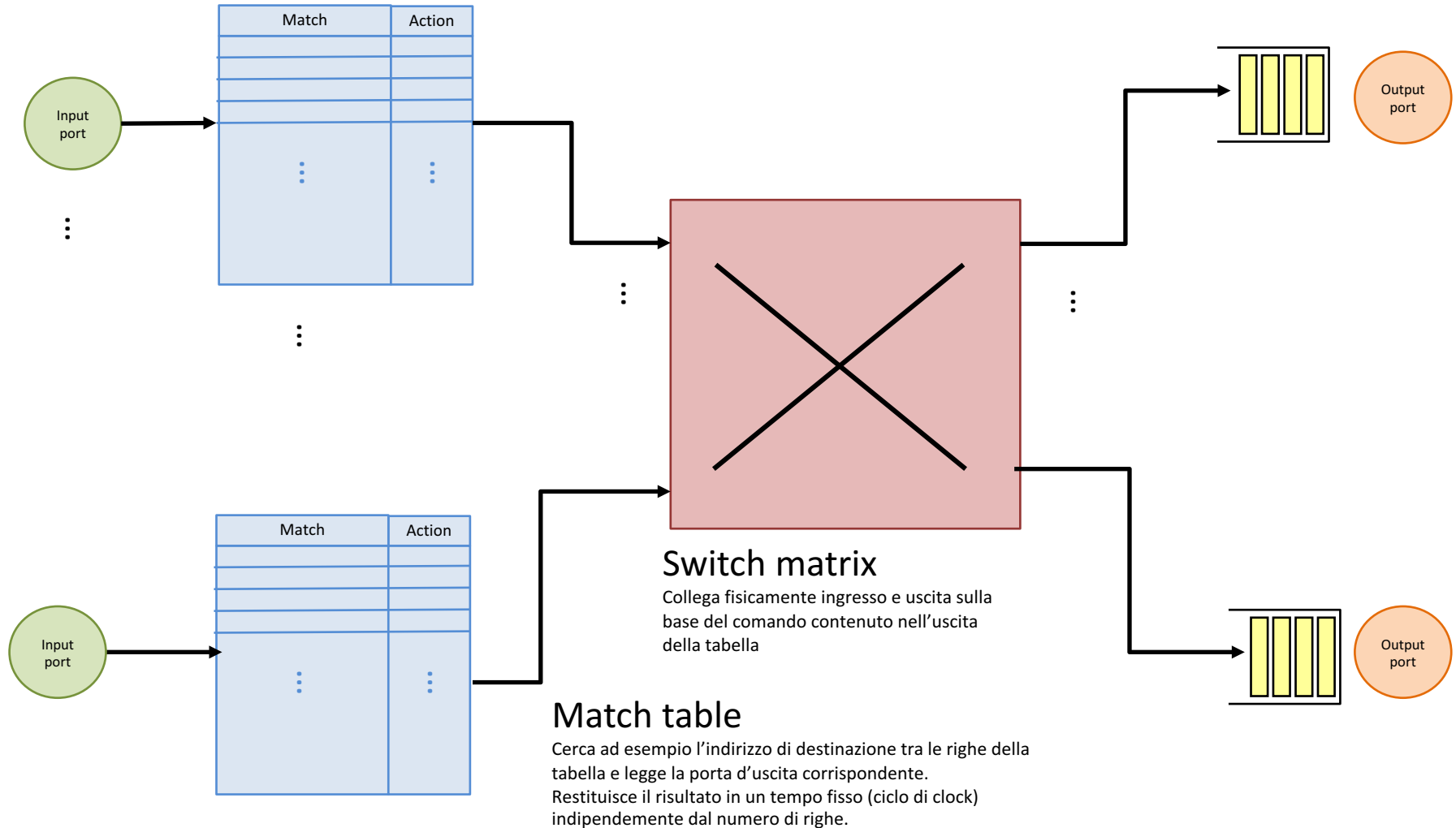


1. Packet input
2. Header processing
 - Routing table lookup
 - DMA transaction
3. Packet output

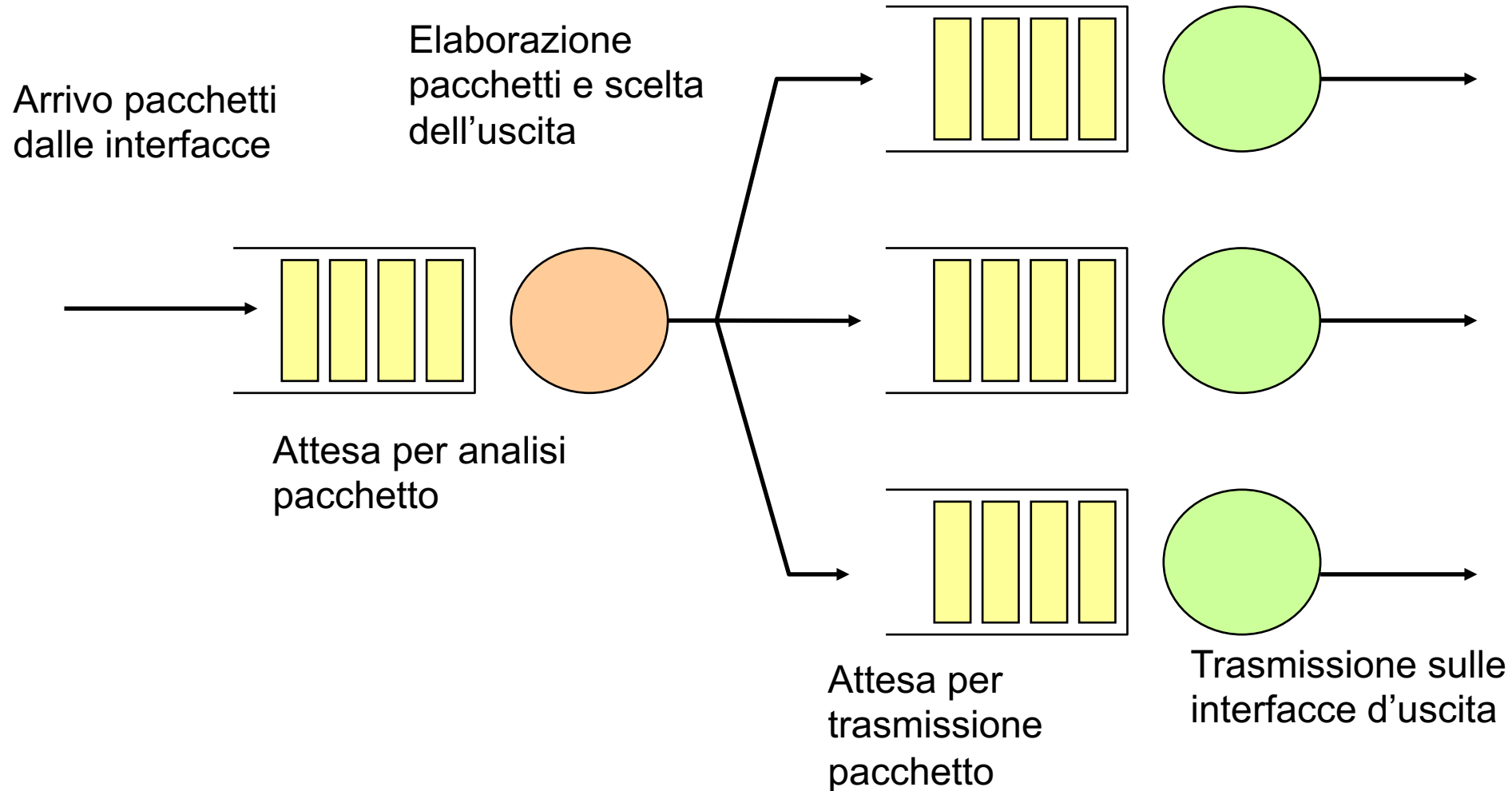
NIC = Network Interface Controller
DMA = Direct Memory Access

Architettura semplificata di un nodo

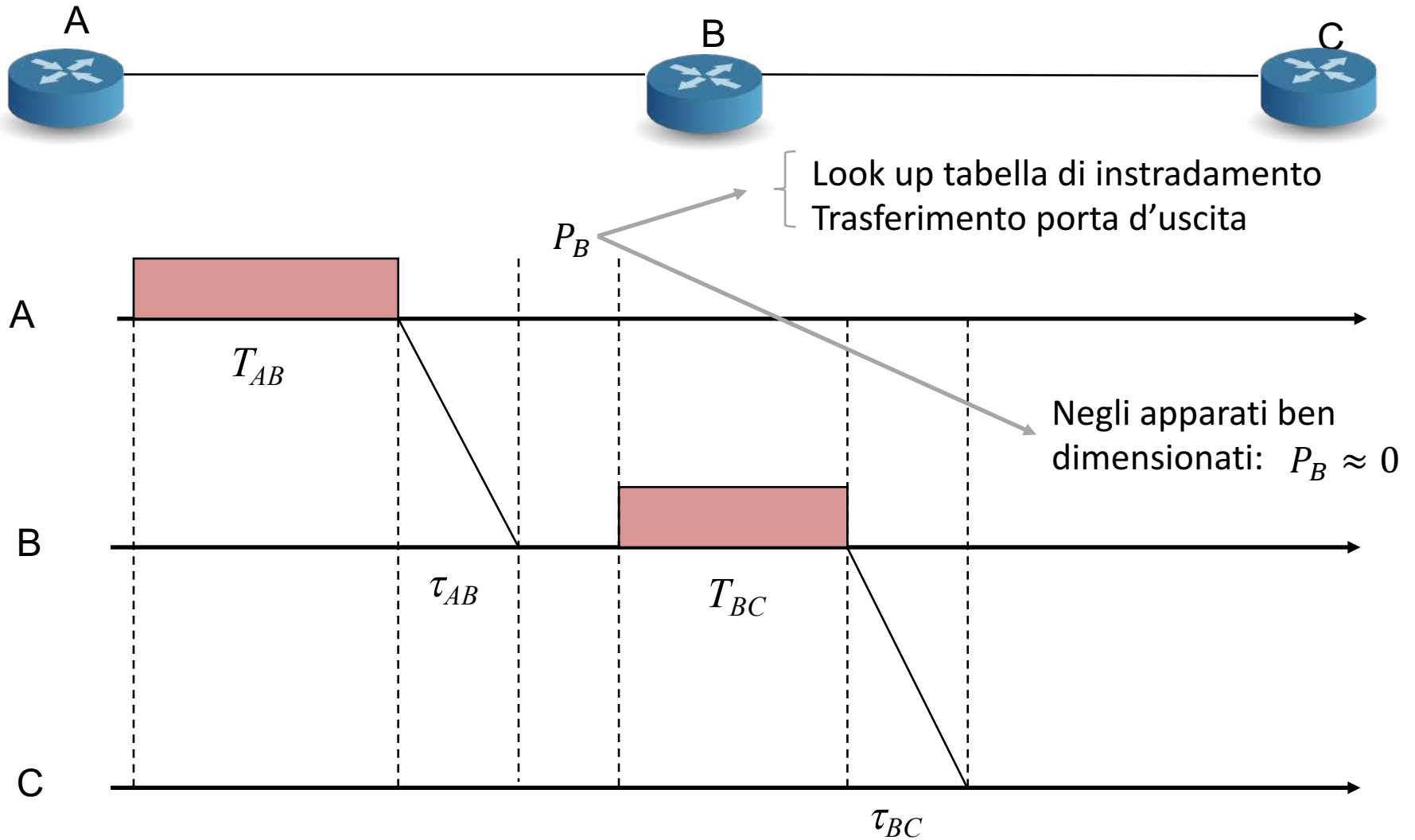
Packet switch su architettura hardware dedicata (hardware-switch)



