

UNIVERSITÀ DEGLI STUDI DI SALERNO

# Fondamenti di Informatica

---

Reti e Sistemi Operativi

Prof. Christian Esposito

Corso di Laurea in Ingegneria Meccanica e Gestionale (Classe I)

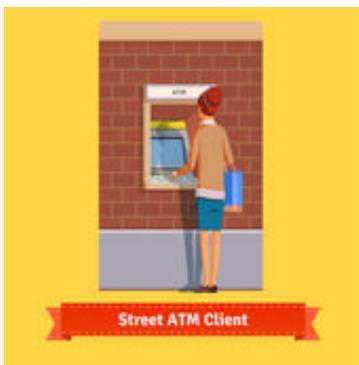
A.A. 2017/18

# Reti di calcolatori – 1/13

---

Una delle tecnologie dominanti ai nostri giorni e la capacità di scambiare informazioni, anche su vasta scala geografica.

Questo è stato fortemente consentita dalla possibilità di interconnettere tra loro i calcolatori formando una cosiddetta **rete**. Ciò ha posto le basi alla nascita dei sistemi distribuiti, ovvero sistemi hardware software che si presentano in maniera omogenea all'utente sebbene siano suddivisi in programmi in esecuzioni su computer geograficamente distribuiti.

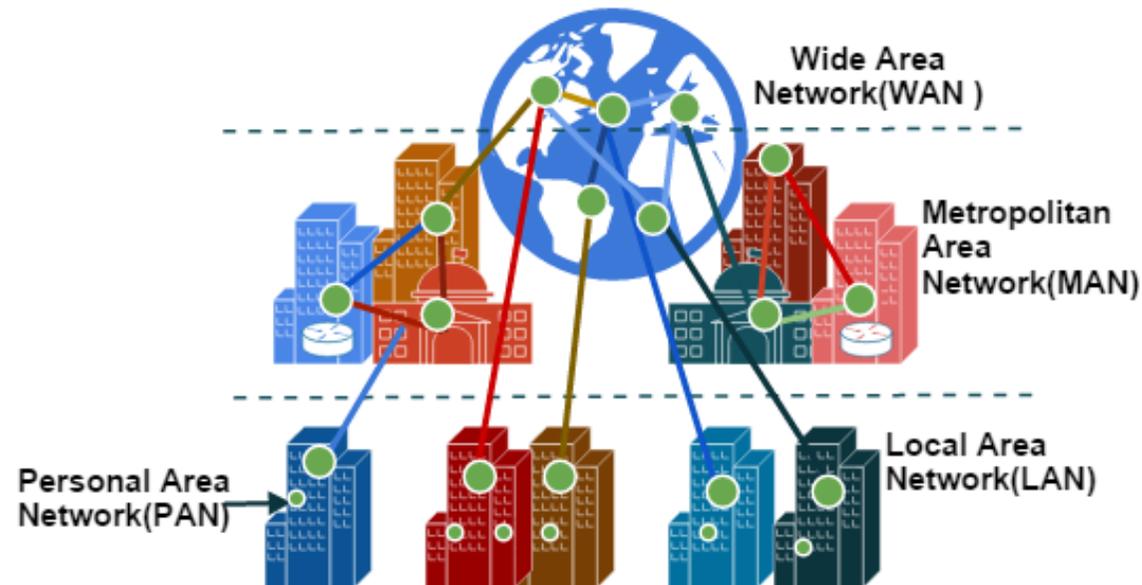


- Quando si usa il bancomat, l'utente interagisce con un terminale che acquisisce gli input dall'utente ed interroga un server remoto per l'esecuzione delle operazioni.

# Reti di calcolatori – 2/13

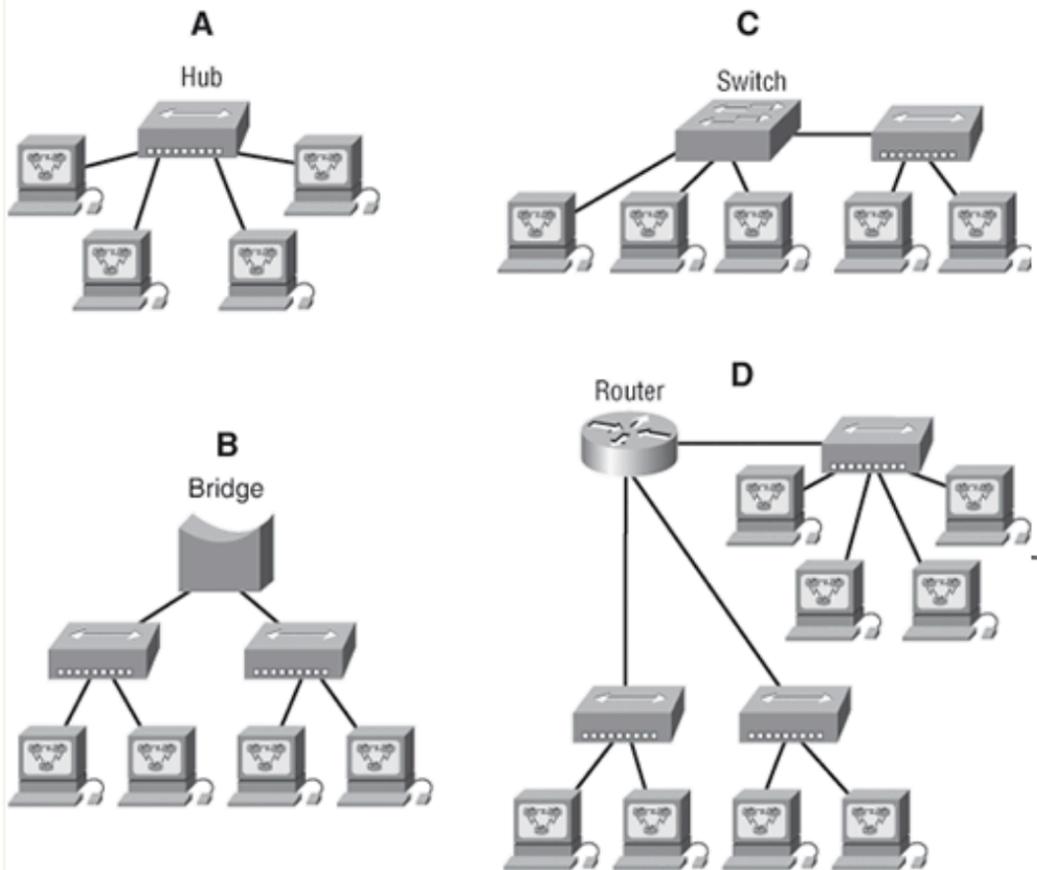
---

Le reti di calcolatori sono classificate in base alla loro estensione:



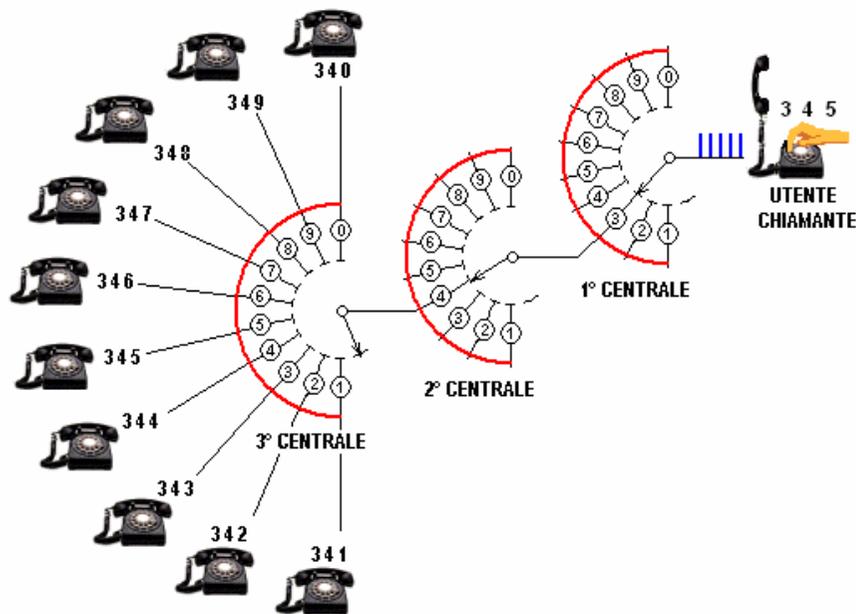
È impensabile pensare di effettuare collegamenti tra tutti i possibili computer di una rete, pertanto sussistono vari dispositivi di rete per un cablaggio (collegamento fisico dei computer) in una rete.

# Reti di calcolatori – 3/13



- A. **Hub** (livello 1) – un ripetitore che ritrasmette i pacchetti su tutte le sue porte di uscita;
- B. **Bridge** (livello 2) – Divide una rete in sotto-reti, usando indirizzi fisici associati ai computer per identificare a quale sottorete inviare un messaggio;
- C. **Switch** (livello 2) – Instradamento pacchetti all'interno delle reti LAN mediante indirizzo fisico della destinazione;
- D. **Router** (livello 3) – Istradamento tra sotto-reti direttamente o indirettamente raggiungibili mediante indirizzi IP.

# Reti di calcolatori – 4/13

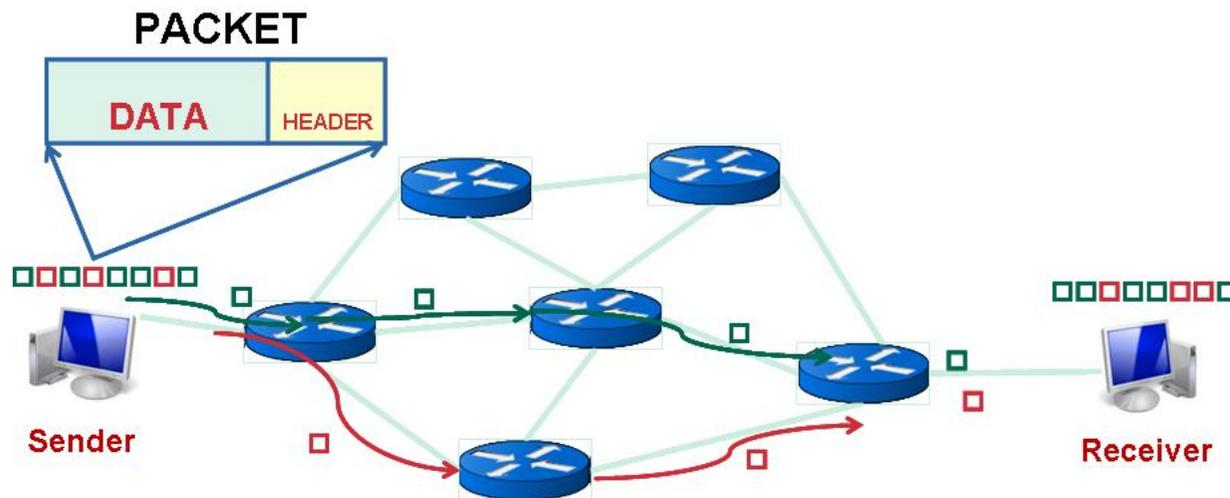


Le reti di calcolatori adottano la tecnica della commutazione di pacchetto: i dati da trasmettere sono suddivisi in una sequenza di byte con una determinata struttura, o pacchetti, trasmessi individualmente e in sequenza, senza stabilire a priori un percorso da seguire. Si differenzia dalla commutazione di circuito nella telefonia.

La commutazione di pacchetto si sviluppa a metà degli anni '60, per garantire le comunicazioni anche in caso di attacco nucleare, e sopperire alla scomparsa improvvisa di alcuni nodi o di intere porzioni della rete.

# Reti di calcolatori – 5/13

All'interno di ogni nodo appartenente alla rete, lo stato di funzionamento dei vari altri nodi ad esso direttamente collegati è aggiornato continuamente. Sulla base di queste informazioni, il nodo attualmente in possesso di un particolare pacchetto può determinare di volta in volta il migliore percorso da far seguire ai dati in modo che raggiungano il nodo destinatario nel minor tempo possibile.



Una porzione specifica del pacchetto, detto header, vi sono informazioni di controllo come l'indirizzo del destinatario.

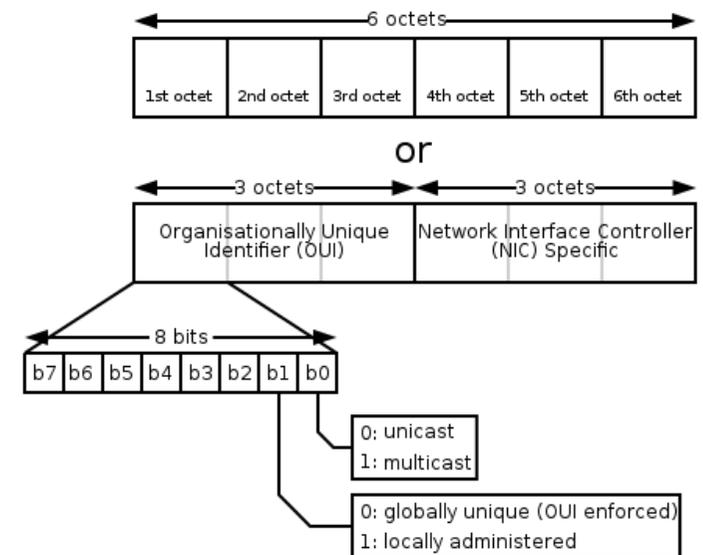
# Reti di calcolatori – 6/13



Ogni computer dispone di una scheda di rete (Network Interface Controller – NIC), ovvero una scheda elettronica alloggiata all'interno di un personal computer, server, stampante, router ecc., offrendo le funzioni necessarie per la comunicazione in rete.

Un'etichetta numerica identifica univocamente un dispositivo, detto host collegato a una rete informatica:

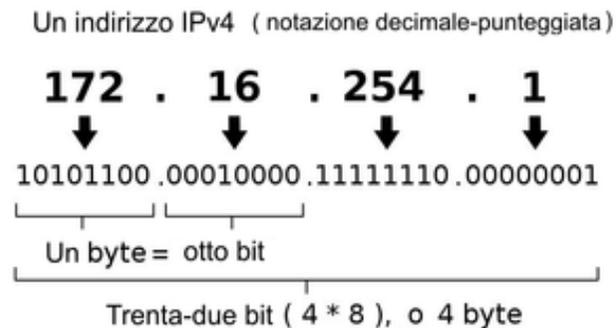
- Un indirizzo MAC è un codice di 48 bit assegnato univocamente dal produttore ad ogni scheda, modificabile a livello software.



# Reti di calcolatori – 7/13

---

- Un indirizzo IP è un'etichetta numerica a 32 bit che identifica univocamente un dispositivo, detto host, collegato a una rete informatica che utilizza l'Internet Protocol come protocollo di rete.



Per rendere gli indirizzi IP più facili da ricordare, si ricorre ad una rappresentazione mnemonica, associando un nome leggibile ad un determinato indirizzo.

L'associazione tra indirizzo mnemonico e quello IP viene risolto da un apposito servizio su rete chiamato Domain Name System – DNS.

# Reti di calcolatori – 8/13

<https://www.dnswatch.info>

**DNSWatch**

Hostname or IP  Type

[DNSWatch](#) > DNS Lookup for [www.unisa.it](#)

Searching for [www.unisa.it](#). A record at K.ROOT-SERVERS.NET. [193.0.14.129] ...took **1 ms**

Searching for [www.unisa.it](#). A record at dns.nic.it. [192.12.192.5] ...took **27 ms**

Searching for [www.unisa.it](#). A record at dns-001.unisa.it. [193.205.160.139] ...took **33 ms**

CNAME record found: [www3.unisa.it](#).

A record found: 193.205.160.20

Domain	Type	TTL	Answer
<a href="#">www.unisa.it</a> .	CNAME	86400	<a href="#">www3.unisa.it</a> .

[Monitor performance and availability of your DNS Server \(e.g. \[dns-001.unisa.it\]\(#\)\) - starting at \\$1/month](#)

Total elapsed query time: **61 ms**

Since these results are *absolutely up-to-date* they may differ from the results of your local nameserver.