Modello di un nodo



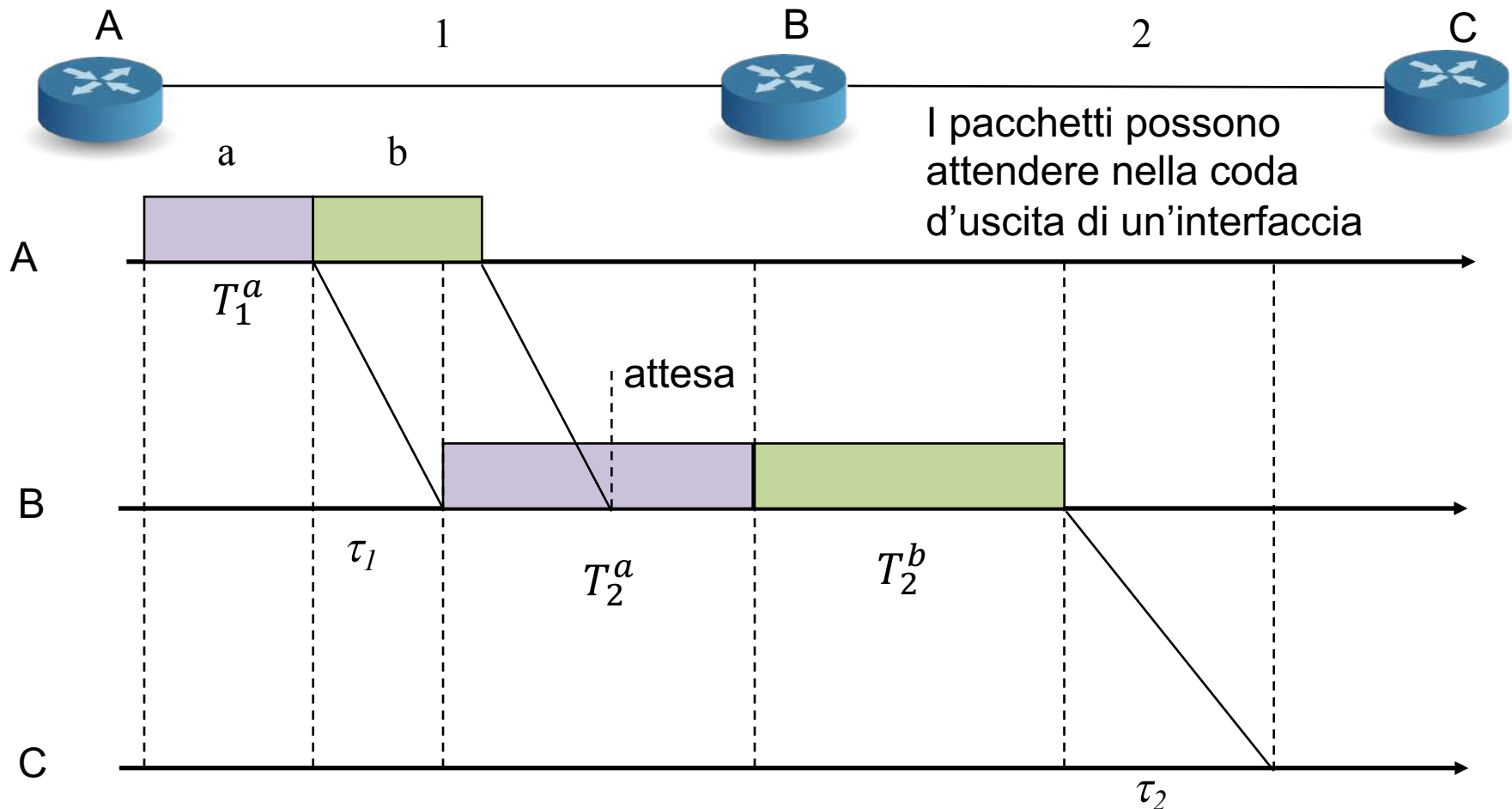
Tempo di elaborazione



Ritardo di accodamento

- Se la linea di uscita è occupata occorre aspettare in coda

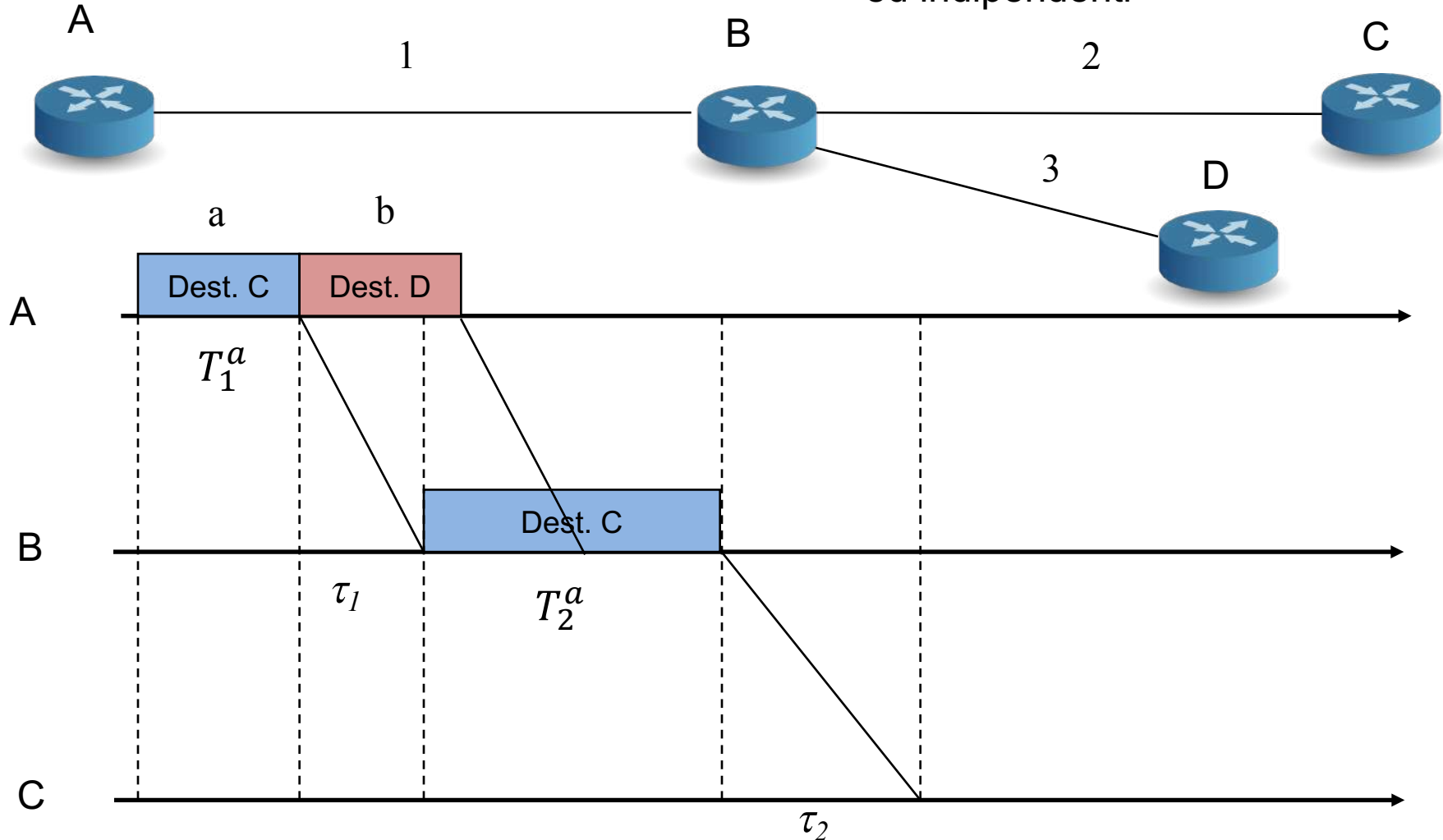
Esempio 1



Ritardo di accodamento

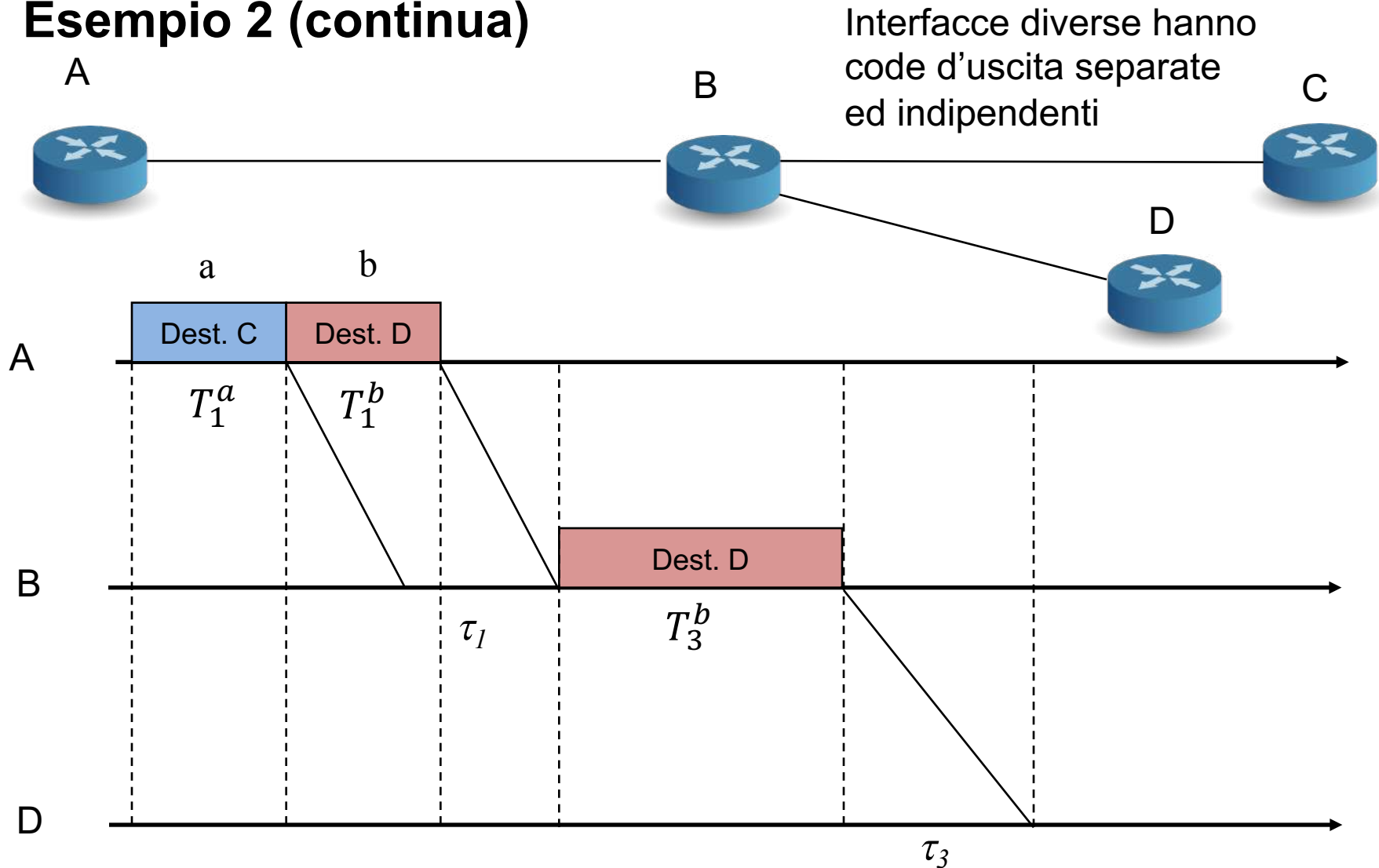
Esempio 2

Interfacce diverse hanno
code d'uscita separate
ed indipendenti



Ritardo di accodamento

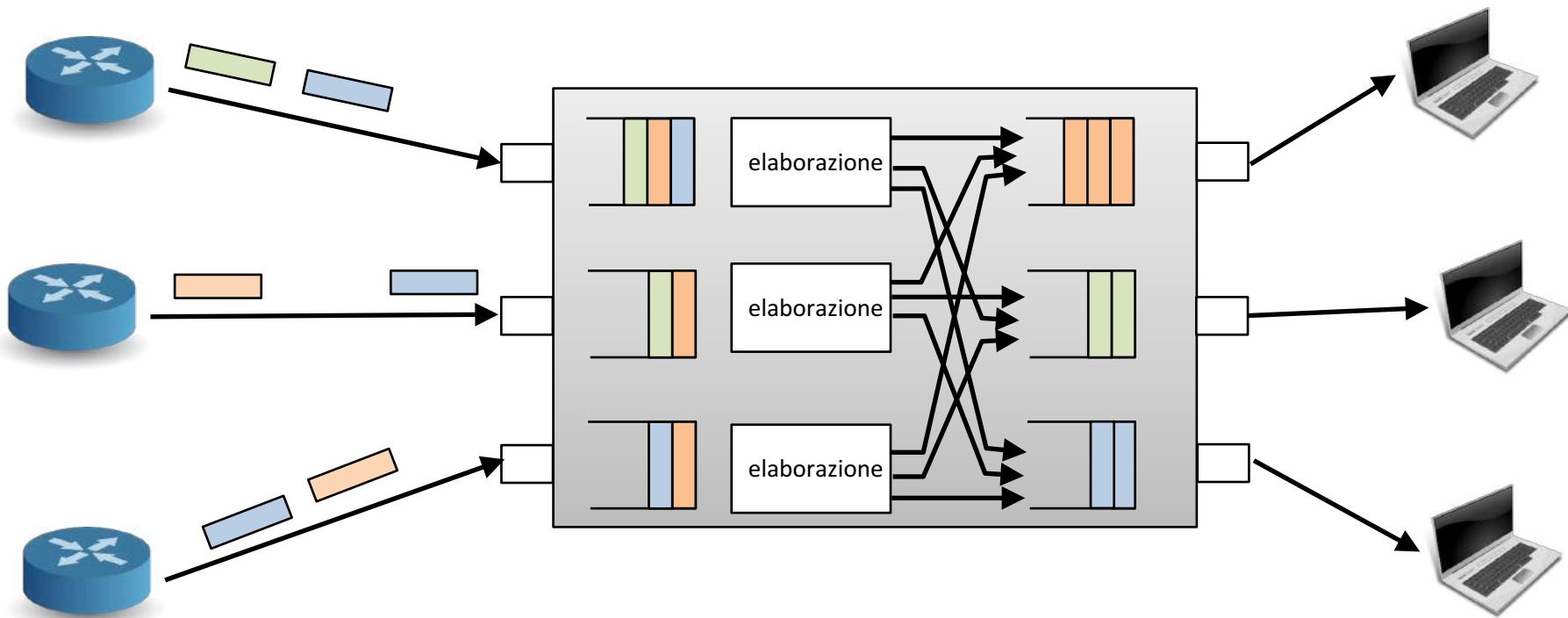
Esempio 2 (continua)