It can take up to the specified "time to live" (TTL) for your nameserver to update its cache.

DNS queries have been sent from Frankfurt am Main, Germany

# Reti di calcolatori – 8/13

https://www.dnswatch.info

**DNSWatch**

Hostname or IP:  Type:

**DNSWatch**

Hostname or IP:  Type:

[DNSWatch](#) > DNS Lookup for www3.unisa.it

[DNSWatch](#) > DNS Lookup for www.unisa.it

Searching for www.unisa.it. A record at K.ROOT-SERVERS.NET. [193.0.14.129] ...took **1 ms**  
Searching for www.unisa.it. A record at dns.nic.it. [192.12.192.5] ...took **27 ms**  
Searching for www.unisa.it. A record at dns-001.unisa.it. [193.205.160.139] ...took **33 ms**

CNAME record found: www3.unisa.it.  
A record found: 193.205.160.20

Domain	Type	TTL	Answer
www.unisa.it.	CNAME	86400	www3.unisa.it.

Searching for www3.unisa.it. A record at K.ROOT-SERVERS.NET. [193.0.14.129] ...took **1 ms**  
Searching for www3.unisa.it. A record at dns.nic.it. [192.12.192.5] ...took **25 ms**  
Searching for www3.unisa.it. A record at dns-001.unisa.it. [193.205.160.139] ...took **34 ms**

A record found: 193.205.160.20

Domain	Type	TTL	Answer
www3.unisa.it.	A	86400	193.205.160.20

[Monitor performance and availability of your DNS Server \(e.g. dns-001.unisa.it\) - starting at \\$1/month](#)

Total elapsed query time: **60 ms**

[Monitor performance and availability of your DNS Server \(e.g. dns-001.unisa.it\) - starting at \\$1/month](#)

Since these results are *absolutely up-to-date* they may differ from the results of your local nameserver. It can take up to the specified "time to live" (TTL) for your nameserver to update its cache.

Total elapsed query time: **61 ms**

DNS queries have been sent from Frankfurt am Main, Germany

Since these results are *absolutely up-to-date* they may differ from the results of your local nameserver. It can take up to the specified "time to live" (TTL) for your nameserver to update its cache.

DNS queries have been sent from Frankfurt am Main, Germany

# Reti di calcolatori – 9/13

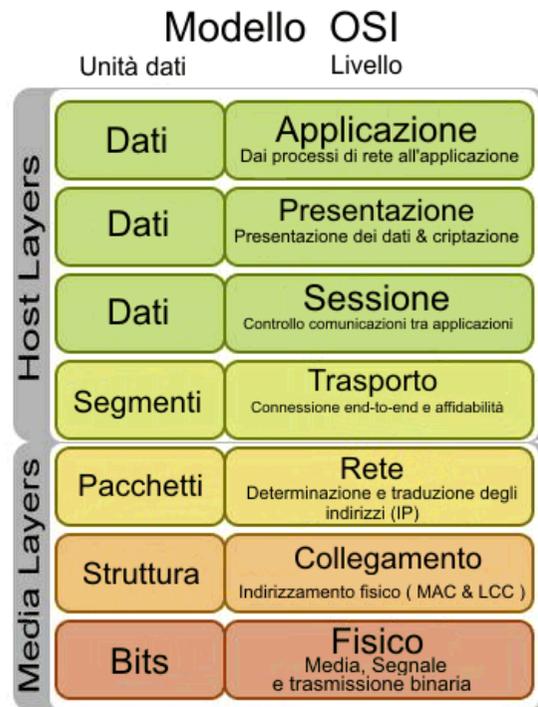
---

Per visualizzare gli indirizzi associati alle schede del proprio PC i sistemi operativi forniscono in genere un comando specifico da digitare su shell testuale: per Windows è `ipconfig /all`, mentre per Linux o MAC OS è `ifconfig -a`:

```
MacBook-Pro-di-Christian:~ christian_esposito$ ifconfig -a
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
    options=1203<RXCSUM,TXCSUM,TXSTATUS,SW_TIMESTAMP>
    inet 127.0.0.1 netmask 0xff000000
    inet6 ::1 prefixlen 128
    inet6 fe80::1%lo0 prefixlen 64 scopeid 0x1
    nd6 options=201<PERFORMNUD,DAD>
gif0: flags=8010<POINTOPOINT,MULTICAST> mtu 1280
stf0: flags=0<> mtu 1280
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
    ether f4:5c:89:a4:9c:ad
    inet6 fe80::84f:9b5e:cdb6:ec5f%en0 prefixlen 64 secured scopeid 0x4
    inet 192.168.1.101 netmask 0xfffff00 broadcast 192.168.1.255
    nd6 options=201<PERFORMNUD,DAD>
    media: autoselect
    status: active
en1: flags=963<UP,BROADCAST,SMART,RUNNING,PROMISC,SIMPLEX> mtu 1500
    options=60<TS04,TS06>
    ether 6a:00:01:b2:1f:b0
    media: autoselect <full-duplex>
    status: inactive
```

# Reti di calcolatori – 10/13

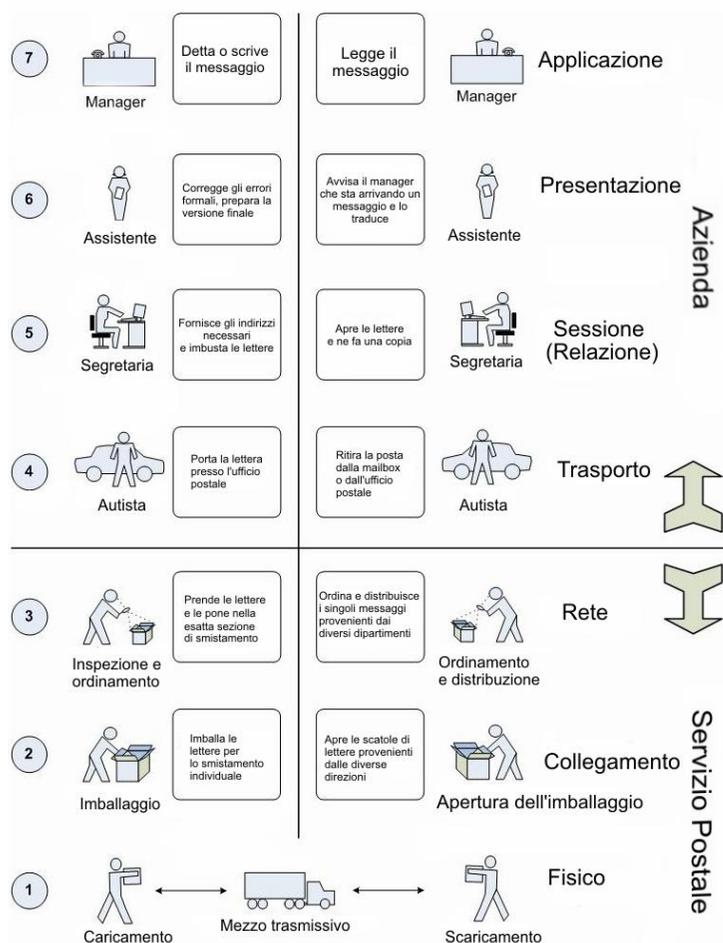
Una serie di protocolli sono necessari per implementare tutte le funzionalità necessarie per la gestione delle connessioni su rete, scambiare dati e implementare la commutazione di pacchetto.



L'Open System Interconnection (OSI) è uno standard per reti di calcolatori promosso dall'International Organization for Standardization (ISO) che definisce tali protocolli e come si relazioni uno con l'altro per supportare la comunicazione su rete.

Il modello ISO/OSI è una pila di protocolli composta da 7 livelli, ognuno regola una precisa parte del processo di comunicazione ed è legato a quello che lo segue e a quello che lo precede.

# Reti di calcolatori – 11/13



Parallelo tra una lettera e modello ISO - OSI

**Livello 1 – Fisico** -- Protocolli che regolano la trasmissione dei dati a livello fisico.

**Livello 2 – Collegamento** -- Protocolli per formare i pacchetti di dati da far viaggiare (framing).

**Livello 3 – Rete** -- Protocolli per l'attività di routing (instradamento).

**Livello 4 – Trasporto** – Protocolli per il trasporto fisico dei dati, come la connessione tra i due host.

**Livello 5 – Sessione** – Protocolli per instaurare, mantenere e abbattere connessioni tra applicazioni cooperanti (sessioni).

**Livello 6 – Presentazione** -- Protocolli per trasformare i dati delle applicazioni in un formato standard e offrire servizi di comunicazione comuni, come la crittografia e la formattazione.

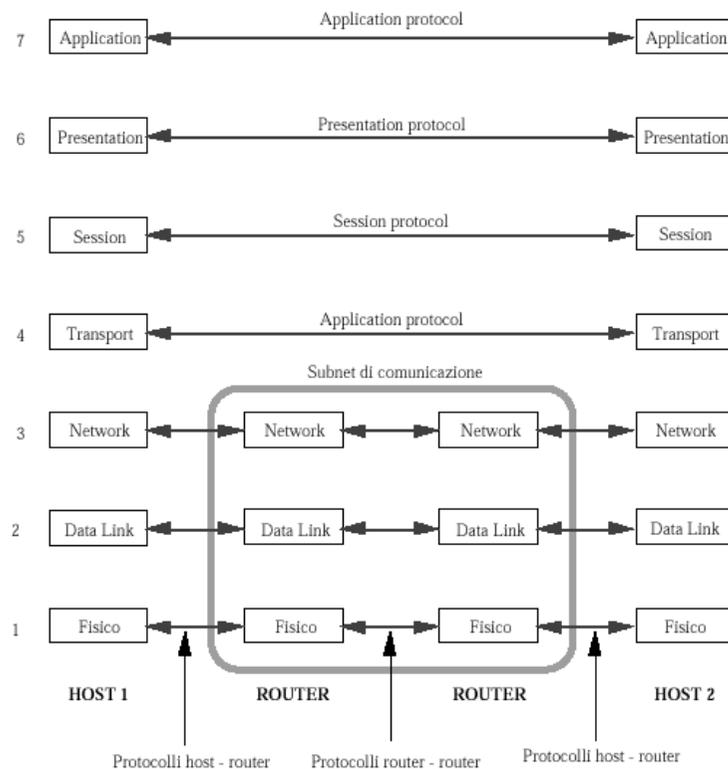
**Livello 7 – Applicazione** -- I protocolli interagiscono direttamente con i programmi con moduli di comunicazione di rete (come ad esempio i client di posta elettronica).

# Reti di calcolatori – 12/13

---

- Internet Protocol (IP) è un protocollo classificato al livello di rete (3) del modello ISO/OSI, nato per interconnettere reti eterogenee realizzando l'indirizzamento e l'instradamento (commutazione), senza connessione e di tipo best effort.
- Transmission Control Protocol (TCP) è un protocollo di livello di trasporto che si occupa di rendere affidabile la comunicazione dati in rete e su di esso si appoggiano gran parte delle applicazioni su rete. Prima ancora del trasferimento dei dati, TCP si occupa di riservare risorse tra processi applicativi che si scambiano informazioni o servizi tra loro (es. server e client).
- User Datagram Protocol (UDP) è un protocollo di livello di trasporto che non gestisce il riordinamento né la ritrasmissione dei pacchetti, ed è perciò non affidabile. In compenso è molto rapido ed efficiente per le applicazioni time-sensitive come trasmissioni di informazioni audio-video in real-time.
- TCP e UDP vengono usati in combinazione con il protocollo di livello di rete IP, e tali combinazioni rendono il nome di TCP/IP o UDP/IP

# Reti di calcolatori – 13/13



I dispositivi di rete, come i router, non dispongono di tutti i livelli ISO/OSI, ma solo i tre livelli inferiori. Gli altri comprendono tutto ciò che serve a connettere tra loro i due utenti finali. I tre livelli inferiori hanno funzioni di tipo box-to-box, mentre quelli di livello superiore hanno funzioni di tipo end-to-end.

# Introduzione al Web – 1/2

---

Il World Wide Web (letteralmente "rete di grandezza mondiale"), abbreviato Web, nel corso degli anni è emerso come una tecnologia per condividere informazioni su Internet. Recentemente, (con l'avvento dei servizi web) rappresenta anche un mezzo per integrare applicazioni su Internet, ovvero consentire ad applicazioni diverse tra loro di poter interagire e condividere informazioni.

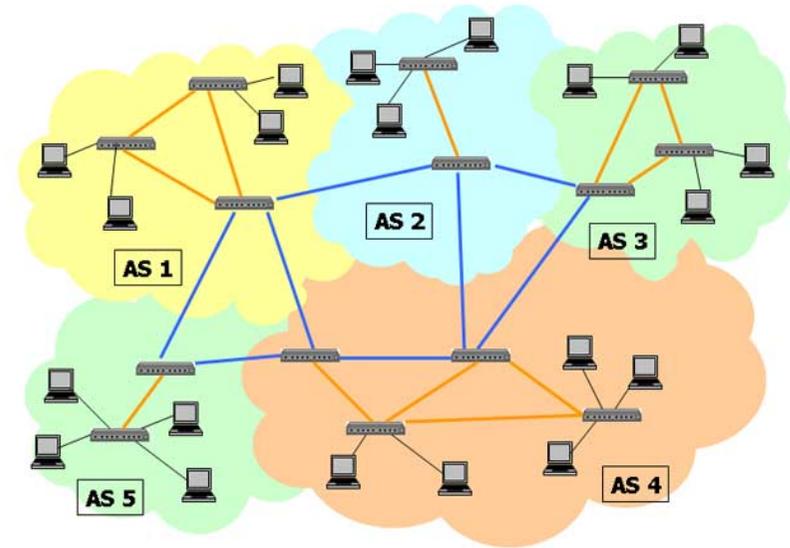
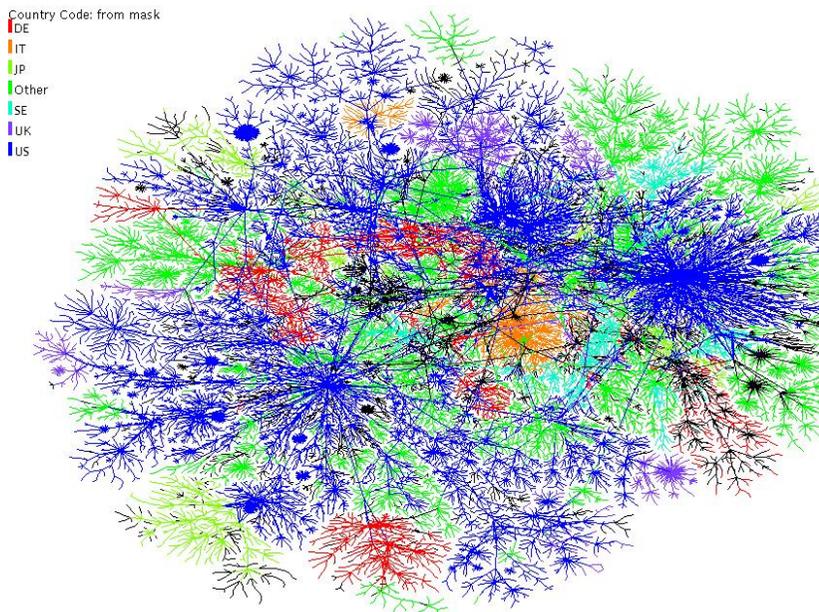
Permette di navigare e usufruire di un insieme vastissimo di contenuti amatoriali (multimediali e non) collegati tra loro attraverso legami (link), e di ulteriori servizi accessibili a tutti o ad una parte selezionata degli utenti di Internet. Questa facile reperibilità di informazioni è resa possibile oltre che dai protocolli di rete anche dalla presenza, diffusione, facilità d'uso ed efficienza dei motori di ricerca e dei web browser in un modello di architettura di rete definito client/server.





# Scambio Informazioni su Internet – 1/10

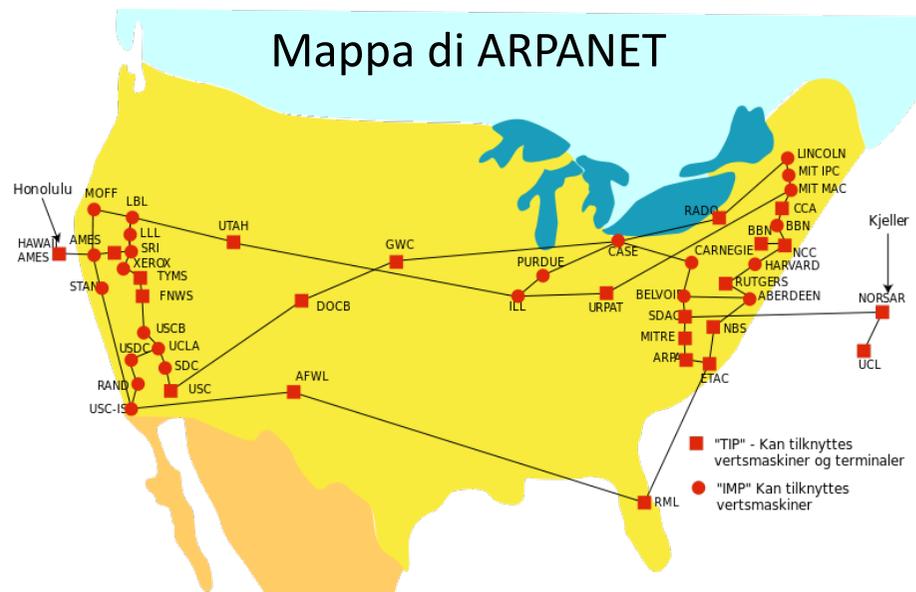
Internet è un sistema globale di reti di calcolatori, composta interconnettendo singole reti con un approccio gerarchico.



Attualmente la topologia di Internet risulta molto complessa, ma generalmente si compone dell'interconnessione di sistemi autonomi (Autonomous Systems, AS), ovvero un gruppo di router e reti sotto il controllo di una singola e ben definita autorità amministrativa.

# Scambio Informazioni su Internet – 2/10

Gli albori di Internet corrispondono con il progetto ARPANet, una rete di computer realizzata nel 1969 dal DARPA, l'agenzia del Dipartimento della Difesa degli Stati Uniti responsabile per lo sviluppo di nuove tecnologie ad uso militare. Per tutti gli anni Settanta ARPANet continuò a svilupparsi in ambito universitario e governativo, ma dal 1974, con l'avvento di TCP/IP, il progetto della rete prese ad essere denominato Internet.

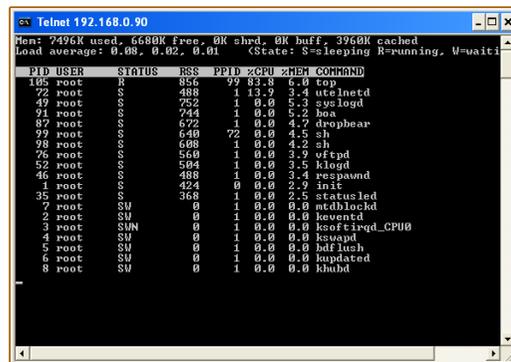


Primo router di ARPANET

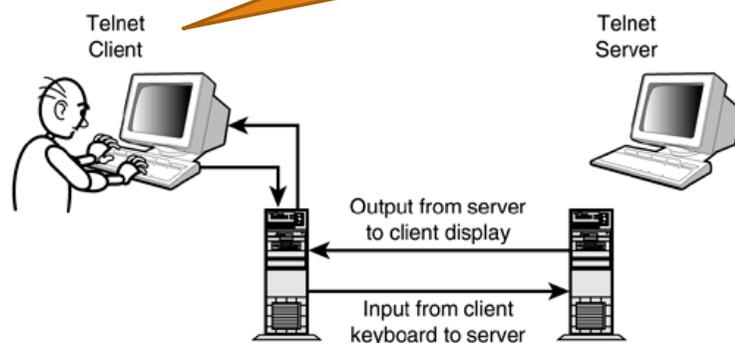
# Scambio Informazioni su Internet – 3/10

I due iniziali standard per lo scambio di informazioni erano:

- Il protocollo Telnet è solitamente utilizzato per fornire all'utente sessioni di login remoto di tipo riga di comando tra host su Internet.



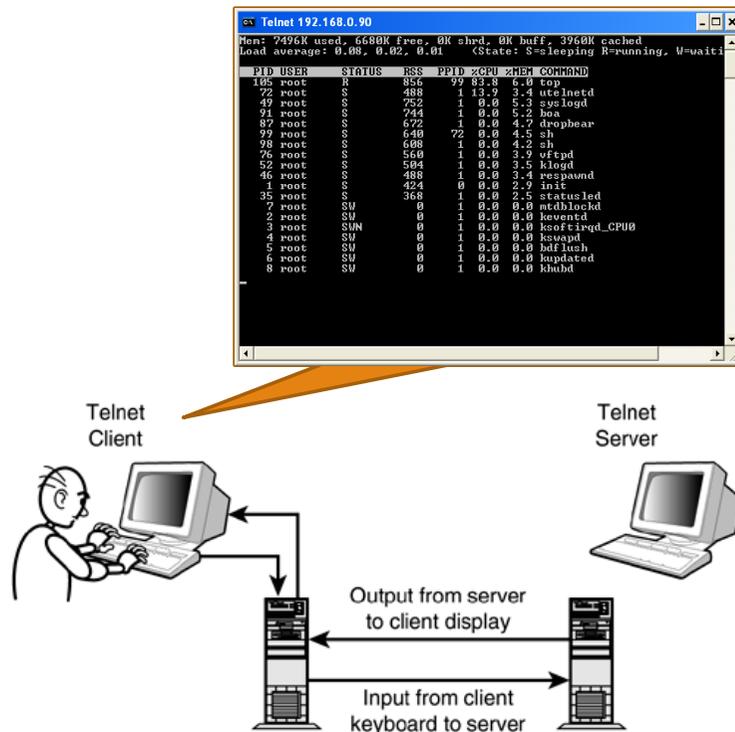
```
Telnet 192.168.0.90
Man: 2496K used, 6680K free, 0K chkd, 0K buff, 3960K cached
Load average: 0.08, 0.02, 0.01  <State: S=sleeping R=running, W=waiti
PID USER      STATUS  RSS    PPID %CPU %MEM  COMMAND
165 root      R       856    99  83.8  6.0  top
72  root      S       488    1  13.9  3.4  utelnsd
49  root      S       752    1  0.0  5.3  syslogd
91  root      S       744    1  0.0  5.2  boad
87  root      S       672    1  0.0  4.7  dropbear
99  root      S       640    72  0.0  4.5  sh
98  root      S       608    1  0.0  4.2  sh
76  root      S       560    1  0.0  3.9  vftpd
52  root      S       504    1  0.0  3.5  klogd
46  root      S       488    1  0.0  3.4  respawnd
1  root      S       424    0  0.0  2.9  init
35  root      S       368    1  0.0  2.5  statusled
7  root      SV      0       1  0.0  0.0  mtdblockd
2  root      SV      0       1  0.0  0.0  kowand
3  root      SV      0       1  0.0  0.0  ksoftirqd_CPU0
4  root      SV      0       1  0.0  0.0  kswapd
5  root      SV      0       1  0.0  0.0  hrtlsh
6  root      SV      0       1  0.0  0.0  kupdated
8  root      SV      0       1  0.0  0.0  khubd
```



# Scambio Informazioni su Internet – 3/10

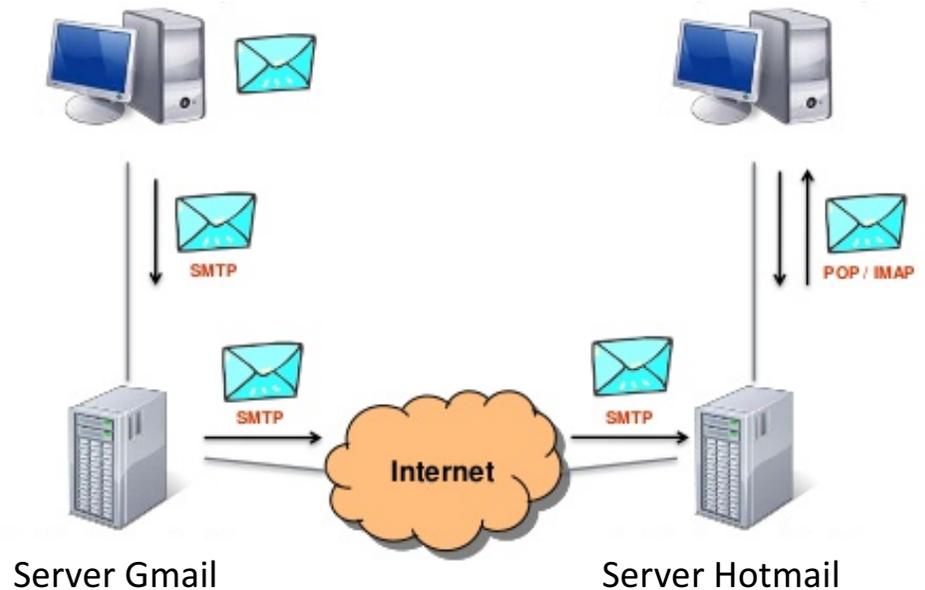
I due iniziali standard per lo scambio di informazioni erano:

- Simple Mail Transfer Protocol (SMTP) è il protocollo per la trasmissione di email, mentre per la ricezione abbiamo POP e IMAP.

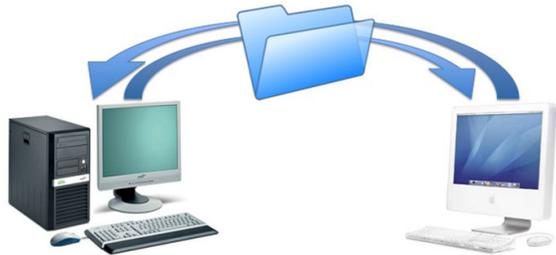


Mittente E-mail

Destinatario E-mail

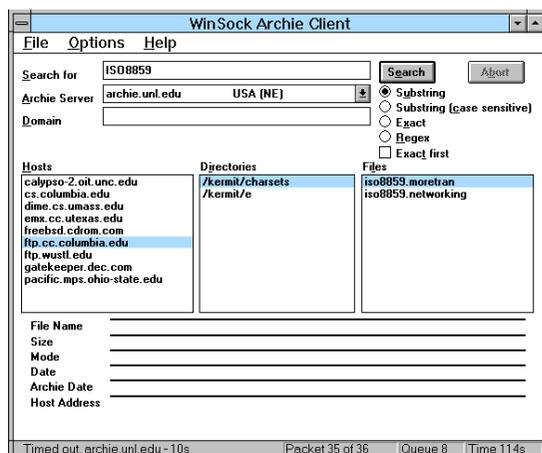


# Scambio Informazioni su Internet – 4/10



Dopo è stato pubblicato File Transfer Protocol (FTP), per trasferimenti di file tra nodi su Internet, e ha consentito lo sviluppo dei primi sistemi informativi distribuiti simile al Web, come

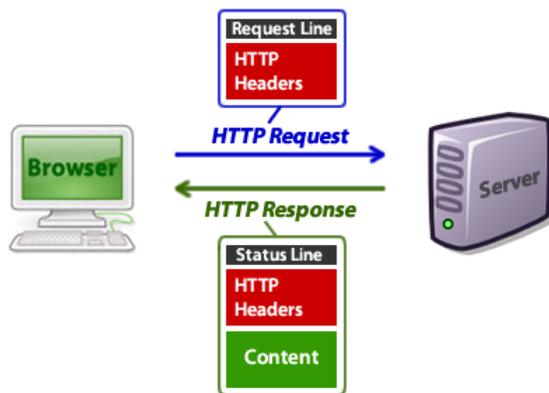
- Archie, un file system distribuito per ricercare file su server FTP,
- Gopher, per la pubblicazione e l'accesso di file di testo su Internet.



# Scambio Informazioni su Internet – 5/10

Le tecnologie alla base del mondo del Web sono un'evoluzione di questi iniziali sistemi e protocolli:

- HyperText Transfer Protocol (HTTP) è un protocollo generico e senza stato per trasferire file sulla rete. Generico nel senso che consente l'accesso verso altri protocolli come FTP e SMTP.



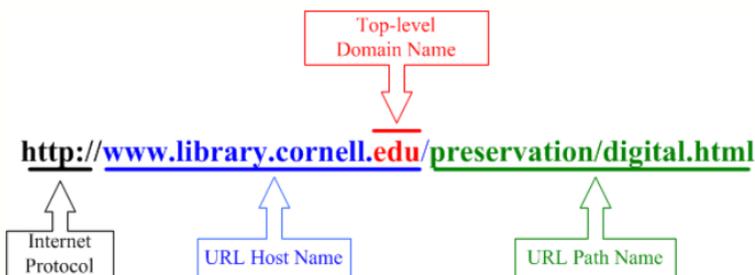
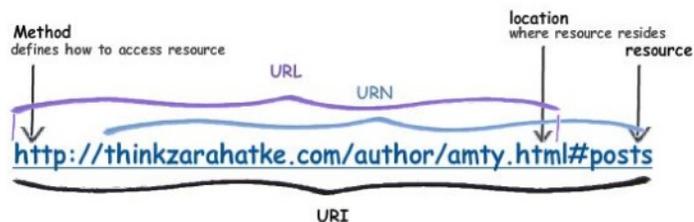
HTTP originariamente aveva il compito di scambiare ipertesti espressi secondo lo standard HyperText Markup Language (HTML), dove i client effettuavano una precisa richiesta e seguiva la consegna dell'ipertesto che soddisfaceva la richiesta.

- HTML è il linguaggio di markup solitamente usato per la formattazione e impaginazione di documenti ipertestuali disponibili nel World Wide Web sotto forma di pagine web.