Ritardo di accodamento

Multiplazione statistica

Più in generale il **ritardo di accodamento** dipende dalla **multiplazione statistica** dovuto all'**arrivo asincrono dei pacchetti** alle code d'uscita (trasmissione)

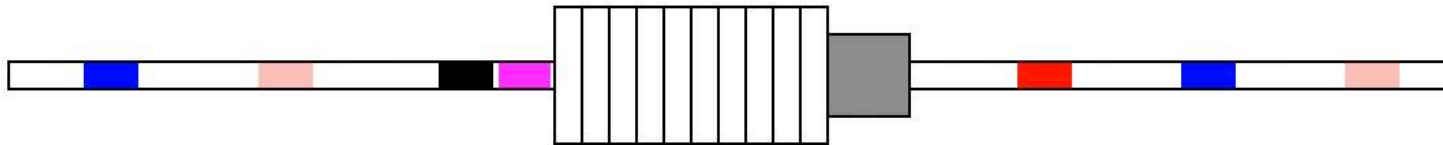


Ritardo di accodamento

Multiplazione statistica

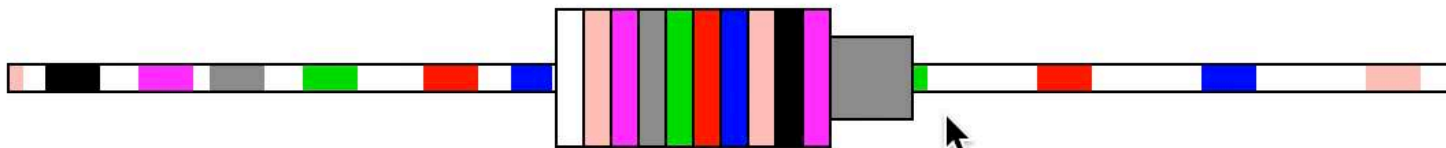
Più in generale il **ritardo di accodamento** dipende dalla **multiplazione statistica** dovuto all'arrivo asicrono dei pacchetti alle code d'uscita (trasmissione)

Emission rate 350 packet/s Transmission rate 350 packet/s Start Reset



112 msec
0 packets dropped out of 32

Emission rate 500 packet/s Transmission rate 350 packet/s Start Reset



84 msec
1 packets dropped out of 35



Ritardo di Accodamento

Del **ritardo di accodamento medio** T_a si possono fare dei modelli statistici basati sulla **teoria delle code**:

R = velocità del link [b/s]

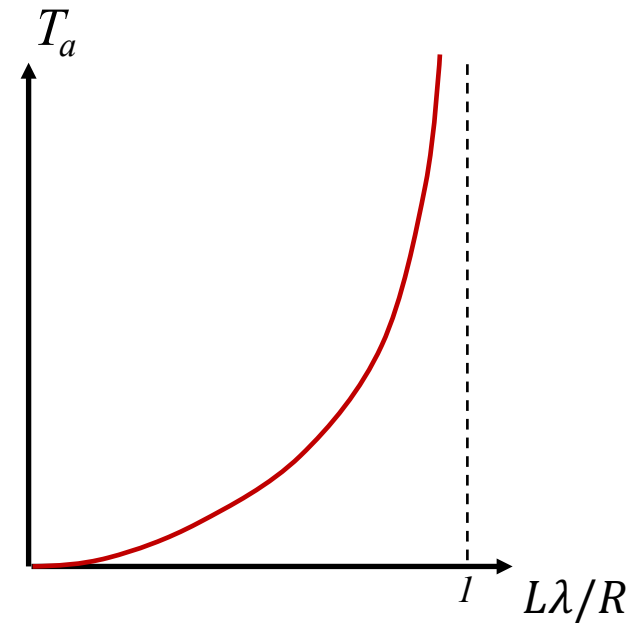
L = lunghezza pacchetto [bits]

λ = frequenza di arrivo dei pacchetti [pack/s]

Intensità di traffico = $L\lambda/R \sim 0$

$L\lambda/R \sim 0$: ritardo in coda piccolo

$L\lambda/R \rightarrow 1$: il ritardo tende all'infinito



Ritardo di Accodamento

Del **ritardo di accodamento medio** T_a si possono fare dei modelli statistici basati sulla **teoria delle code**:

R = velocità del link [b/s]

L = lunghezza pacchetto [bits]

λ = frequenza di arrivo dei pacchetti [pack/s]

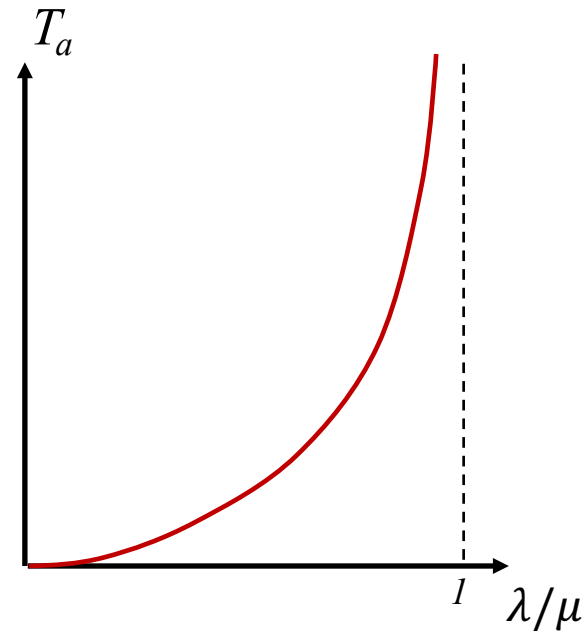
Definiamo:

μ = frequenza di trasmissione dei pacchetti [pack/s]

Si ha:

$$\mu = \frac{R}{L}$$

Si può mostrare che sotto alcuni condizioni sulla statistica degli arrivi e la distribuzione delle lunghezze dei pacchetti:

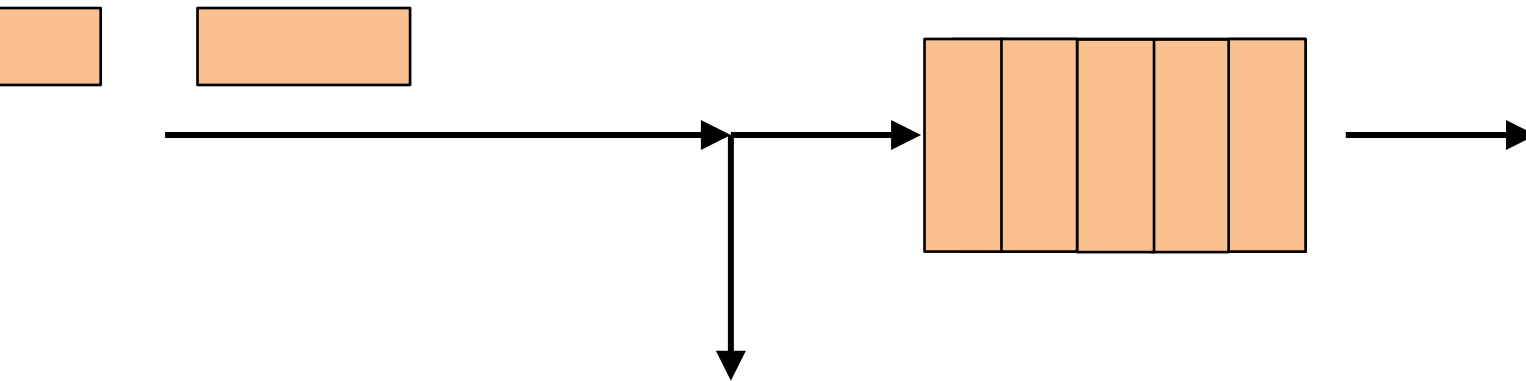


$$T_a = \frac{1}{\mu - \lambda} - \frac{1}{\mu}$$



Perdite di pacchetti in Internet

- Le code hanno **dimensioni limitate**
- In **congestione** (coda piena) i pacchetti che arrivano vengono **scartati**
- I pacchetti persi possono essere **ritrasmessi** o meno a seconda del livello/protocollo che gestisce l'evento di perdita (vedremo esempi a livello di linea e di trasporto).



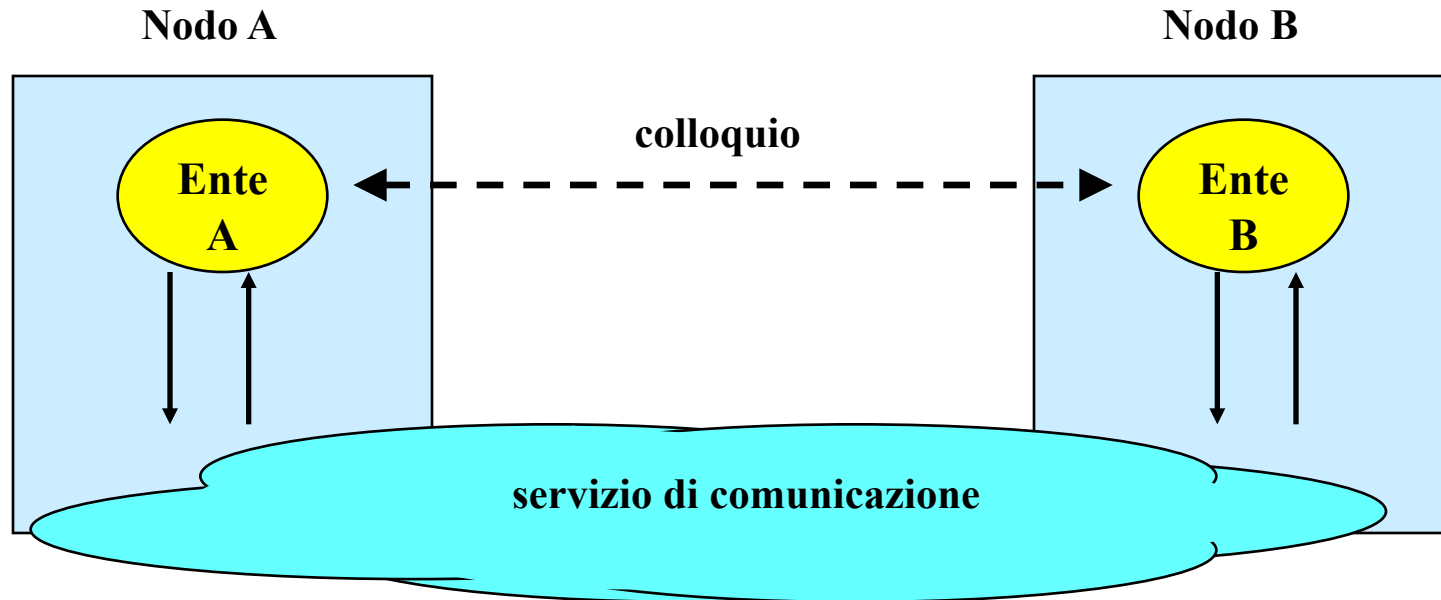


1d – Modelli di servizio e livelli dei protocolli

Servizio di comunicazione, protocolli,
primitive di servizio, architetture a livelli,
funzioni di rete

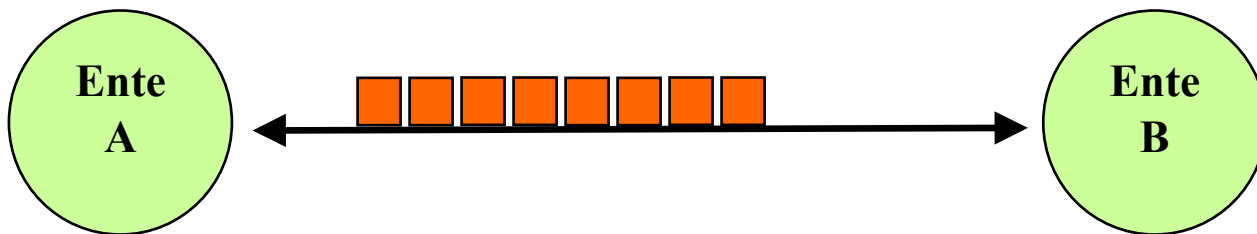
Il servizio di comunicazione

- Date due o più entità remote
- Possiamo descrivere il servizio di comunicazione per scambio di messaggi come un
“fornitore del servizio di trasporto dell’informazione”



Il servizio di comunicazione

- **Gestisce lo scambio di informazione fra due “entità”**
- **E' in generale un servizio di trasferimento di unità informative:**
 - Messaggi applicativi (richieste e risposte di browser e server web, messaggi di email, file, ecc.)
 - Flussi multimediali (flussi audio, video, ecc.)
 - Gruppi di bit (pacchetti)
 - Bit



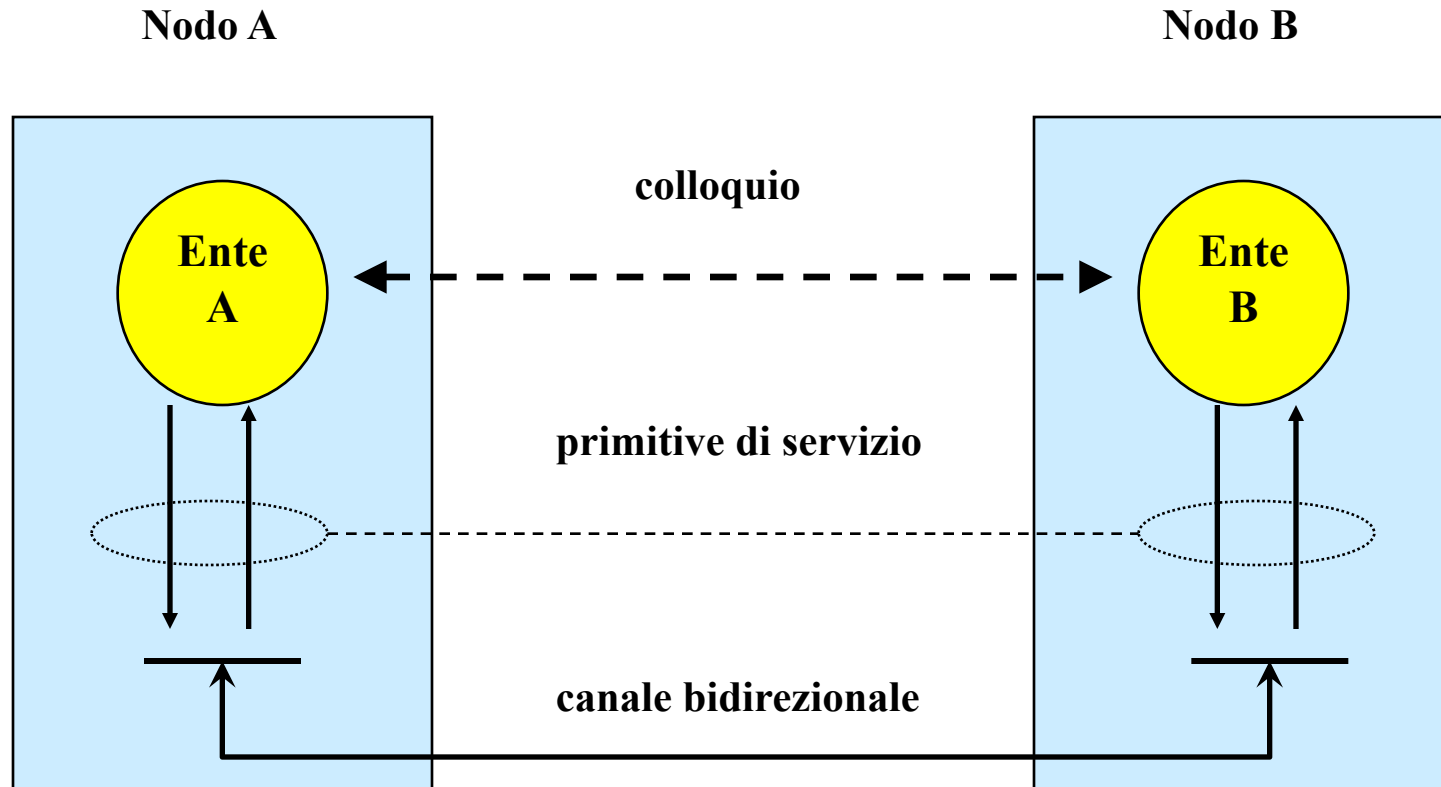
Primitive di servizio

- il servizio di comunicazione può essere descritto mediante delle *chiamate di servizio* dette **primitive di servizio**
- le primitive di servizio servono a descrivere il servizio, a richiederlo e a ricevere informazioni sul servizio dal fornitore
- le primitive di servizio sono caratterizzate da parametri tra cui:
 - informazione da trasferire
 - indicazione del destinatario
 - caratteristiche del servizio richiesto
 - ecc.

NOTA: le primitive di servizio dell'interfaccia socket in Python saranno trattate in laboratorio, dove vedrete meccanismi per creare le interfacce e passare i parametri tramite funzione.

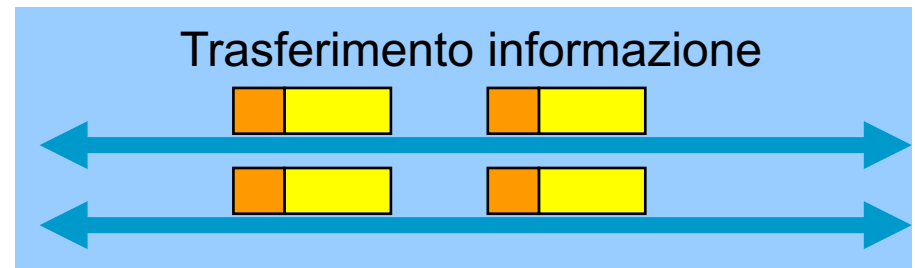
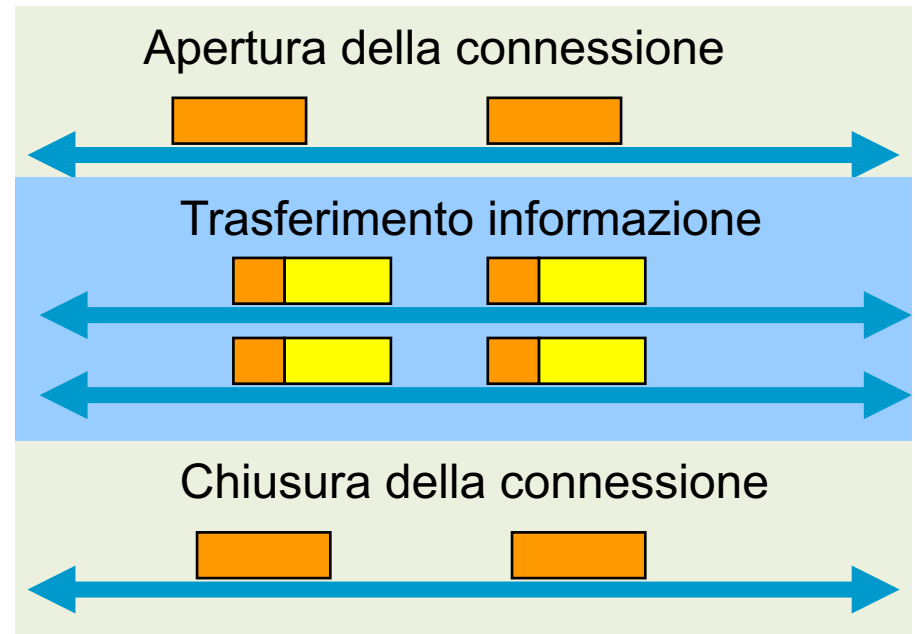


Primitive di servizio



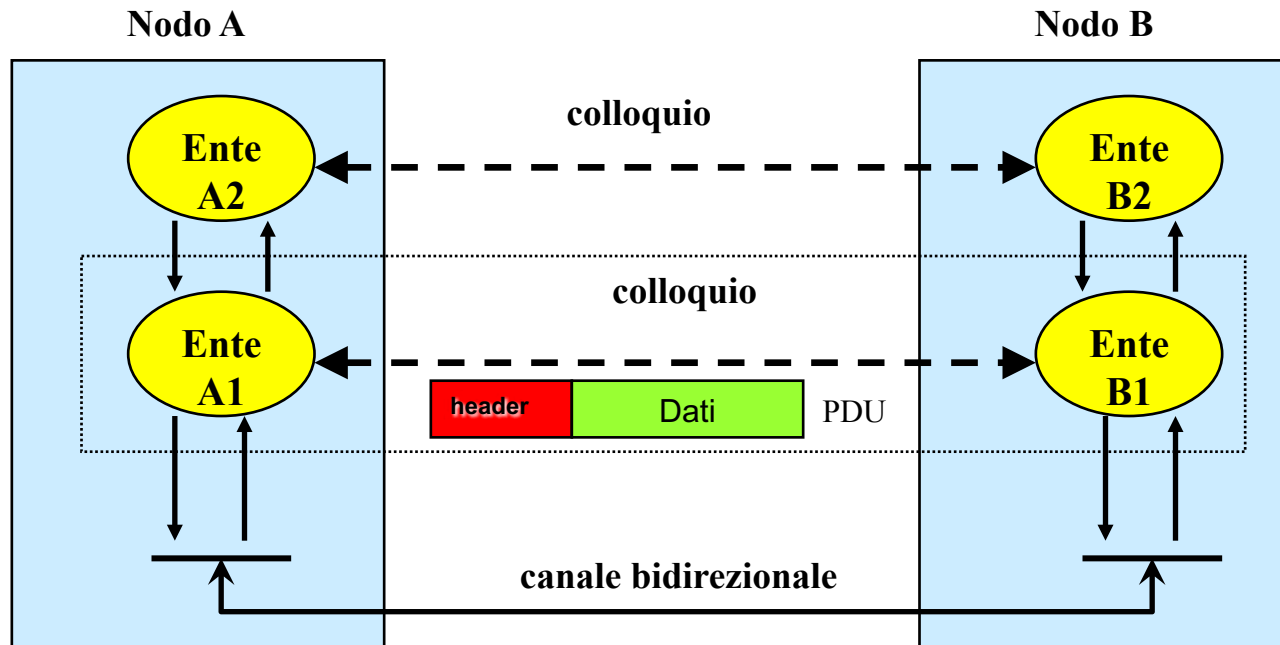
Caratteristiche del servizio di comunicazione

- **modalità a connessione**
 - instaurazione della connessione
 - trasferimento dell'informazione
 - rilascio delle connessione
- **modalità senza connessione**
 - una sola fase



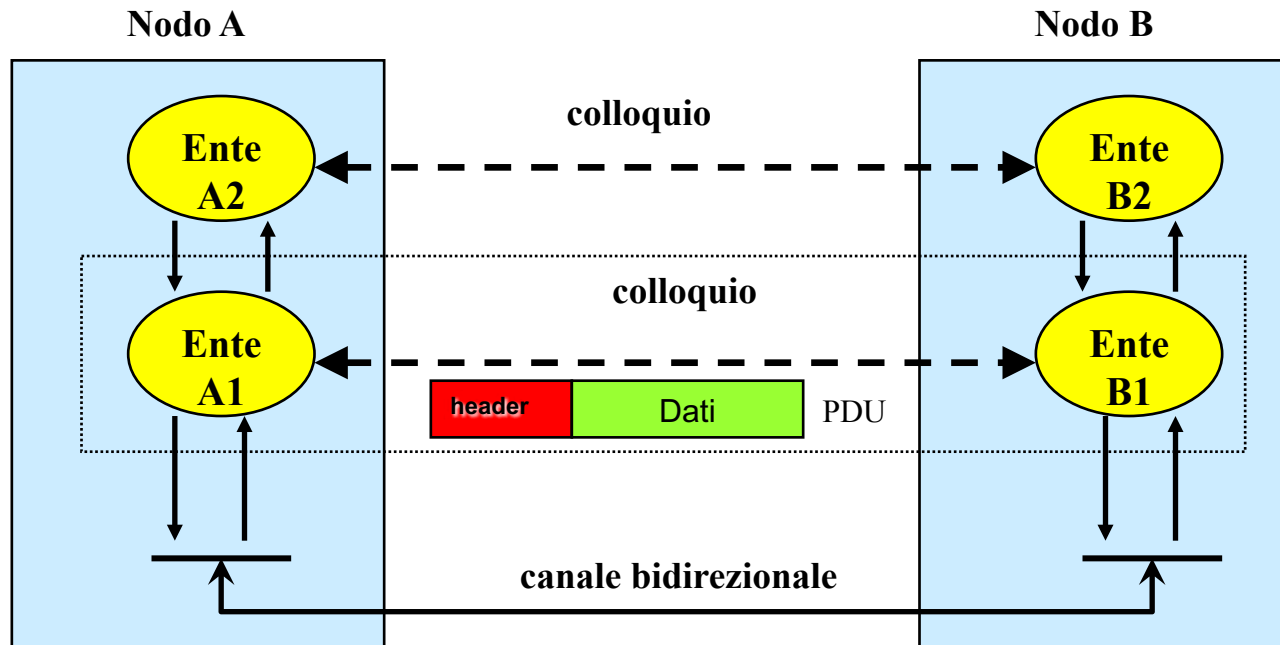
Livelli

- Le entità che colloquiano in un servizio di telecomunicazione possono anche offrire un servizio di comunicazione a entità terze, dette di **livello superiore**



Funzioni dei livelli

- Il servizio di comunicazione offerto al livello superiore è più ricco e complesso grazie alle **funzioni** implementate dal livello inferiore



Protocolli di comunicazione

- Le entità di un livello collaborano per fornire il servizio di comunicazione al livello superiore e si scambiano messaggi mediante il servizio offerto dal livello inferiore
- **Protocollo:**
 - Insieme delle regole che gestiscono il colloquio tra entità dello stesso livello
 - formato dei messaggi
 - informazioni di servizio
 - algoritmi di trasferimento
 - ecc.