# Scambio Informazioni su Internet – 6/10

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Example</title>
5     <link rel="stylesheet" href="sty
6   </head>
7   <body>
8     <h1>
9     <a href="/">Header</a>
10    </h1>
11    <nav>
12      <a href="one/">One</a>
13      <a href="two/">Two</a>
14      <a href="three/">Three</a>
15    </nav>
```

L'HTML non è un linguaggio di programmazione, di gestire i contenuti associandone allo stesso tempo la struttura grafica (layout) grazie all'utilizzo di tag diversi, ognuno specifica un diverso ruolo dei contenuti che esso contrassegna.



- HTTP effettua lo scambio di documenti HTML identificati univocamente usando un Uniform Resource Identifiers (URI): identificati univocamente con Uniform Resource Name (URN) e sono indirizzabili per mezzo di Uniform Resource Locators (URL), che specifica la locazione nella rete del documento, e descrive come accedervi.

# Scambio Informazioni su Internet – 7/10

---

- HTTP è basato su un modello di interazione di tipo client/server:

# Scambio Informazioni su Internet – 7/10

---

- HTTP è basato su un modello di interazione di tipo client/server:
  - **è presente un client HTTP, come un browser;**

HTTP Client



# Scambio Informazioni su Internet – 7/10

---

- HTTP è basato su un modello di interazione di tipo client/server:
  - **è presente un client HTTP, come un browser, e un server HTTP che ospita un insieme di documenti ipertestuali in HTML.**

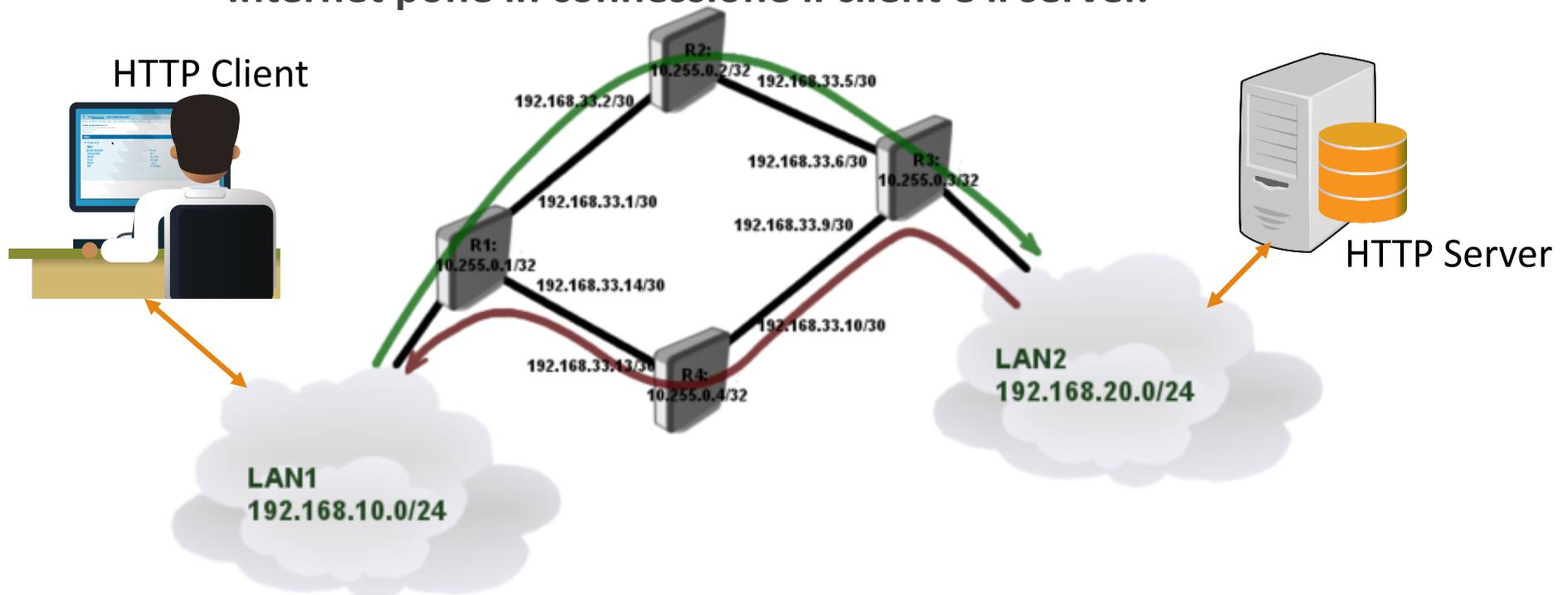
HTTP Client



HTTP Server

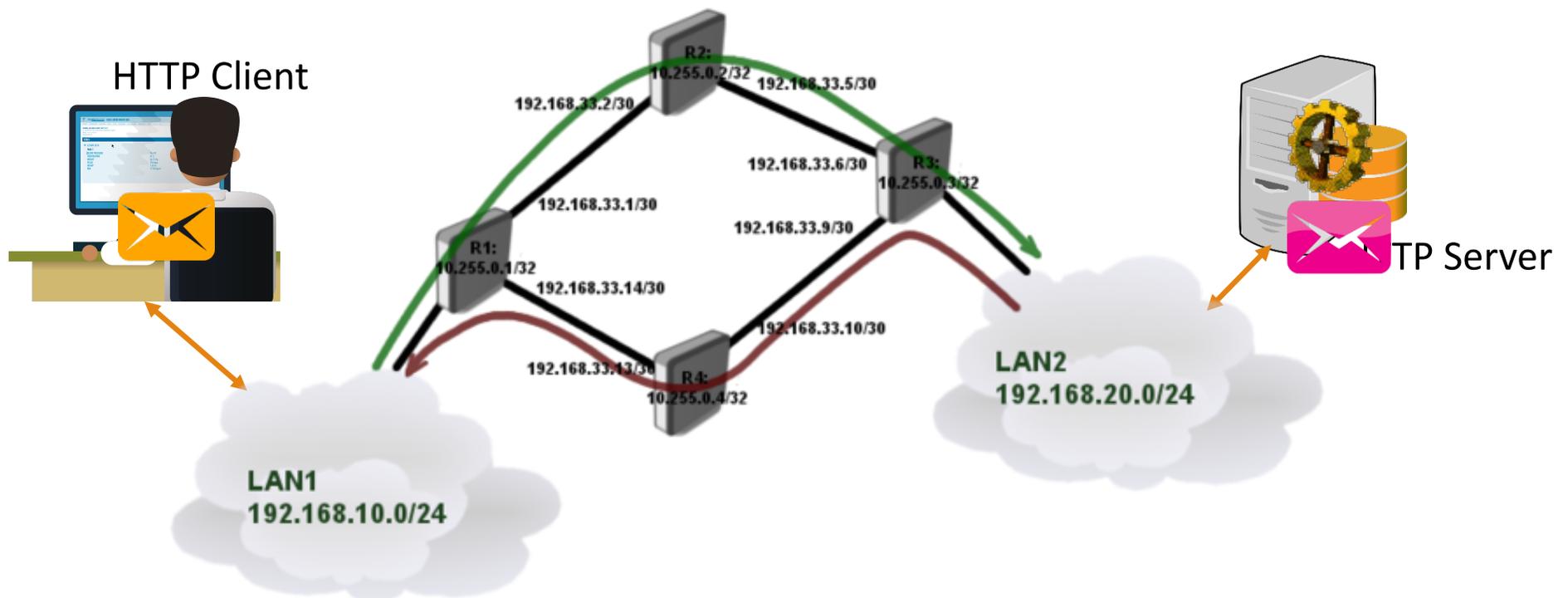
# Scambio Informazioni su Internet – 7/10

- HTTP è basato su un modello di interazione di tipo client/server:
  - è presente un client HTTP, come un browser, e un server HTTP che ospita un insieme di documenti ipertestuali in HTML.
  - Internet pone in connessione il client e il server.



# Scambio Informazioni su Internet – 7/10

- HTTP è basato su un modello di interazione di tipo client/server:
  - **Attraverso la suite TCP/IP, il client apre una connessione ed effettua una richiesta. Il server soddisfa la richiesta e ne restituisce l'esito.**

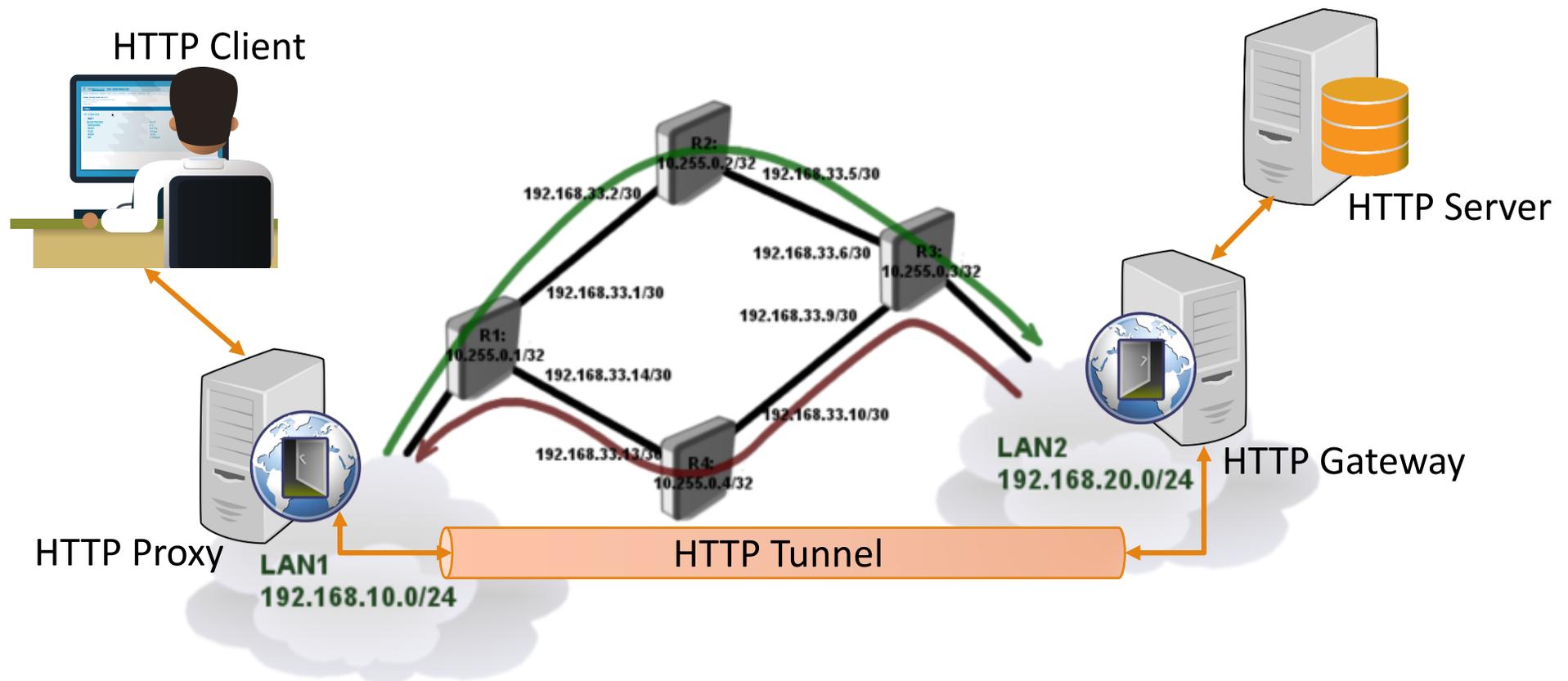


# Scambio Informazioni su Internet – 8/10

---

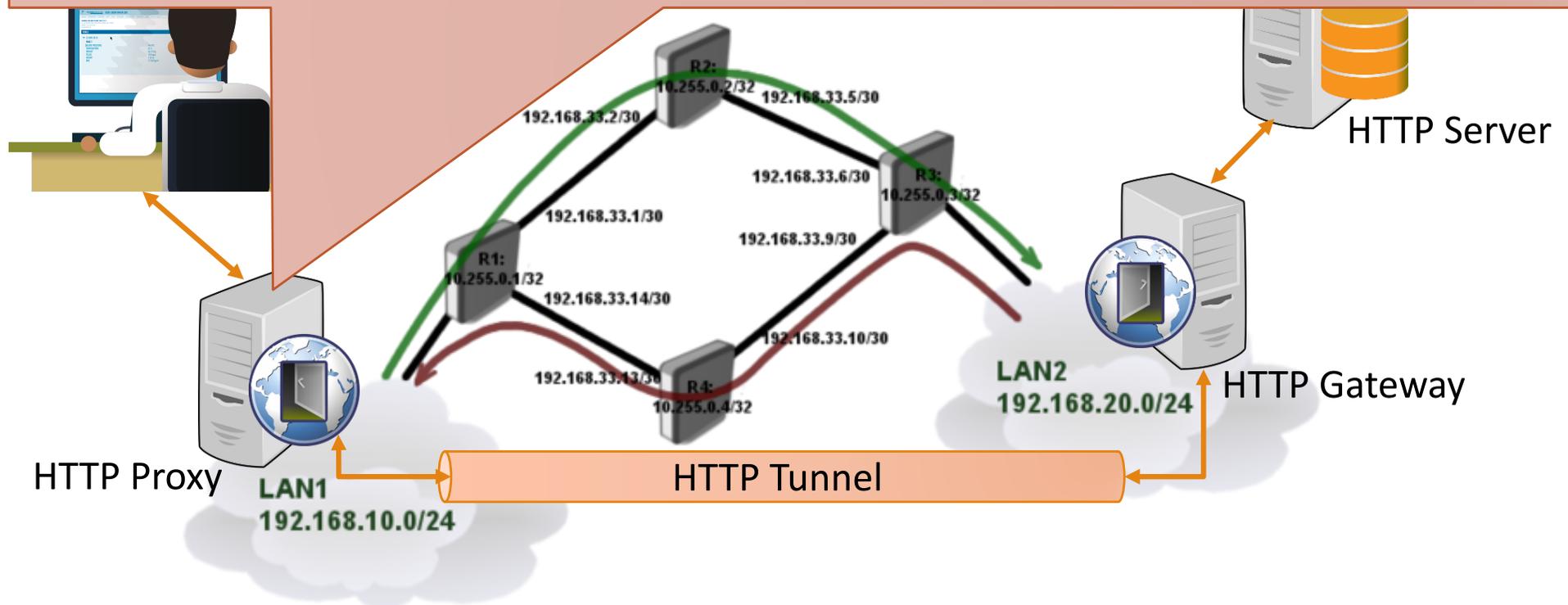
- Un client HTTP deve indicare nella richiesta l'operazione che deve svolgere il server:
  - OPTIONS, per sapere le opzioni supportate dal server;
  - GET, per recuperare un determinato documento identificato da un URL, oppure un documento ottenuto dall'esecuzione di un programma specificato nella richiesta;
  - POST, per aggiungere o collegare le informazioni presenti nella richiesta alla risorsa indicata dal client;
  - DELETE, per rimuovere dal server la risorsa indicata nella richiesta.
- Client e server HTTP possono non vedersi direttamente ma avere degli intermediari che si frappongono tra loro:
  - HTTP Proxy: contattato direttamente dal client per invio richieste
  - HTTP Gateway: ricevente delle richieste per conto del server;
  - HTTP Tunnel: consente di incapsulare diversi protocolli all'interno del protocollo HTTP, che funge da wrapper.

# Scambio Informazioni su Internet – 9/10



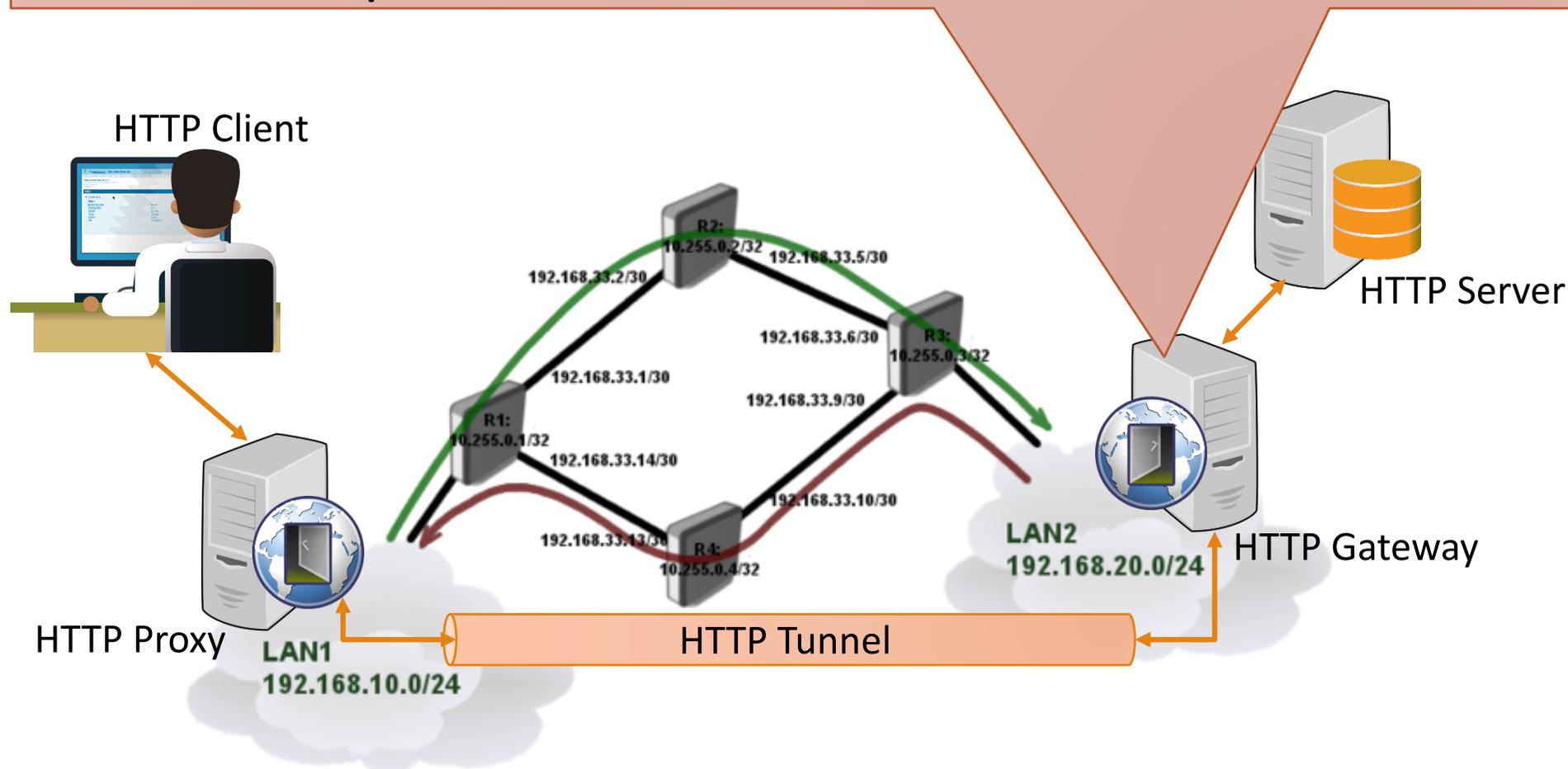
# Scambio Informazioni su

Per utilizzare un proxy è possibile configurare il client in modo che si colleghi al proxy invece che al server, oppure definire un proxy trasparente; in questo caso, alcune connessioni (ad esempio quelle HTTP) vengono automaticamente indirizzate al proxy. Un proxy può essere usato per permettere ad una rete privata di accedere all'esterno, per caching, monitoraggio e controllo, e anche per garantire un maggiore livello di privacy mascherando il vero indirizzo IP del client in modo che il server non venga a conoscenza di chi ha effettuato la richiesta.



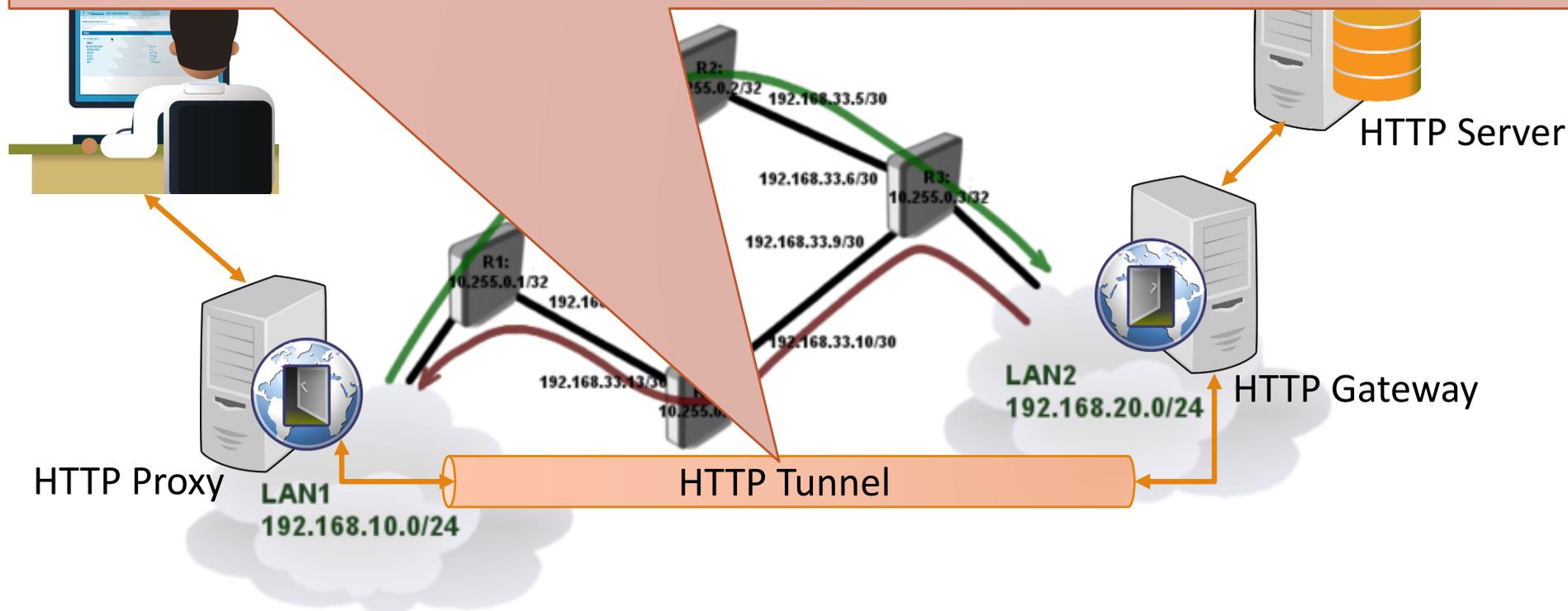
# Scambio Informazioni su

Un gateway HTTP è un server che funge da server originario ricevendo la richiesta per poi rispondere. Il client richiedente potrebbe non essere consapevole di stare in comunicazione con un gateway. Il gateway viene utilizzato per interconnettere reti locali diverse in modo da permettere lo scambio di informazioni tra di loro.



# Scambio Informazioni su

Per tunneling si intendono le tecniche che incapsulano un protocollo all'interno di un altro dello stesso livello o di livello superiore; in particolare l'HTTP-tunneling è una tecnica che consente di incapsulare diversi protocolli all'interno del protocollo HTTP, che funge da wrapper, consentendo di far uscire sulla porta assegnata all'HTTP tutto il traffico che altrimenti troverebbe lo sbarramento del firewall, creando una sorta di tunnel virtuale. Lo scopo di un tunnel HTTP è di superare, ad esempio, le restrizioni alla navigazione imposte su una rete o i controlli del firewall.



# Scambio Informazioni su Internet – 10/10

---

Sebbene i suoi vantaggi e la larga diffusione, HTTP è soggetto a un insieme di limitazioni, molte delle quali sono state affrontate per mezzo di una serie di estensioni dell'originario protocollo HTTP:

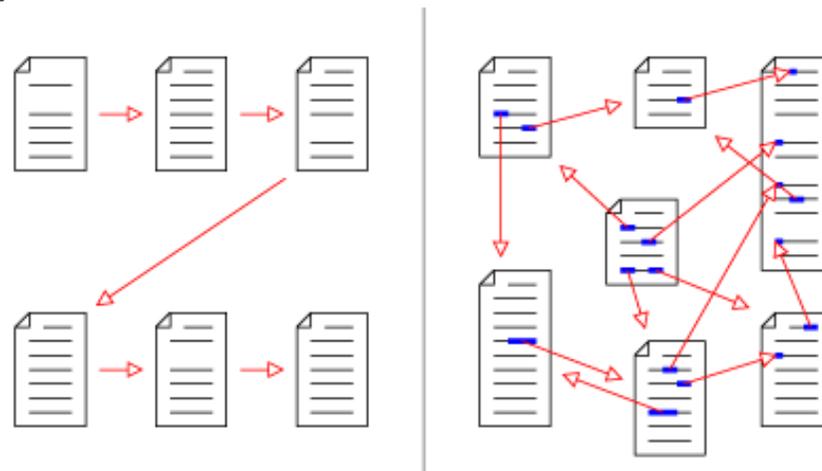
- Non sono presenti delle caratteristiche di sicurezza per proteggere le comunicazioni tra client e server da malintenzionati a compromettere i messaggi scambiati o a carpirne le informazioni contenute.
- Il protocollo è senza stato e non tiene traccia di dati per richieste diverse, e ogni richiesta è a se stante.
- Il protocollo è particolarmente verboso, e non risulta particolarmente efficiente considerando i tempi di consegna e la quantità di traffico imposto sulla rete durante una sessione.

# Iperpertexti e HTML – 1/7

---

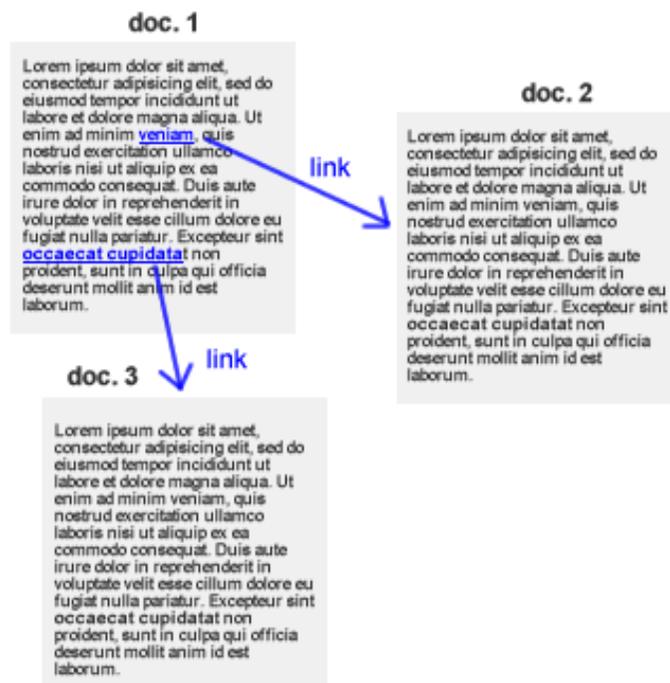
Un ipertesto è un insieme di documenti messi in relazione tra loro per mezzo di parole chiave. Può essere visto come una rete; i documenti ne costituiscono i nodi. La caratteristica principale di un ipertesto è che la lettura può svolgersi in maniera non lineare: qualsiasi documento della rete può essere "il successivo", in base alla parola chiave scelta.

La scelta di una parola chiave diversa porta all'apertura di un documento diverso: all'interno dell'ipertesto sono possibili praticamente infiniti percorsi di lettura.



# Ipertesti e HTML – 2/7

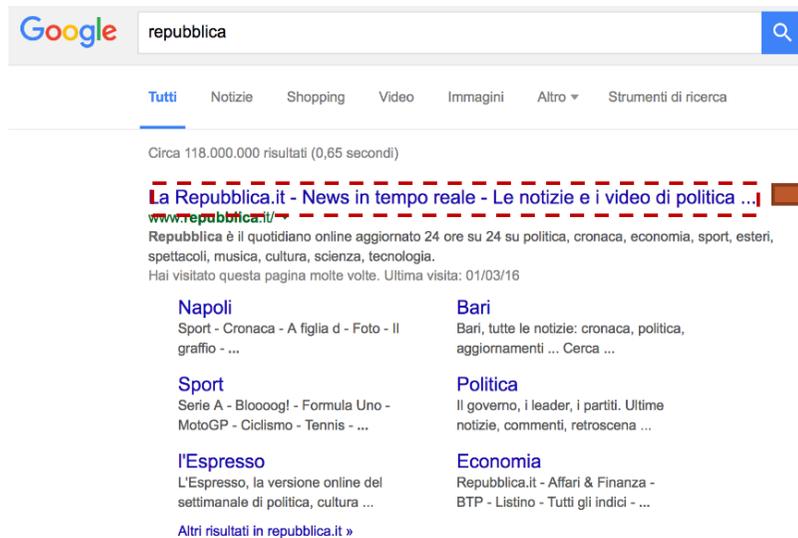
L'ipertesto informatico è la versione di ipertesto più usata e più diffusa oggi. L'ipertesto consiste in un collegamento alla parola chiave (opportunamente evidenziata allo scopo), che talvolta appare nello schermo anche sotto forma di icona o immagine.



Selezionando o posizionandosi su tale parola o oggetto e facendo clic con il mouse oppure dando l'invio (per navigazione basata sulla tastiera) si ottiene come conseguenza l'apertura di un altro documento, che si può trovare sullo stesso server o altrove. Quindi le parole chiave funzionano come collegamenti ipertestuali (hyperlink in inglese), che consentono all'utente di navigare verso informazioni aggiuntive.

# Ipertesti e HTML – 3/7

Dopo la nascita del World Wide Web (1993) l'ipertesto ha avuto un notevolissimo sviluppo. Tutto il web, è stato concepito dall'inglese Tim Berners-Lee, come un ipertesto globale in cui tutti i siti mondiali possono essere consultati da tutti. La pagina web è il singolo documento e la "navigazione" è il passaggio da un sito all'altro, o da una pagina all'altra, tramite i "link".



Google search results for "repubblica". The search bar shows "repubblica" and a magnifying glass icon. Below the search bar, there are tabs for "Tutti", "Notizie", "Shopping", "Video", "Immagini", "Altro", and "Strumenti di ricerca". The search results show "Circa 118.000.000 risultati (0,65 secondi)". The first result is "La Repubblica.it - News in tempo reale - Le notizie e i video di politica ..." with a red dashed box around the title and an orange arrow pointing to the right. Below the search results, there are several categories: Napoli, Sport, L'Espresso, Bari, Politica, and Economia, each with a brief description and a "Cerca ..." link.



la Repubblica.it homepage. The header includes "NETWORK", "L'Espresso", "R&S LE INCHIESTE", "18 maggio 2016 - Aggiornato alle 09.19", "LAVORO ANNUNCI ASTE", and "Accedi". The main logo is "la Repubblica.it" with the tagline "Il mondo in diretta" and "24 ore su 24". Below the logo, there are navigation links for "BARI BOLOGNA FIRENZE GENOVA MILANO NAPOLI PALERMO PARMA ROMA TORINO" and a search bar. The main navigation menu includes "Home", "Politica", "Economia", "Sport", "Spettacoli", "Tecnologia", "Motori", and "Tutte le sezioni". Below the navigation menu, there are several news articles. The first article is "Hillary vince in Kentucky, l'Oregon a Sanders" with a sub-headline "E il caso Nevada agita i democratici: risse e denunce di brogli Trump senza rivali: 'Sono pronto a incontrare Kim Jong-un'". The second article is "Guzzanti è Mario Bamba ecco l'editoriale supercazzola" with a sub-headline "Video il trailer del nuovo show Foto tutti i volti del genio comico".

# Iper testi e HTML – 4/7

---

Quando si naviga il Web, le pagine che si visualizzano sono soltanto documenti di testo, immagini ed altri media, residenti su un computer remoto.

Il testo in una tipica pagina web è racchiuso tra tag HTML, che forniscono al browser le informazioni sulla struttura del documento. Sulla base di queste informazioni, il browser provvederà a visualizzare i contenuti in una modalità che rispecchi la logica e la struttura del documento.

*Un documento HTML corretto:*