Packet Data Units (PDU)

- Un protocollo utilizza per il colloquio tra entità dello stesso livello delle **unità di trasferimento dati** dette PDU o anche trame del protocollo
- Le PDU possono contenere:

informazione di servizio
necessaria al
coordinamento tra le
entità

informazione vera e
propria ricevuta dai
livelli superiori

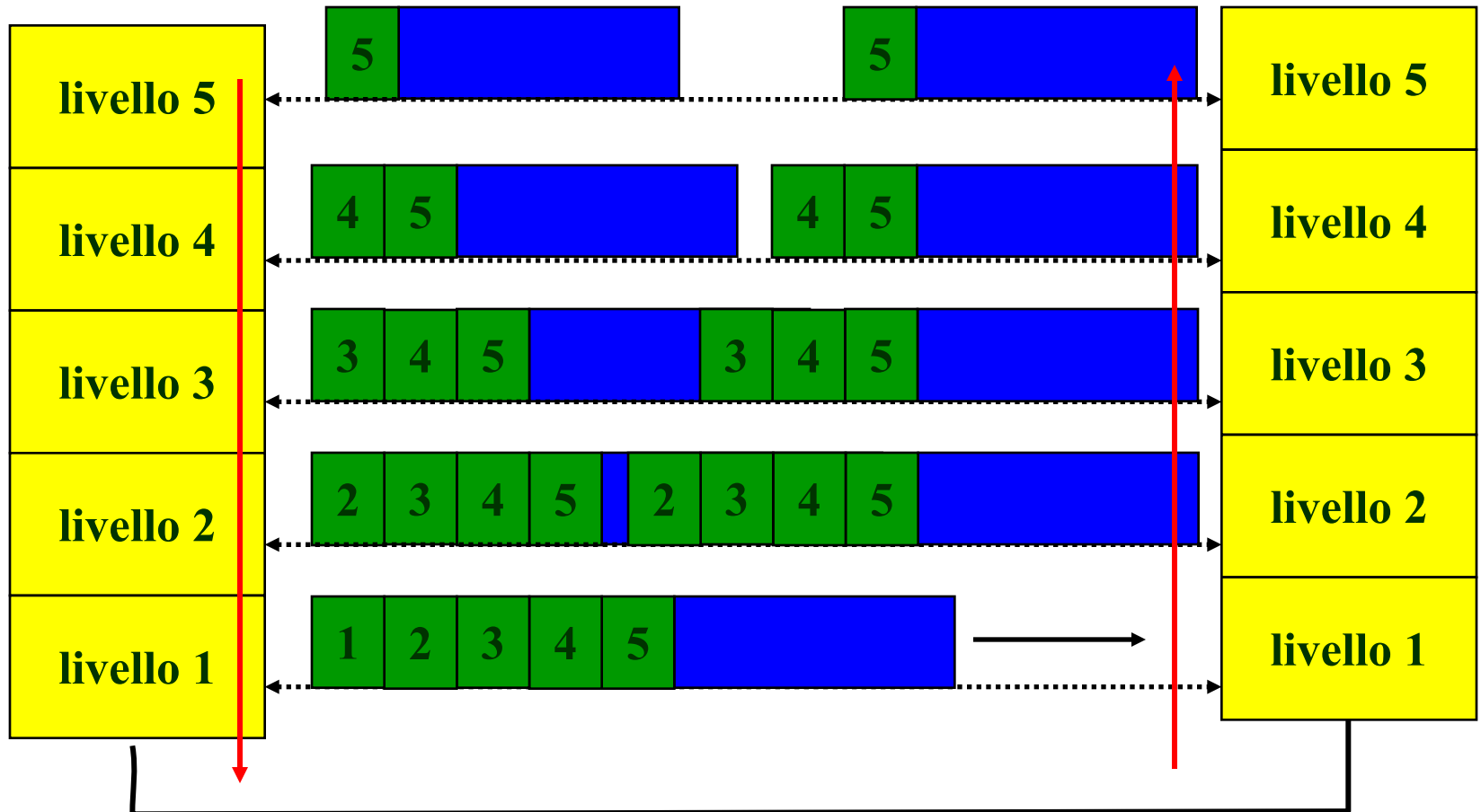


Architettura a livelli

- **I servizi di comunicazione complessi possono essere articolati a livelli**
 - da un livello che garantisce solo il trasporto dei bit
 - a un livello dove sono definite complessi servizi caratterizzati da molti parametri e funzionalità



Architettura completa



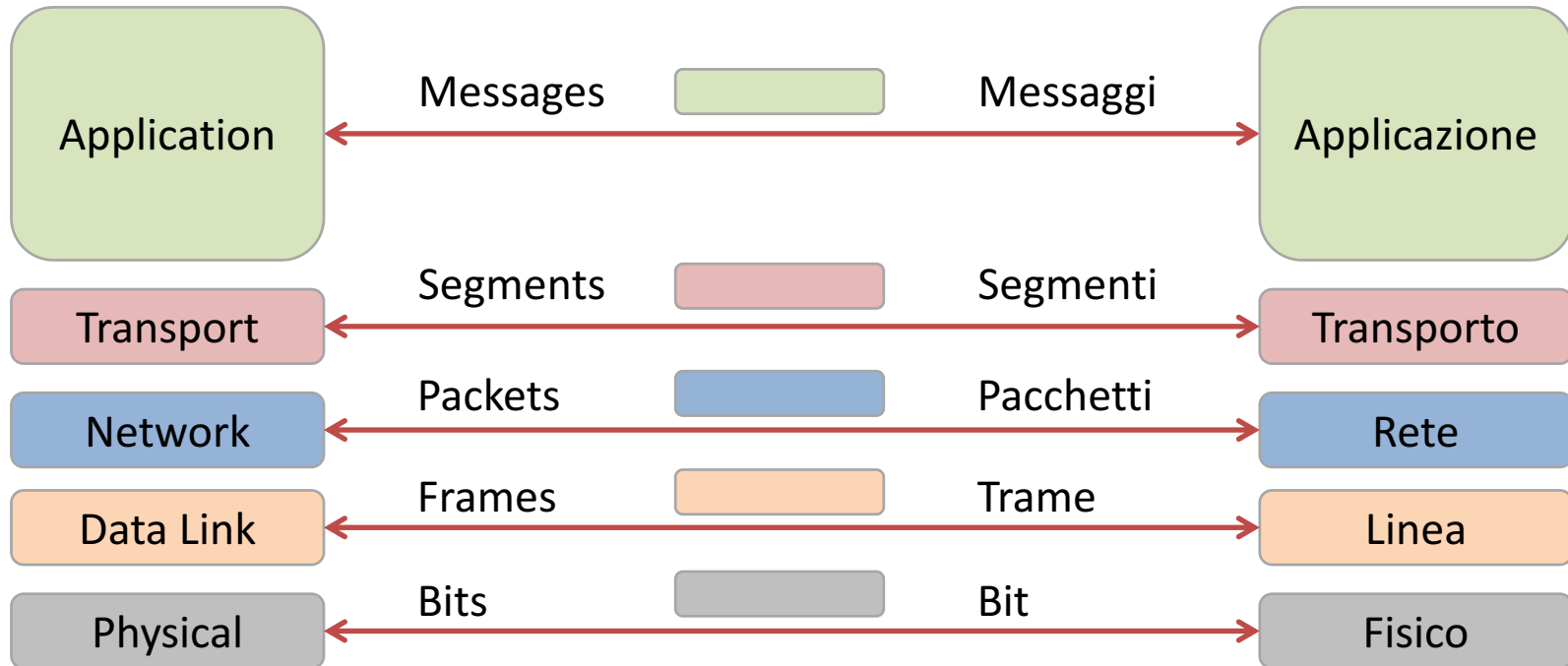
Perchè un'architettura a livelli?

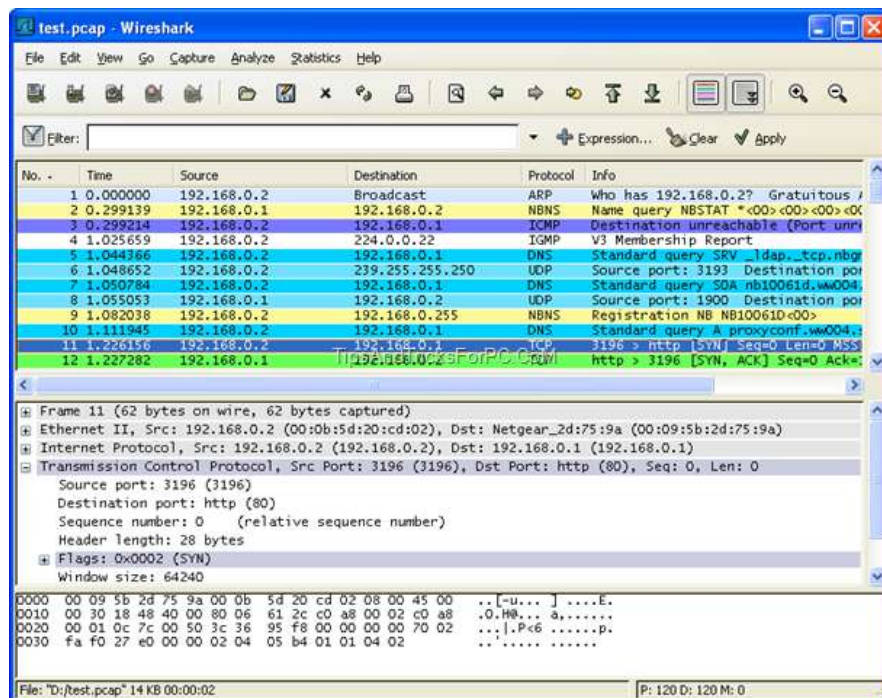
Sistemi complessi:

- **Facile identificazione dei servizi (implementazione, discussione)**
- **Facile gestione ed update**
 - Cambiamenti in un livello sono trasparenti agli altri
- **Q: quando la suddivisione in livelli può essere pericolosa?**



Modello a livelli di Internet (TCP/IP)





Passare ad attività con wireshark per analisi livelli protocollari



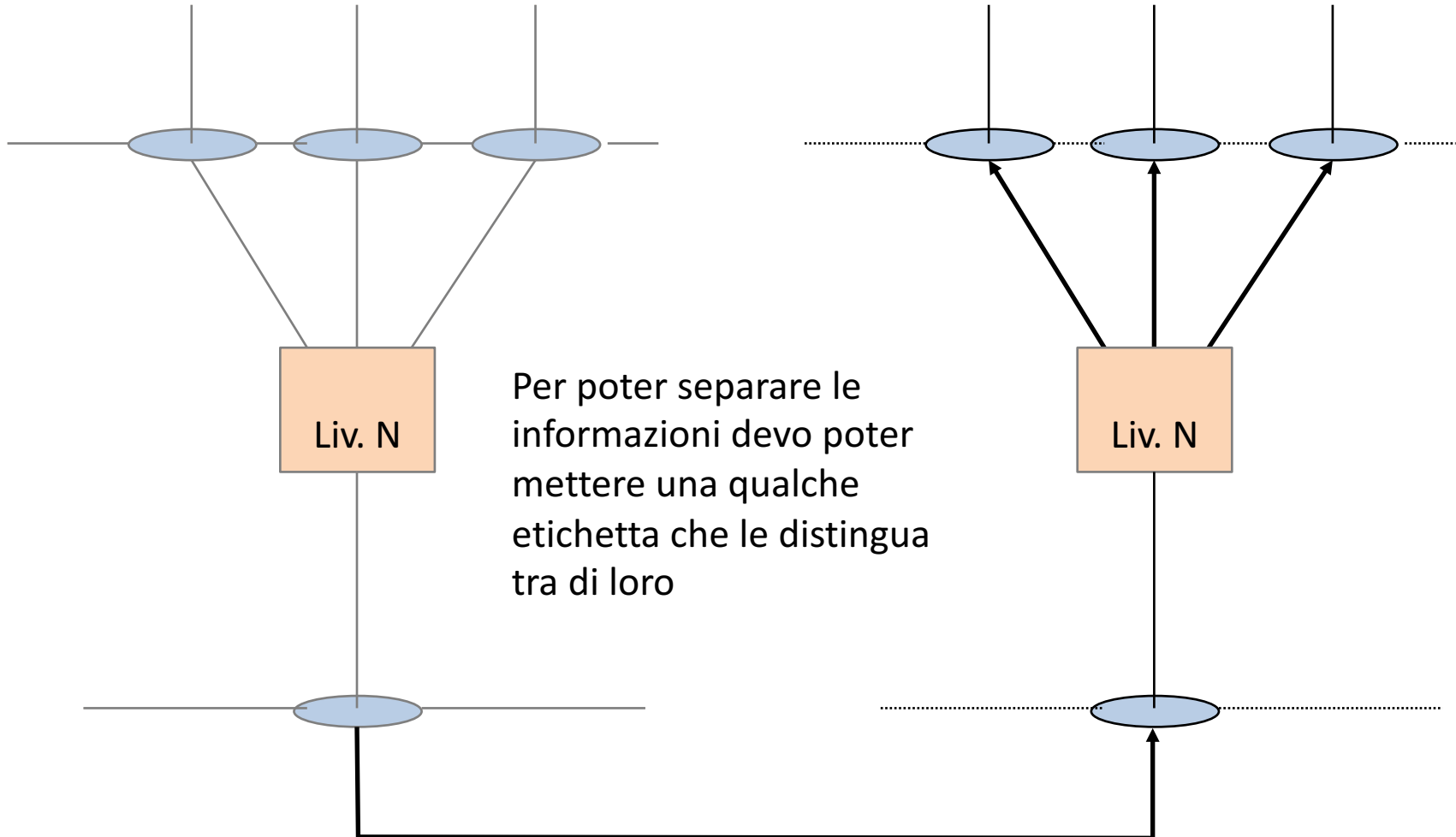
Funzioni

- **Molteplici sono le funzioni che possono essere svolte da uno livello**
- **Vediamo alcuni esempi importanti concentrandoci sui principi di base, poi i dettagli saranno ripresi più avanti**
 - Esempi:
 - Funzione di Multiplazione e De-multiplazione
 - Funzione di controllo d'errore
 - Funzione di instradamento



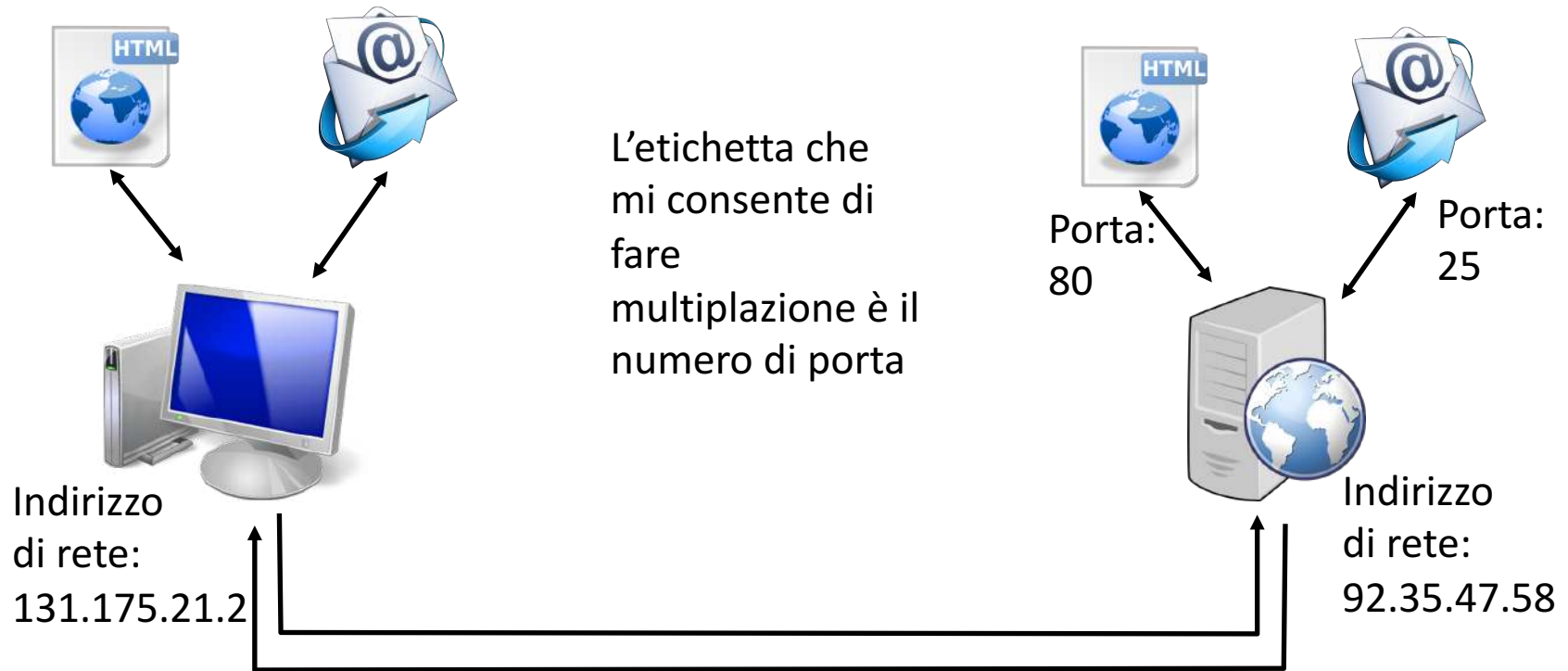
Esempio: Funzione Multiplazione

- Più livelli superiori possono condividere lo stesso servizio di comunicazione



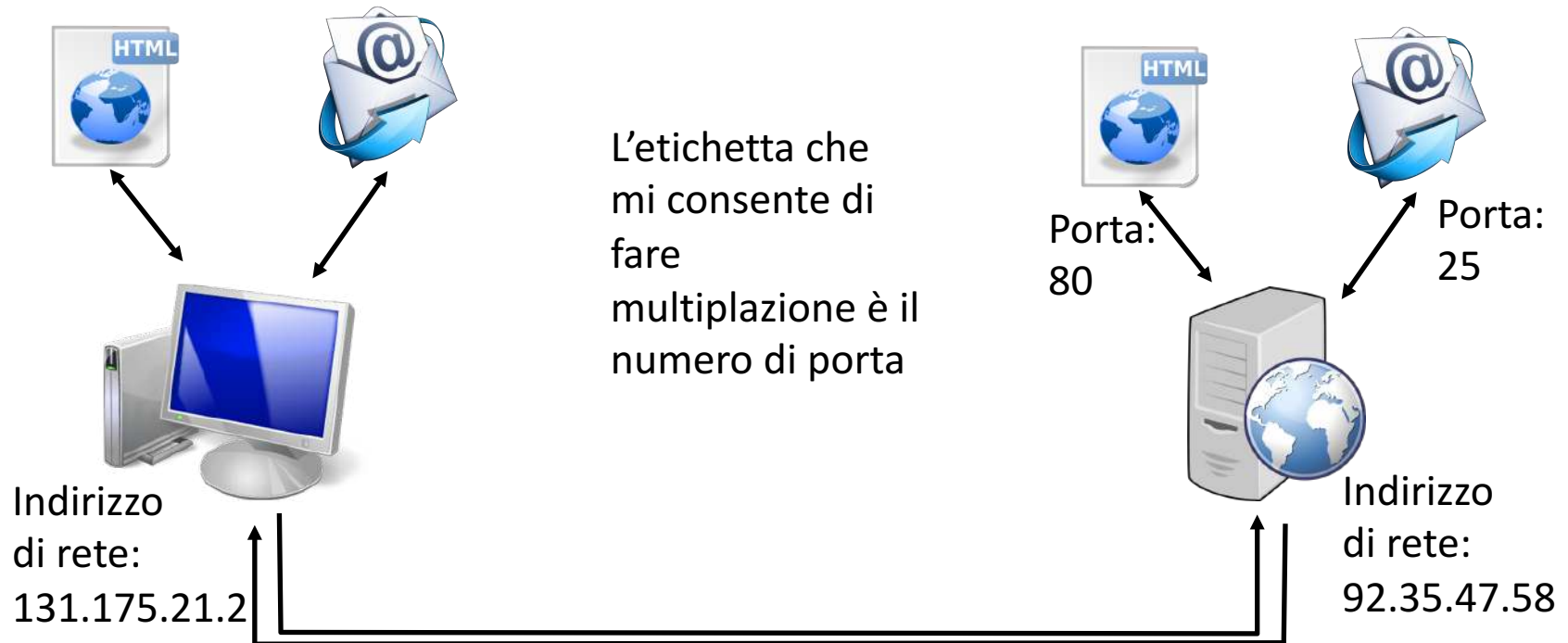
Esempio: Funzione Multiplazione

- Più applicazioni condividono la stessa interfaccia di rete (identificata da un suo indirizzo di rete)
- Le applicazioni sono distinte in base a un numero di porta



Esempio: Funzione Multiplazione

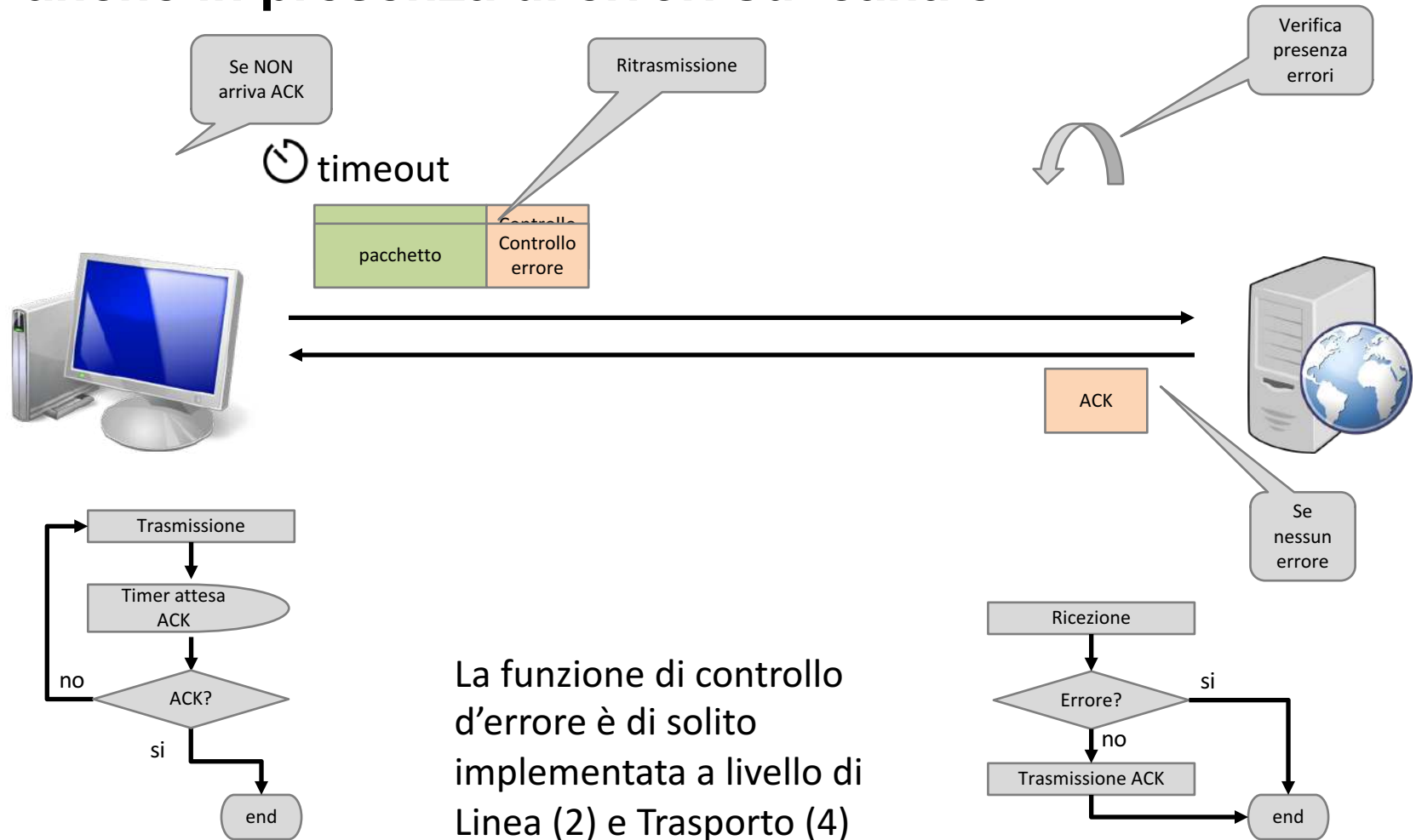
- Più applicazioni condividono la stessa interfaccia di rete (identificata da un suo indirizzo di rete)
- Le applicazioni sono distinte in base a un numero di porta