```
<html>
  <head>
    <title>La mia prima pagina</title>
  </head>
  <body>
    <h1>Salve!</h1>
    <p>Grazie per aver visitato la mia pagina .</p>
    <p>Spero che sia di vostro gradimento.</p>
    <ul>
      <li><a href="Antipasti.html">Antipasti</a></li>
      <li><a href="Pizze.html">Pizze </a></li>
      <li><a href="Bevande.html">Bevande</a></li>
    </ul>
  </body>
</html>
```

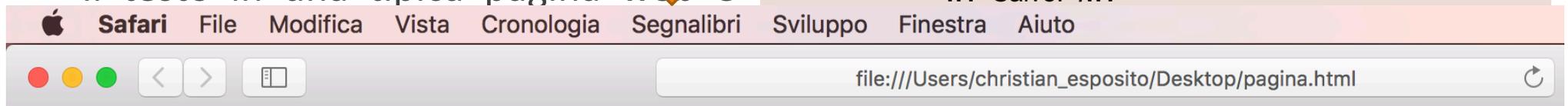
# Iper testi e HTML – 4/7

Quando si naviga il Web, le pagine che si visualizzano sono soltanto documenti di testo, immagini ed altri media, residenti su un computer remoto.

Un documento HTML corretto:

```
<html>
  <head>
    <title>La mia prima pagina</title>
  </head>
  <body>
    <h1>Salve!</h1>
```

Il testo in una tipica pagina web è



## Salve!

Grazie per aver visitato la mia pagina .

Spero che sia di vostro gradimento.

- [Antipasti](#)
- [Pizze](#)
- [Bevande](#)

# Iper testi e HTML – 5/7

---

I browser visualizzano alcuni elementi HTML, applicando stili differenti e molto elementari (i titoli sono grandi e in grassetto, i paragrafi sono seguiti da una riga vuota e così via).

Per andare oltre questo rendering di struttura semplice, si possono utilizzare i fogli di stile scritti in CSS (Cascading Style Sheets), linguaggio per definire la presentazione visuale di un documento HTML.

L'esempio sottostante:

```
body { color: red; }
```

mostra un'istruzione CSS o regola che indica al browser di visualizzare qualsiasi elemento di testo della sezione body utilizzando il colore rosso.

Ma dove si inseriscono le regole CSS?

# Iper testi e HTML – 6/7

Invece di utilizzare la regola CSS vista in precedenza, si può utilizzare l'attributo text del tag di apertura <body>, in questo modo:

```
<html>
  <head>
    <title>Il menù della casa</title>
    <style>
      body {
        color: red;
        background-color: #808080;
        font-style: italic;
        font-weight: bold;
        font-family: Arial;
      }
    </style>
  </head>
  <body>
    <!-- Non si riportano le altre righe, e questo
         è un commento HTML. -->
  </body>
</html>
```

```
<body text="red">
```

Ma nella sezione <head>, si può utilizzare una coppia di tag che possono contenere le regole CSS, e precisamente:

**<style>** e **</style>**.

Questo tag, come lascia intendere il nome, è il posto dove si possono scrivere le regole CSS che si intende applicare alla pagina HTML.

# Iper testi e HTML – 6/7

Invece di utilizzare la regola CSS vista in precedenza, si può utilizzare l'attributo text del tag di apertura <body>, in questo modo:

```
<html>
  <head>
    <title>Il menù della casa</title>
    <style>
      body {
        color: red;
        background-color: #808080;
```

<body text="red">

Ma, nella sezione <head>, si può utilizzare una coppia di tag che possono contenere le regole CSS e



Safari File Modifica Vista Cronologia Segnalibri Sviluppo Finestra Aiuto

file:///Users/christian\_esposito/Desktop/pagina.html

**Salve!**

*Grazie per aver visitato la mia pagina .*

*Spero che sia di vostro gradimento.*

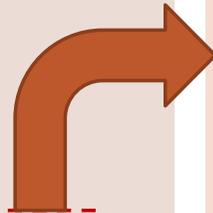
- [Antipasti](#)
- [Pizze](#)
- [Bevande](#)

# Iper testi e HTML – 7/7

Posso scrivere un file CSS per applicare le regole di rendering a più documenti HTML, che possono riferirsi ad esso

Documento: LaPrimaPagina.html

```
<html>
  <head>
    <title>Il menù della casa</title>
    <link rel="stylesheet" href="screen.css" type="text/css" />
  </head>
  <body>
    <h1 class="loud">Salve!</h1>
    <p id="highlight">Il menù della casa</p>
    <p>Spero che sia di vostro gradimento.</p>
    <ul>
      <li><a href="Antipasti.html">Antipasti</a></li>
      <li class="loud"><a href="Pizze.html">Pizze</a></li>
      <li><a href="Bevande.html">Bevande</a></li>
    </ul>
  </body>
</html>
```



Foglio di stile: Screen.css

```
body {
  color: red;
  background-color: #808080;
  font-style: italic;
  font-weight: bold;
  font-family: Arial;
}
a { font-style: italic; }