La funzione di multiplazione può essere implementata in tutti i livelli

Esempio: Funzione di controllo d'errore

- E' possibile garantire affidabilità delle comunicazioni anche in presenza di errori sul canale

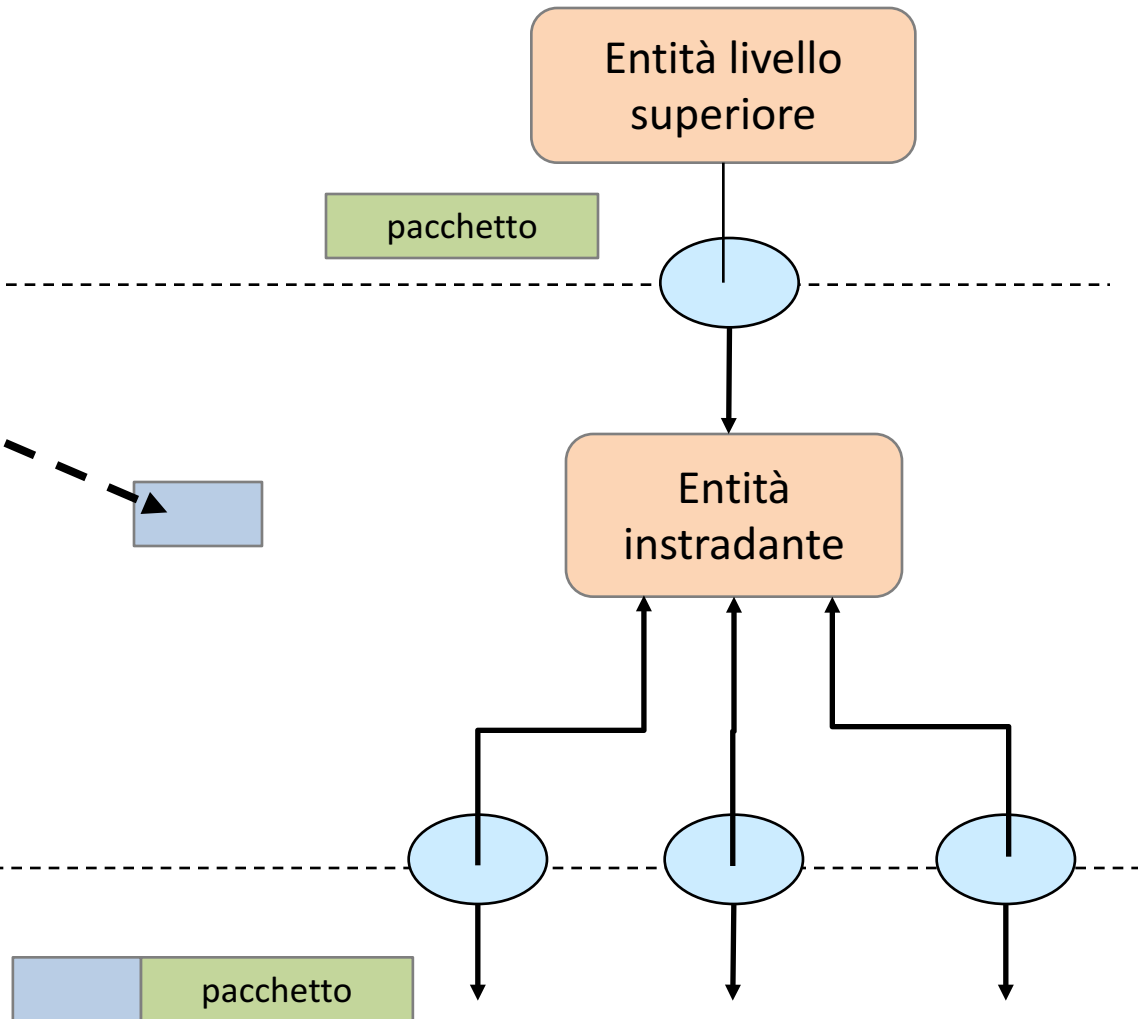


Esempio: Funzione di instradamento (routing)

Il pacchetto può arrivare dal livello superiore passato col parametro **INDIRIZZO**

L'indirizzo viene scritto nell'header perché possa essere instradato da altri nodi

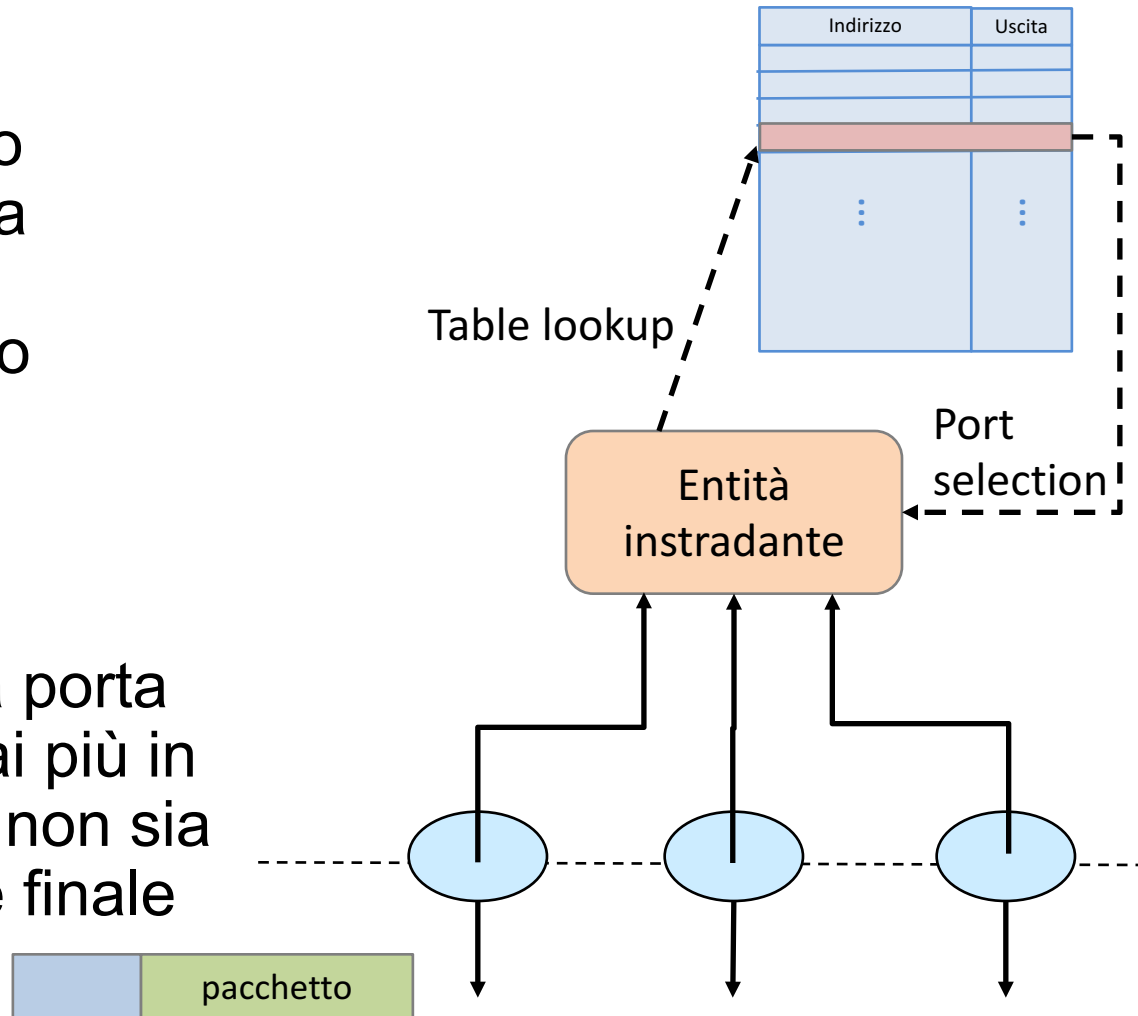
Il pacchetto può arrivare da una porta d'ingresso



Esempio: Funzione di instradamento (routing)

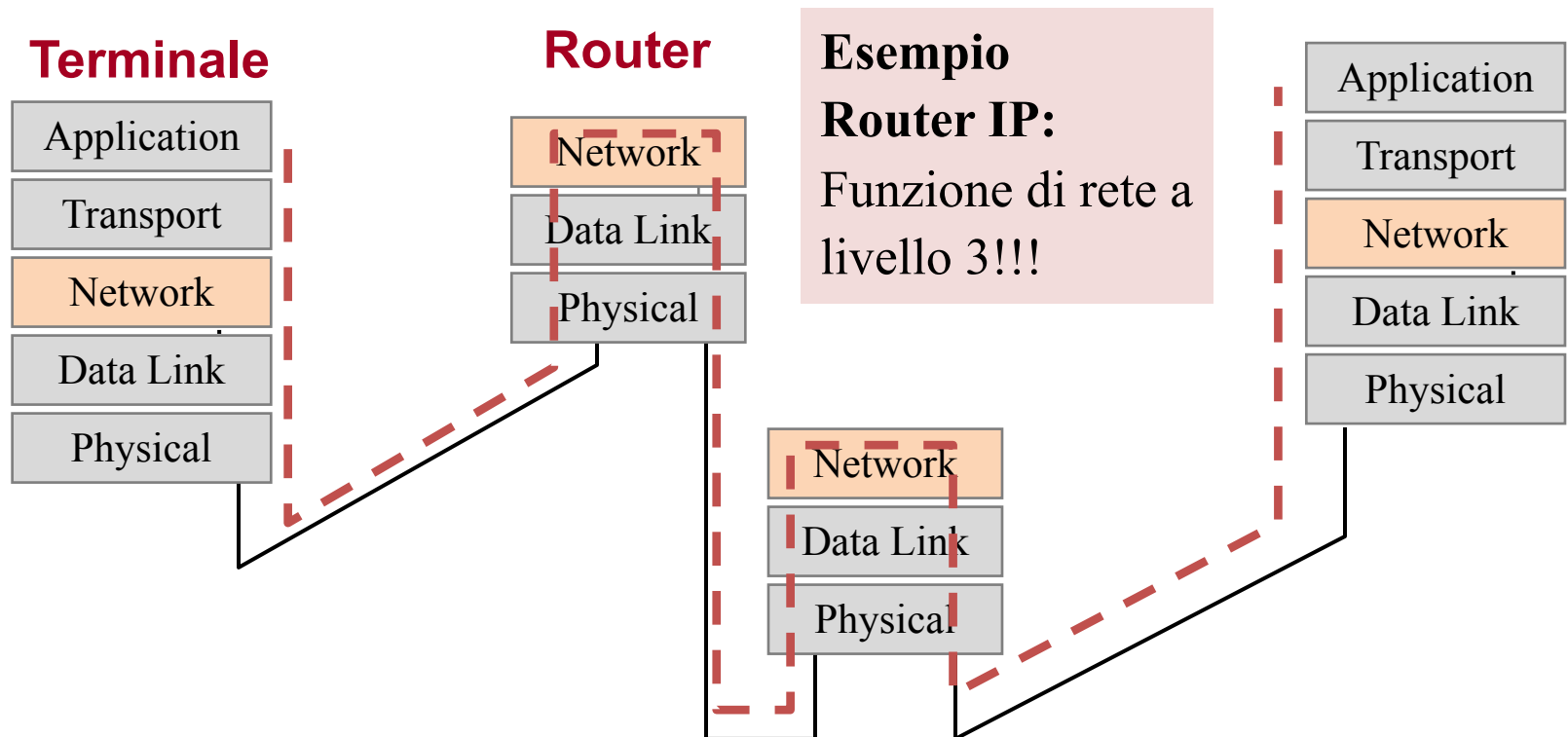
L'entità instradante decide dove inoltrare (forwarding) il pacchetto sulla base di una tabella di instradamento che viene consultato usando l'indirizzo come parametro di ricerca

OSSERVAZIONE: Se il pacchetto arriva da una porta d'ingresso, non sale mai più in alto nei livelli salvo che non sia giunto alla destinazione finale