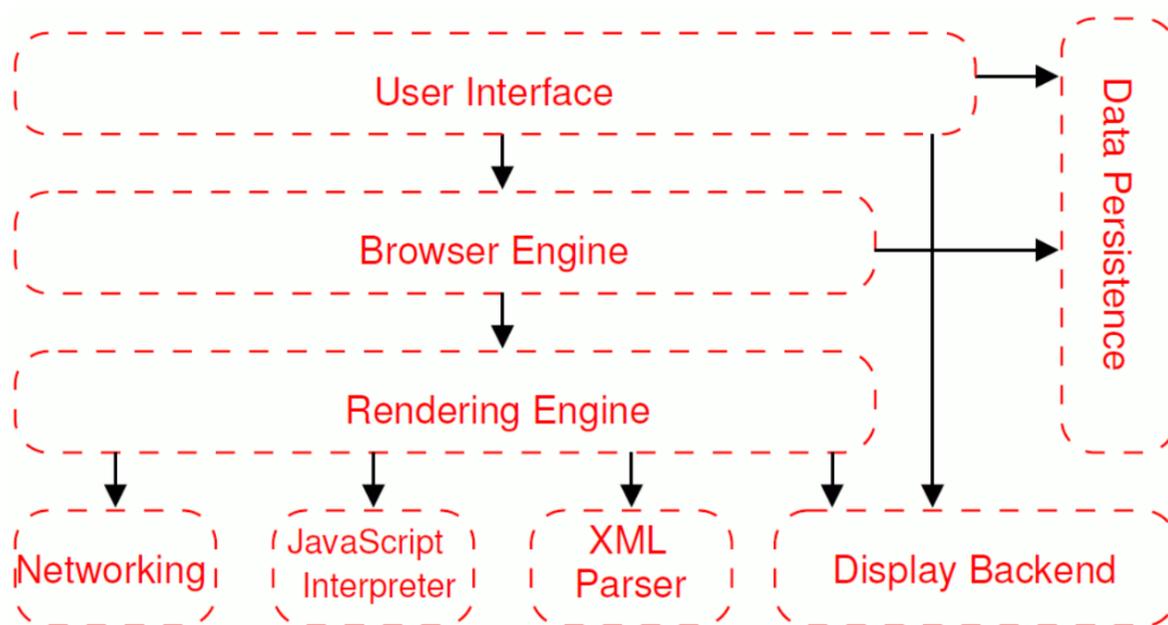
h1 a { font-style: italic; }

.loud { font-style: italic; }

#highlight { background-color: yellow;
}
```

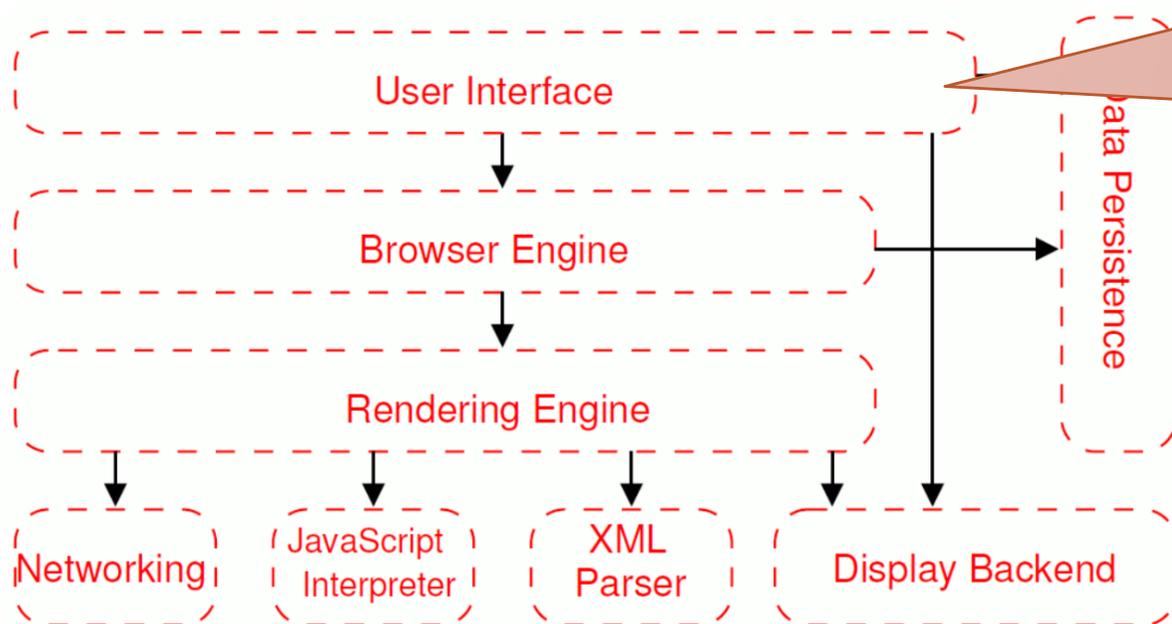
# Web Browsers – 1/3

Il web browser (o più semplicemente browser) è un'applicazione per il recupero, la presentazione e la navigazione di risorse sul web. Consentono l'accesso a documenti ipertestuali, presentandoli con l'interpretazione dei tag HTML, e la navigazione (sfogliandole) di pagine Web seguendo i collegamenti ipertestuali.



# Web Browsers – 1/3

Il web browser (o più semplicemente browser) è un'applicazione per il recupero, la presentazione e la navigazione di risorse sul web. Consentono l'accesso a documenti ipertestuali, presentandoli con l'interpretazione dei tag HTML, e la navigazione (sfogliandole) di pagine Web seguendo i collegamenti ipertestuali.

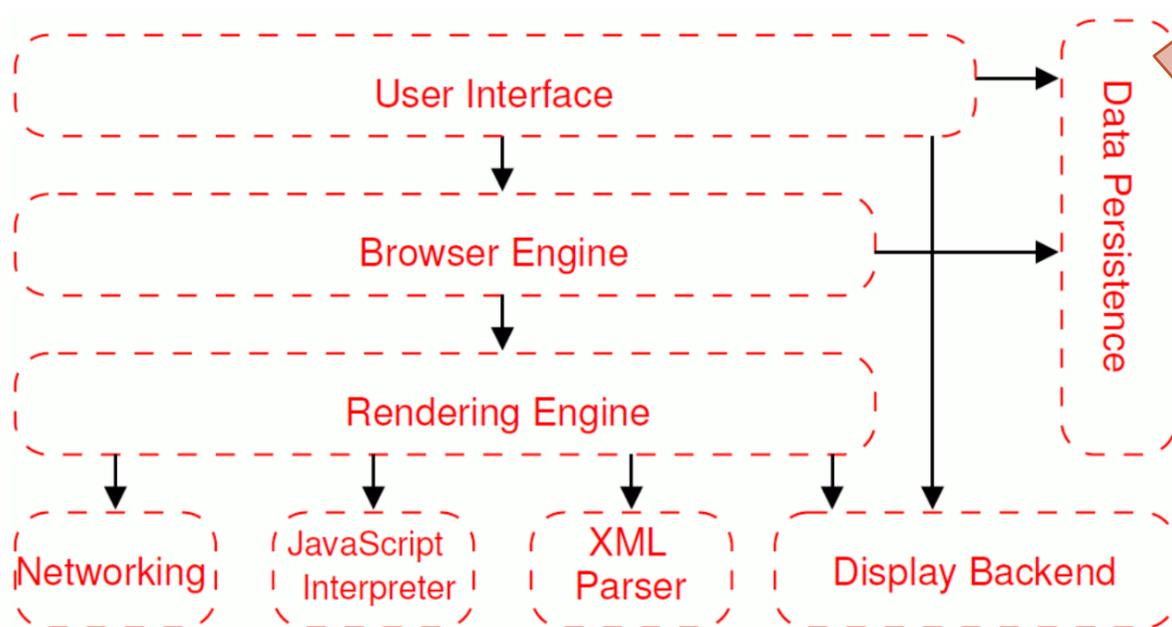


**Fornisce funzionalità per interagire con l'utente catturando i suoi input e mostrando in output i contenuti ipertestuali richiesti.**



# Web Browsers – 1/3

Il web browser (o più semplicemente browser) è un'applicazione per il recupero, la presentazione e la navigazione di risorse sul web. Consentono l'accesso a documenti ipertestuali, presentandoli con l'interpretazione dei tag HTML, e la navigazione (sfogliandole) di pagine Web seguendo i collegamenti ipertestuali.

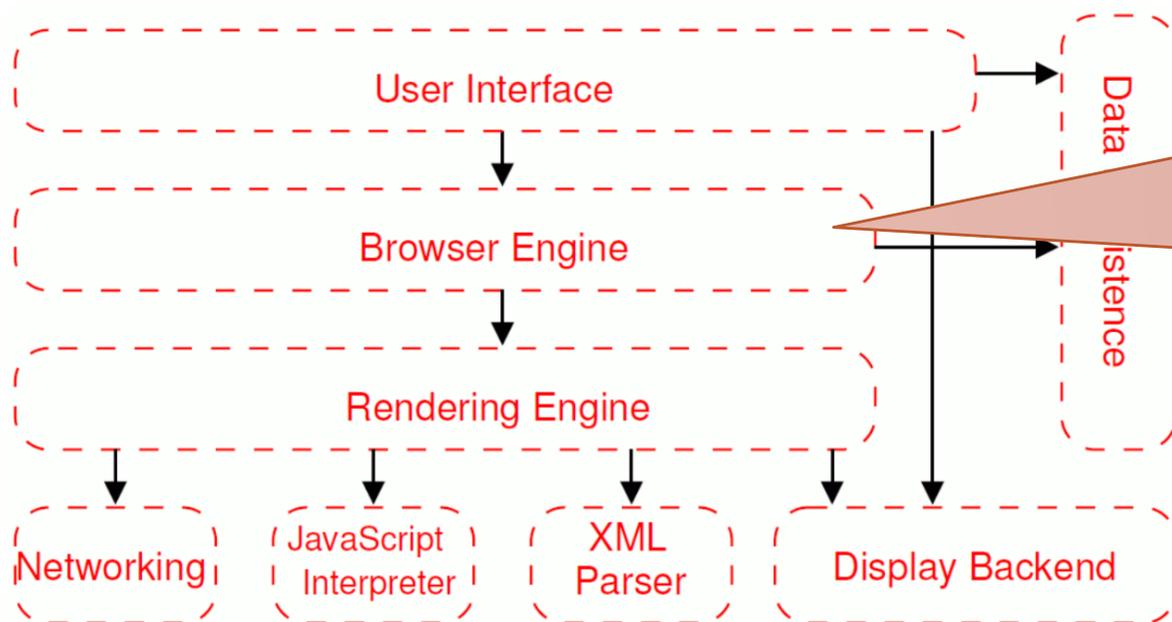


**Memorizza in maniera persistente dati rilevanti rispetto a successive navigazioni degli utenti, come password, storico delle navigazioni ed altro.**



# Web Browsers – 1/3

Il web browser (o più semplicemente browser) è un'applicazione per il recupero, la presentazione e la navigazione di risorse sul web. Consentono l'accesso a documenti ipertestuali, presentandoli con l'interpretazione dei tag HTML, e la navigazione (sfogliandole) di pagine Web seguendo i collegamenti ipertestuali.

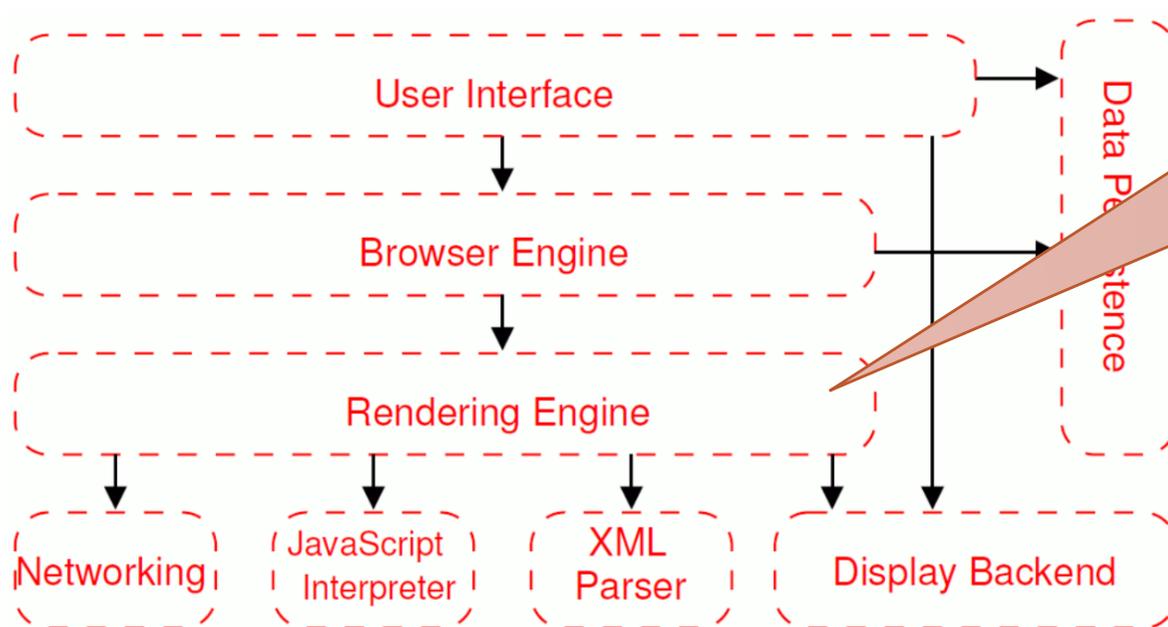


**Rappresenta il cervello del browser: gestendo la pila delle pagine visitate, permettendo back e forward, mantiene le impostazioni utente, e controlla i plugin.**



# Web Browsers – 1/3

Il web browser (o più semplicemente browser) è un'applicazione per il recupero, la presentazione e la navigazione di risorse sul web. Consentono l'accesso a documenti ipertestuali, presentandoli con l'interpretazione dei tag HTML, e la navigazione (sfogliandole) di pagine Web seguendo i collegamenti ipertestuali.



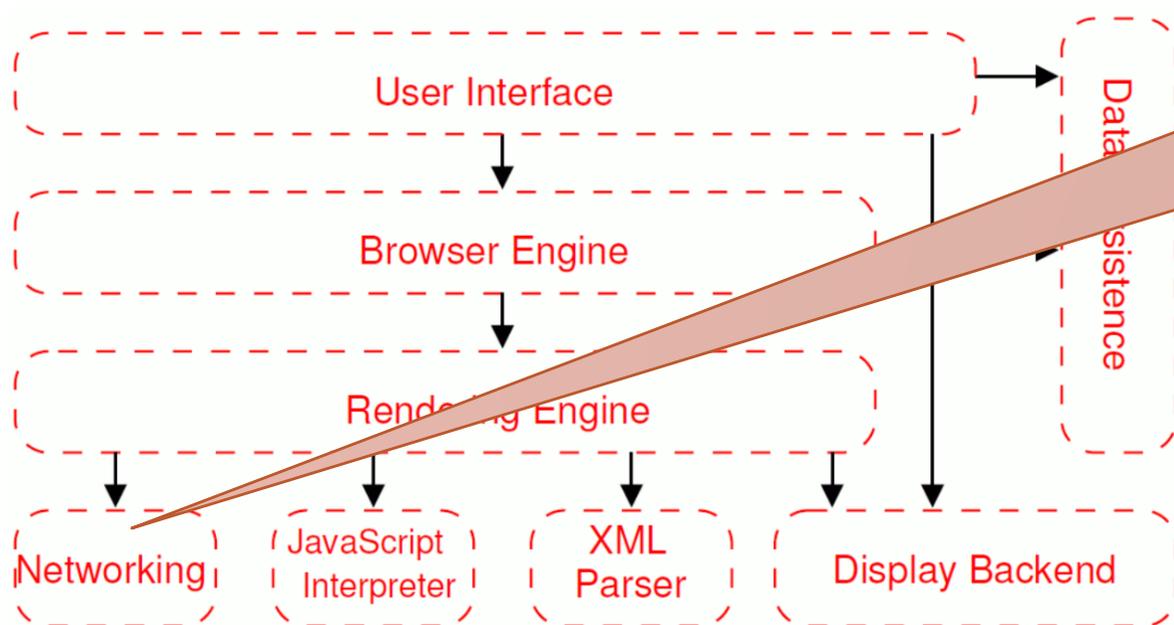
**Interpreta i tag HTML e i fogli di stile dei documenti ipertestuali e determina come rappresentare l'output e la resa grafica dei documenti.**



# Web Browsers – 1/3

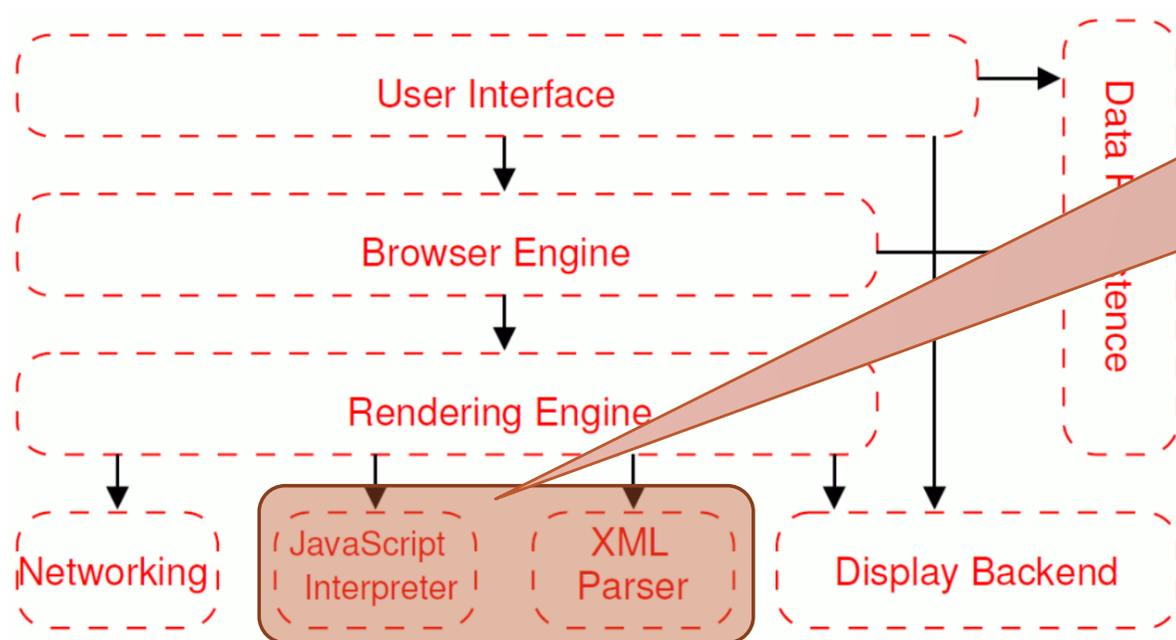
Il web browser (o più semplicemente browser) è un'applicazione per il recupero, la presentazione e la navigazione di risorse sul web. Consentono l'accesso a documenti ipertestuali, presentandoli con l'interpretazione dei tag HTML, e la navigazione (sfogliandole) di pagine Web seguendo i collegamenti ipertestuali.

**Implementa il client del protocollo HTTP per l'istituzione della connessione, l'invio delle richieste e la ricezione delle risposte.**



# Web Browsers – 1/3

Il web browser (o più semplicemente browser) è un'applicazione per il recupero, la presentazione e la navigazione di risorse sul web. Consentono l'accesso a documenti ipertestuali, presentandoli con l'interpretazione dei tag HTML, e la navigazione (sfogliandole) di pagine Web seguendo i collegamenti ipertestuali.

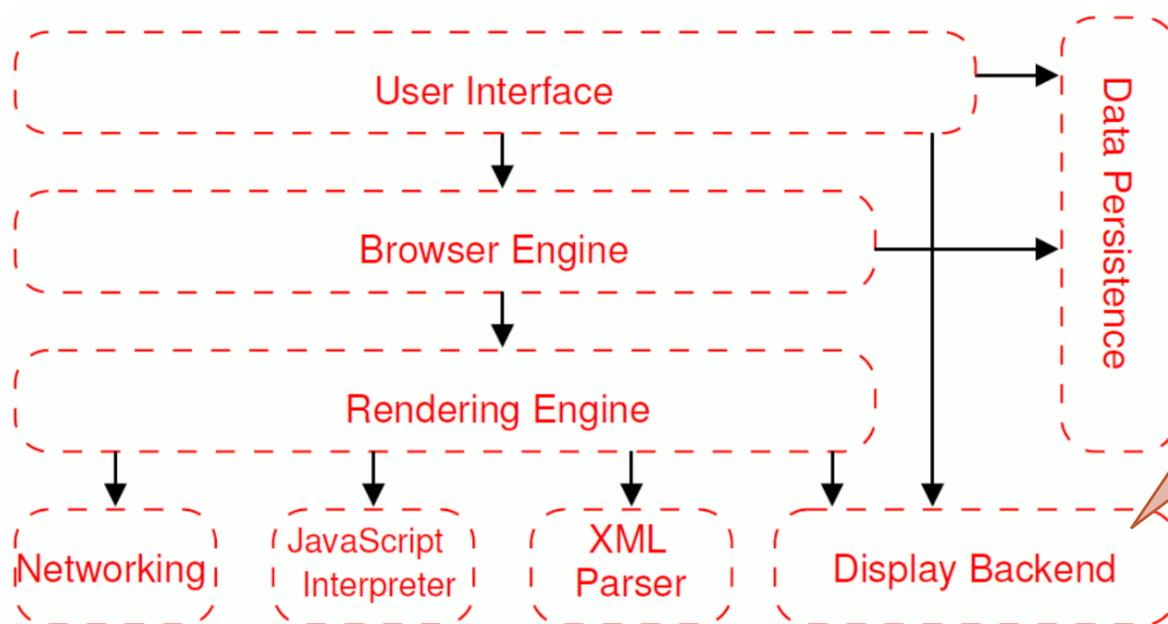


**Componenti per l'interpretazione ed esecuzione di codice JavaScript per la dinamicità dei documenti ipertestuali e per elaborare documenti in XML.**



# Web Browsers – 1/3

Il web browser (o più semplicemente browser) è un'applicazione per il recupero, la presentazione e la navigazione di risorse sul web. Consentono l'accesso a documenti ipertestuali, presentandoli con l'interpretazione dei tag HTML, e la navigazione (sfogliandole) di pagine Web seguendo i collegamenti ipertestuali.



**Interagire con il sottosistema grafico del Sistema Operativo per tradurre i comandi di visualizzazione impartiti dal Rendering engine.**



# Web Browsers – 2/3

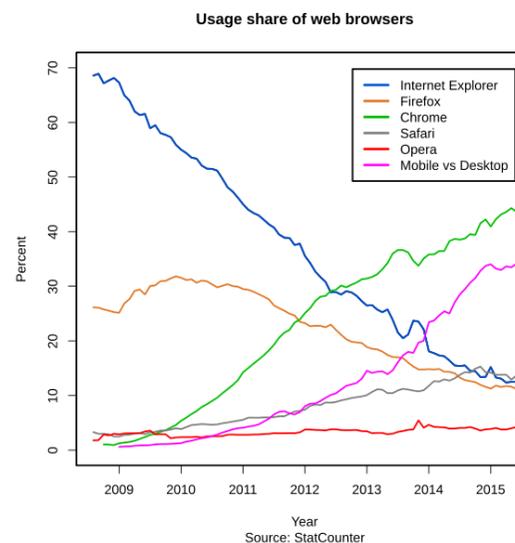
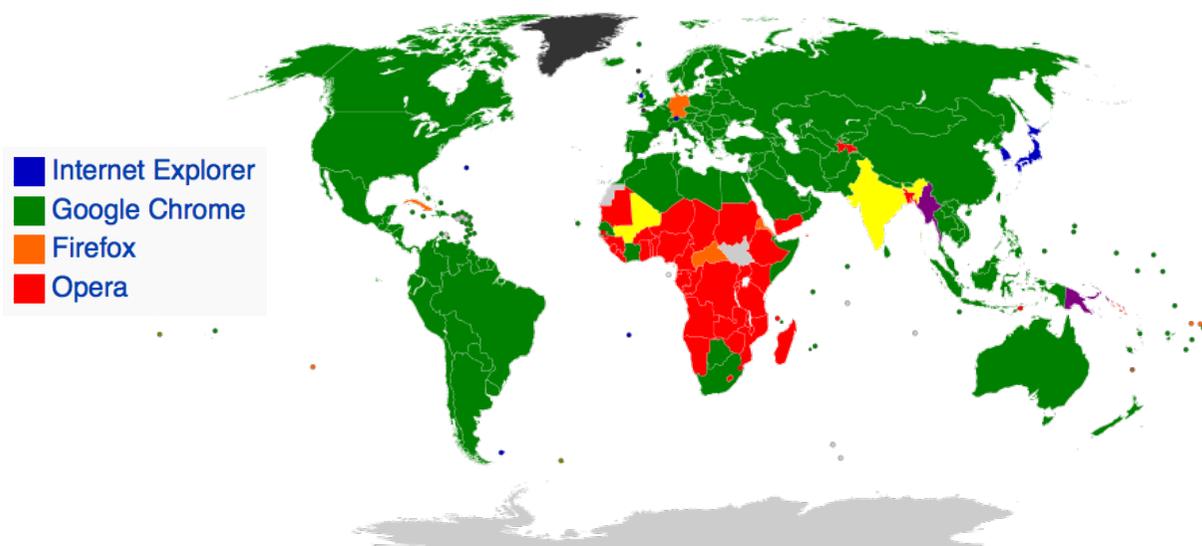
---

Le principali funzionalità dei browser disponibili includono:

- navigazione a schede (Tabbed browsing);
- supporto alla navigazione off-line tramite la memoria cache e plug-in dedicati per mantenere i link tra le pagine salvate;
- funzione di download manager con arresto/ripresa sempre tramite la memoria cache antepresa delle pagine da scaricare;
- sintesi vocale;
- integrazione dei feed RSS e di client di posta elettronica o di chat o di assistenza remota installazione di componenti aggiuntivi ed estensioni per diversi scopi;
- comando di pulizia (cache, cookie, cronologia, ecc);
- barre comandi di moltissime applicazioni.

# Web Browsers – 3/3

I primi browser (1989-1991) erano solo testuali: pagine senza grafica, bassa usabilità. Nel 1992 è comparso Mosaic, il primo browser grafico comandato via mouse, comportando l'esplosione della popolarità del WWW. L'evoluzione tecnologica è stata per molti anni rallentata dalla browser war tra Internet Explorer e Netscape Navigator, che ha visto il primo imporsi a livello mondiale. Negli ultimi anni si sono avute innovazioni significative e l'affermarsi di nuove soluzioni.



# Applicazioni web – 1/9

---

Se i browser svolgono il ruolo di client nel web, all'estremo opposto troviamo applicazioni e servizi web, che possono essere identificati e consultati dai browser per la fruizione di risorse.

Un'**applicazione web** indica genericamente tutte le applicazioni distribuite nel web, ovvero accessibile/fruibile via web per mezzo di una rete di computer, appoggiandosi ai consueti protocolli di rete. Questo modello applicativo è divenuto piuttosto popolare alla fine degli anni novanta con la diffusione di Internet, in un contesto di web dinamico.

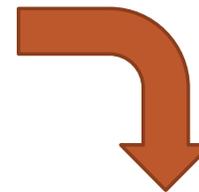
Inizialmente, il web consisteva nella condivisione di documenti ipertestuali, memorizzati presso un server HTTP, ed ottenibile da un client HTTP effettuando una dovuta richiesta. Tale visione era alquanto statica nel senso che il contenuto del documento era fissato ed invariabile.

# Applicazioni web – 2/9

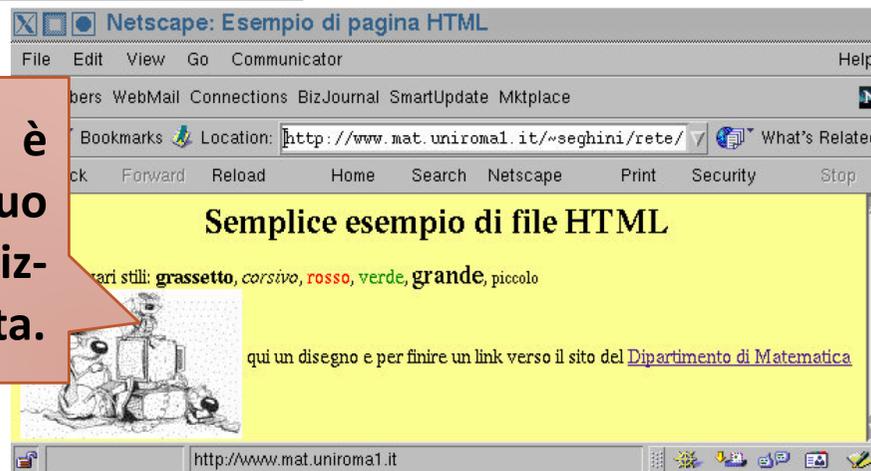
```
<HTML>
<HEAD>
  <TITLE>Esempio di pagina HTML</TITLE>
</HEAD>