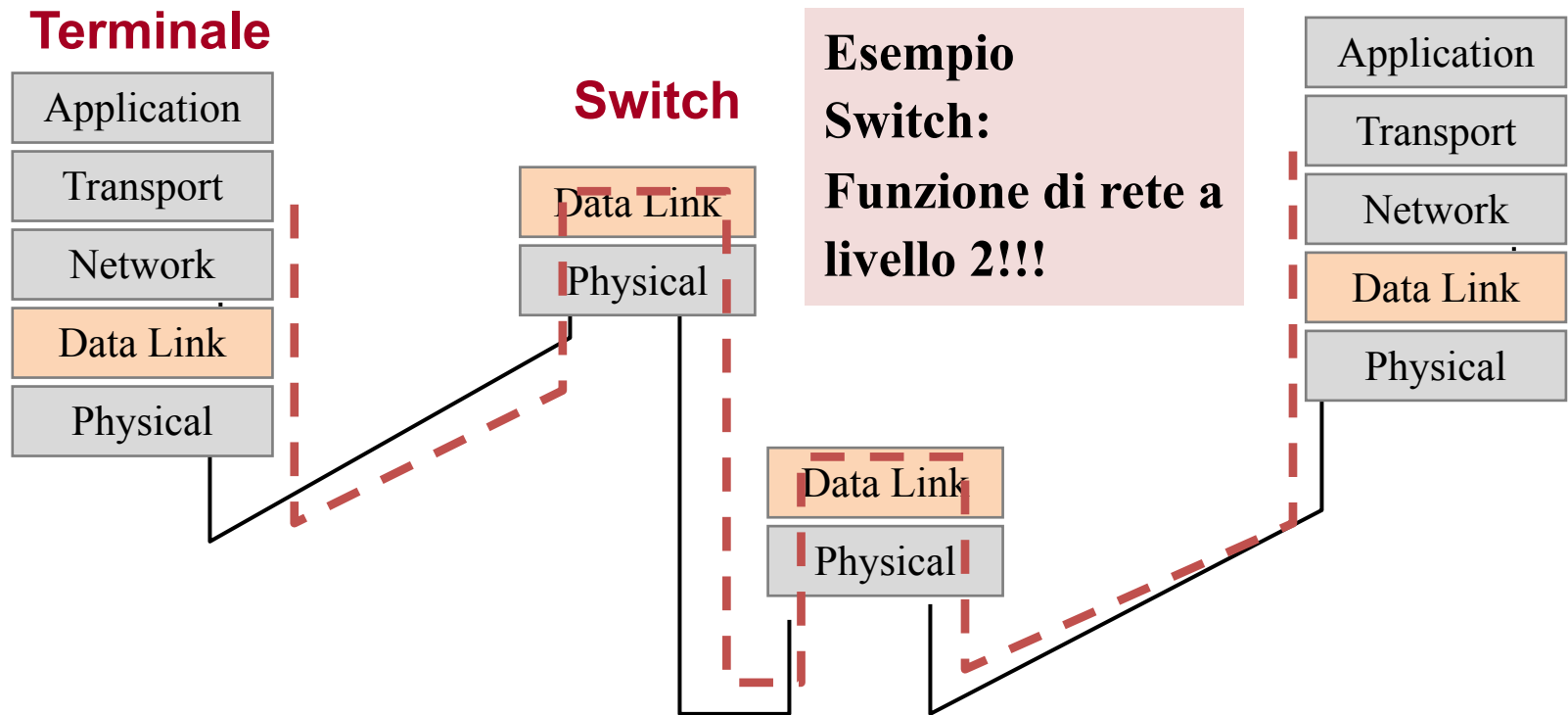
Esempio: Funzione di instradamento (routing)

- I nodi di rete non implementano i livelli superiori a quello di instradamento
- La funzione di instradamento può essere implementate in vari livelli in base alle circostanze (vedremo in dettaglio)
- Esempio: **Router IP**



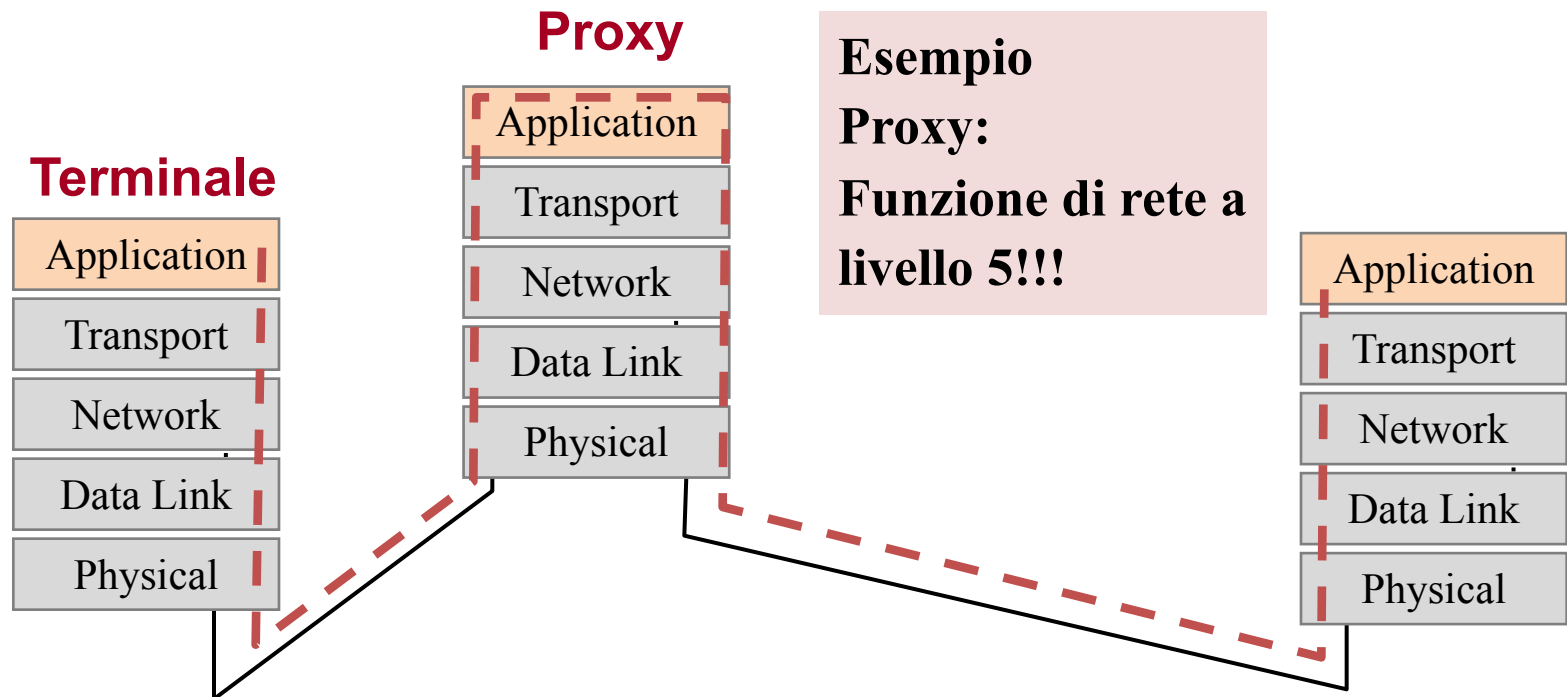
Esempio: Funzione di instradamento (routing)

- Esempio: **LAN Switch**



Esempio: Funzione di instradamento (routing)

- Esempio: **proxy**



Esempio: Funzione di instradamento (routing)

- Scelta dell'uscita avviene sulla base delle informazioni memorizzate in una tabella

tabella di routing	
indirizzo	uscita

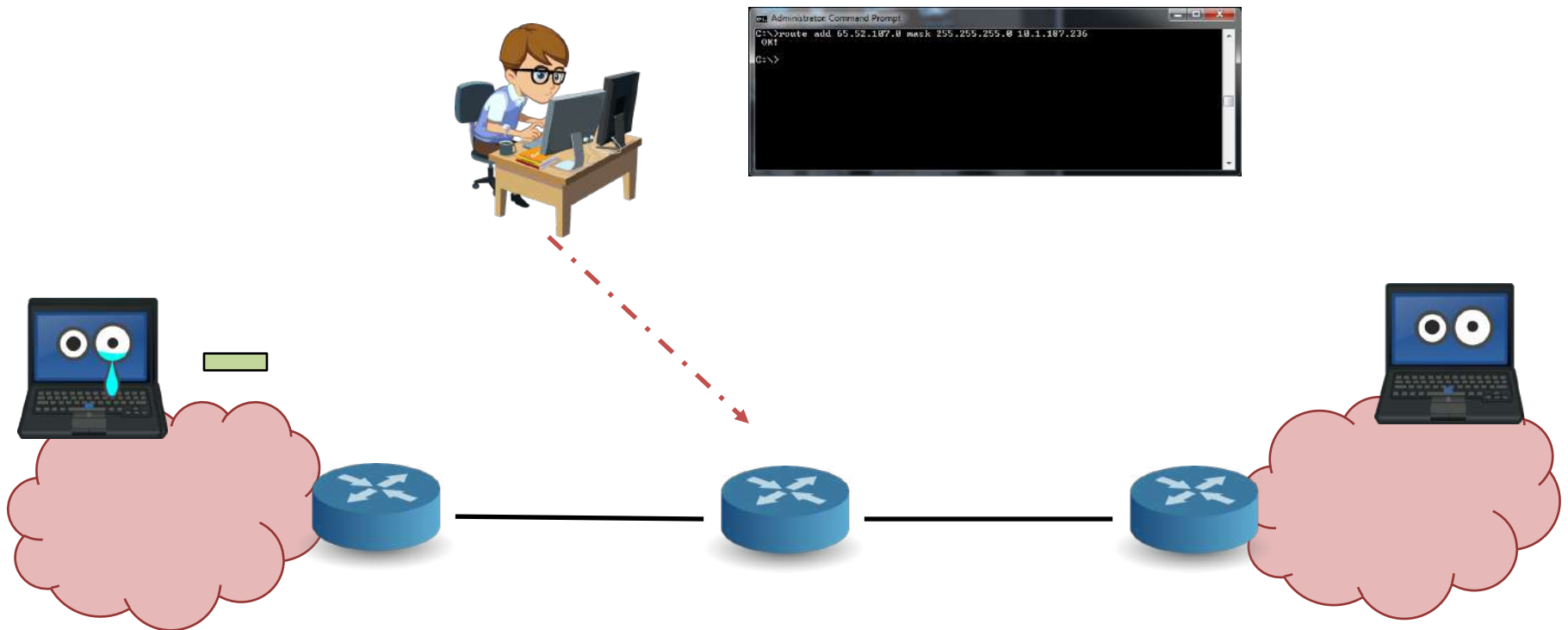
Ma chi scrive le tabelle di instradamento



Esempio: Funzione di instradamento (routing)

“Human Defined Networking”

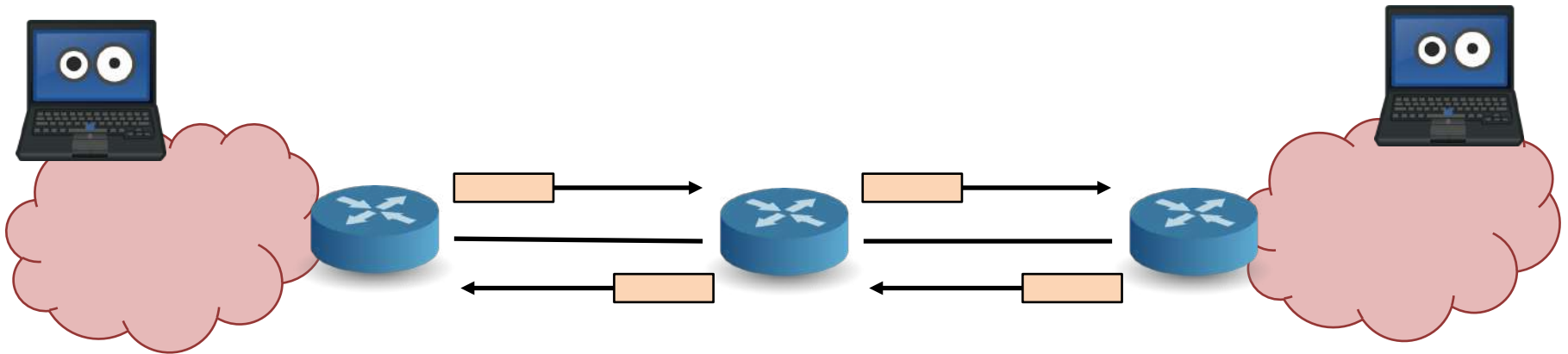
- Le tabelle di instradamento possono essere **scritte a mano** (faremo molti esercizi sulle tabelle di instradamento)
- Nei router IP le rotte configurate manualmente sono dette **rotte statiche**



Esempio: Funzione di instradamento (routing)

Protocolli di instradamento distribuiti

- L'approccio classico è però basato su uno **scambio di informazioni tra i router** che consente di compilare le tabelle di instradamento in modo **automatico e distribuito**
- Esempio:
 - MAC learning (si veda livello di linea)
 - Instradamento per cammini minimi (si veda livello IP)



Esempio: Funzione di instradamento (routing)

Software Defined Networking

- L'approccio emergente denominato **SDN** consente di operare come nell'approccio manuale, ma con un'**applicazione software centralizzata** che compila e modifica le tabelle di instradamento