<BODY BGCOLOR=#ffff99>
  <CENTER><H1>Semplice esempio di file HTML</H1></CENTER>
  <P>
    Usiamo vari stili:
    <B>grassetto</B>,
    <I>corsivo</I>,
    <FONT COLOR=red>rosso</FONT>,
    <FONT COLOR=green>verde</FONT>,
    <BIG>grande</BIG>,
    <SMALL>piccolo</SMALL>
    <BR>
    <img alt="topino" src=topino.jpg align=middle>
    qui un disegno e per finire un link verso il sito del
    <A HREF="http://www.mat.uniroma1.it">
      Dipartimento di Matematica</A>
  </BODY>
</HTML>
```

Il documento ipertestuale è composto da una serie di tag e di informazioni e memorizzato in un file con estensione “.html”.



Ogni volta che il documento è acceduto da un browser il suo rendering, ovvero la visualizzazione sullo schermo, è invariata.



# Applicazioni web – 3/9

---

Il termine **web dinamico** identifica tutte quelle applicazioni web che interagiscono attivamente con l'utente modificando le informazioni mostrate in base alle informazioni ricevute dall'utente stesso.

Molti siti web sono considerati "dinamici" dato che consentono all'utente di personalizzare l'impaginazione o le informazioni mostrate o aggiornare in maniera efficiente (dinamica appunto) i contenuti modo da renderli più vicine alle richieste dell'utente o del programmatore web.

Per realizzare tale scenario interattivo e dinamico si fa frequentemente uso di applicazioni web quali applet, servlet, CMS ecc., tramite opportuni programmi detti script, scritti in vari possibili linguaggi di scripting quali JavaScript, PHP, ASP, .NET, inseriti tipicamente all'interno della pagina web HTML e che, su particolare richiesta dell'utente, vengono attivati ed elaborati lato client tramite browser o lato server restituendo il contenuto dinamico sotto forma di codice HTML interpretato poi dal browser e visualizzato all'utente.

# Applicazioni web – 4/9

---

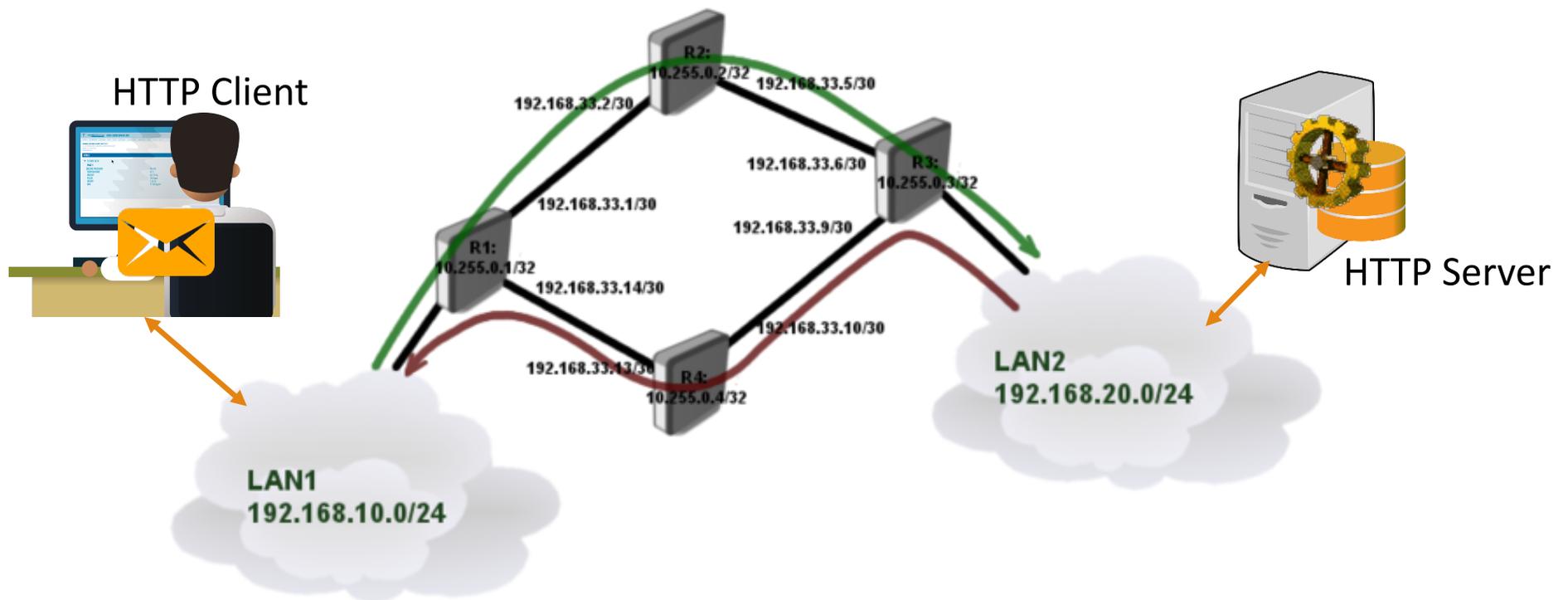
Esistono due approcci diversi per l'implementazione del web dinamico:

- il primo viene definito **server-side**: il server HTTP alla ricezione di richieste HTTP, non si limita alla ricerca di un documento HTML ed alla sua restituzione, ma effettua una elaborazione considerando come input la richiesta cliente e restituendo in output un documento HTML adattato alle esigenze del client.

# Applicazioni web – 4/9

Esistono due approcci diversi per l'implementazione del web dinamico:

- il primo viene definito **server-side**.

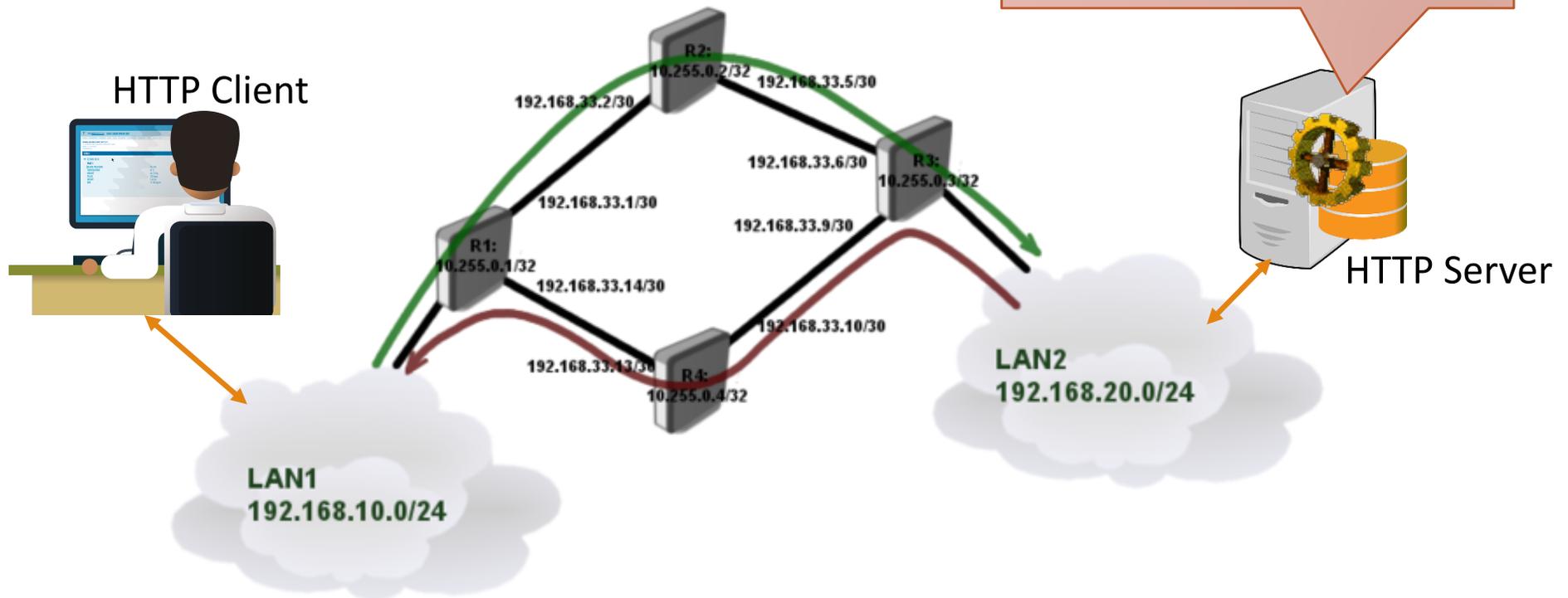


# Applicazioni web – 4/9

Esistono due approcci diversi per l'implementazione

- il primo viene definito **server-side**.

**È necessario che il server disponga di un interprete del linguaggio di scripting selezionato.**

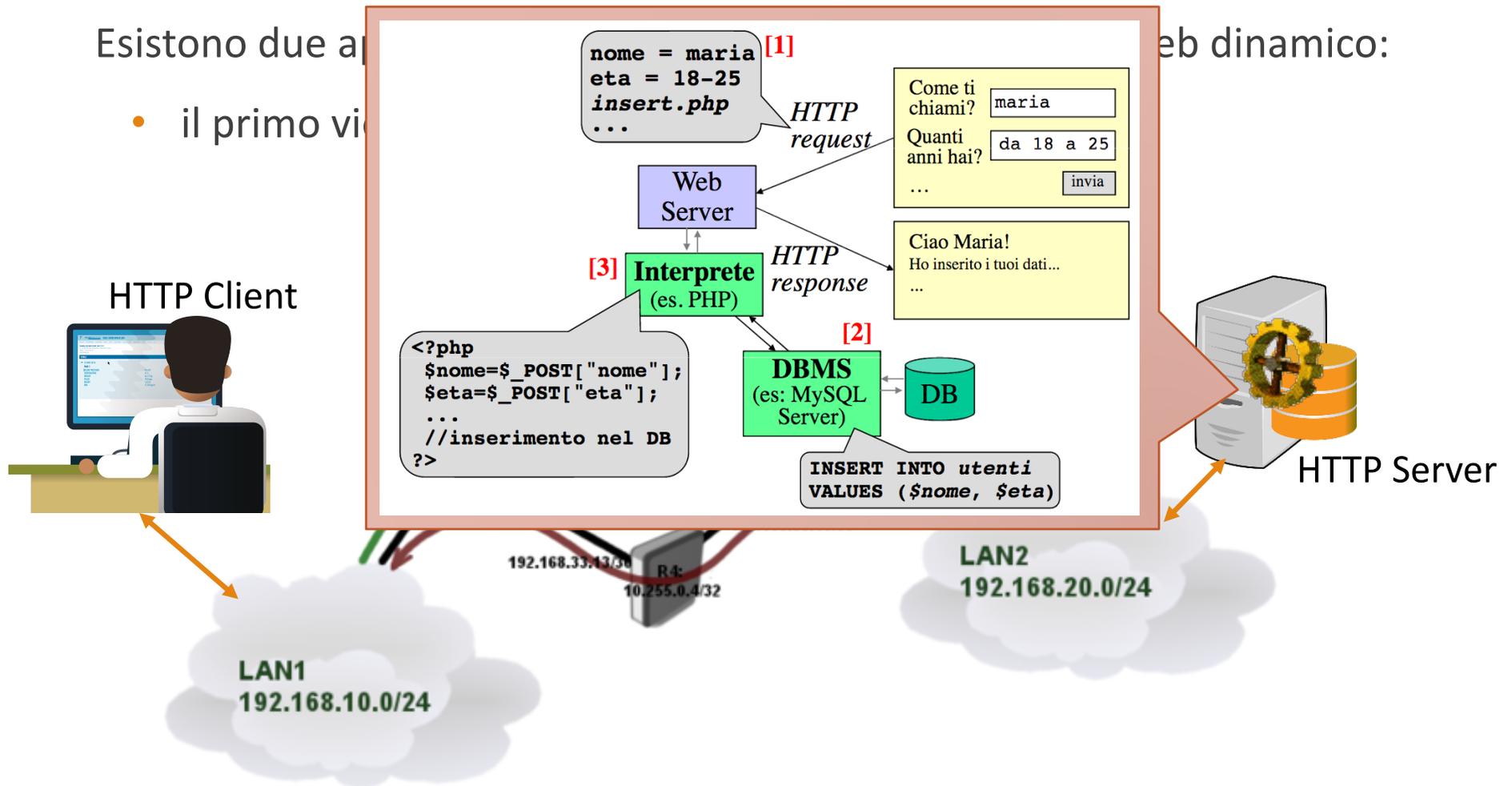


# Applicazioni web – 4/9

Esistono due architetture per applicazioni web dinamiche:

- il primo vi

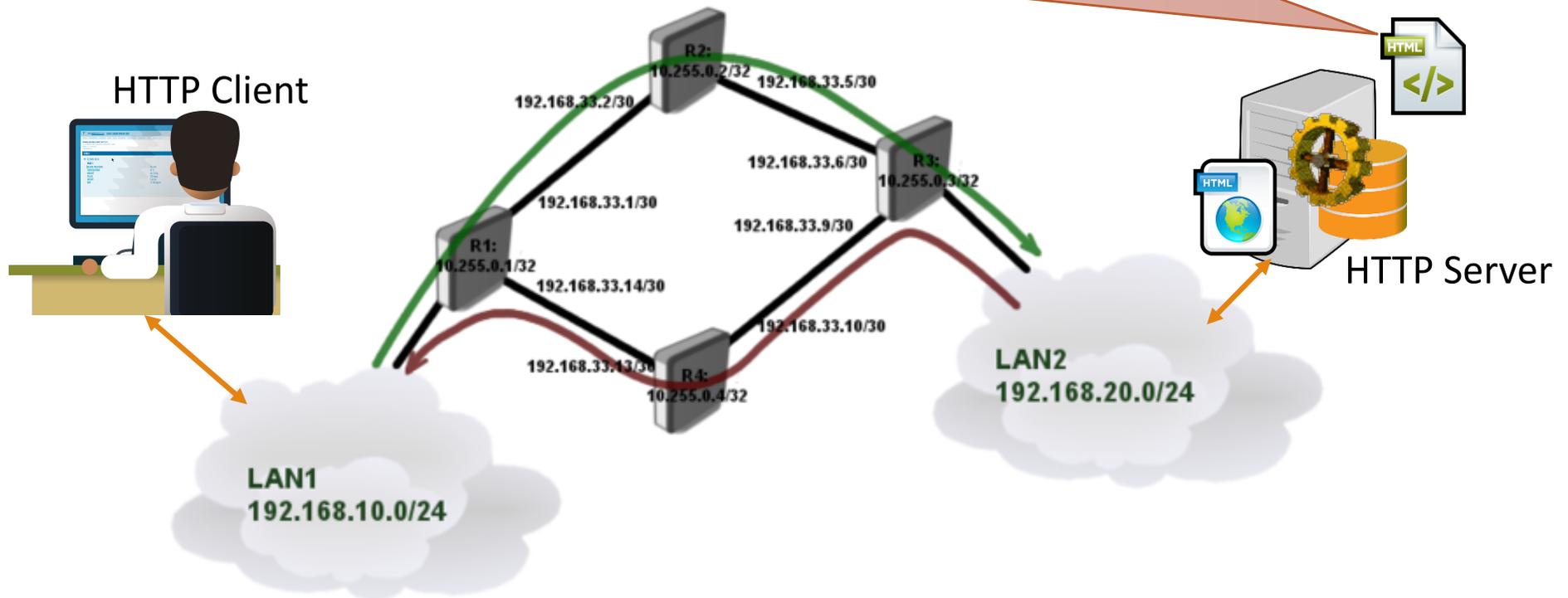
web dinamico:



# Applicazioni web – 4/9

Es

Il server HTTP mantiene una serie di documenti che non contengono solo tag HTML, ma anche del codice di scripting server-side, scritto in linguaggi come ASP, Python o PHP. Il server esegue questo codice che produce in uscita tag HTML ed informazioni che arricchiscono il documento e che adattano per la richiesta utente ricevuta.



# Applicazioni web – 4/9

Esistono due approcci diversi per l'implementazione di applicazioni web:

- il primo viene definito **server-side**.

**Il documento HTML ottenuto dall'esecuzione dello scripting lato server, viene restituito al client.**



# Applicazioni web – 5/9

---

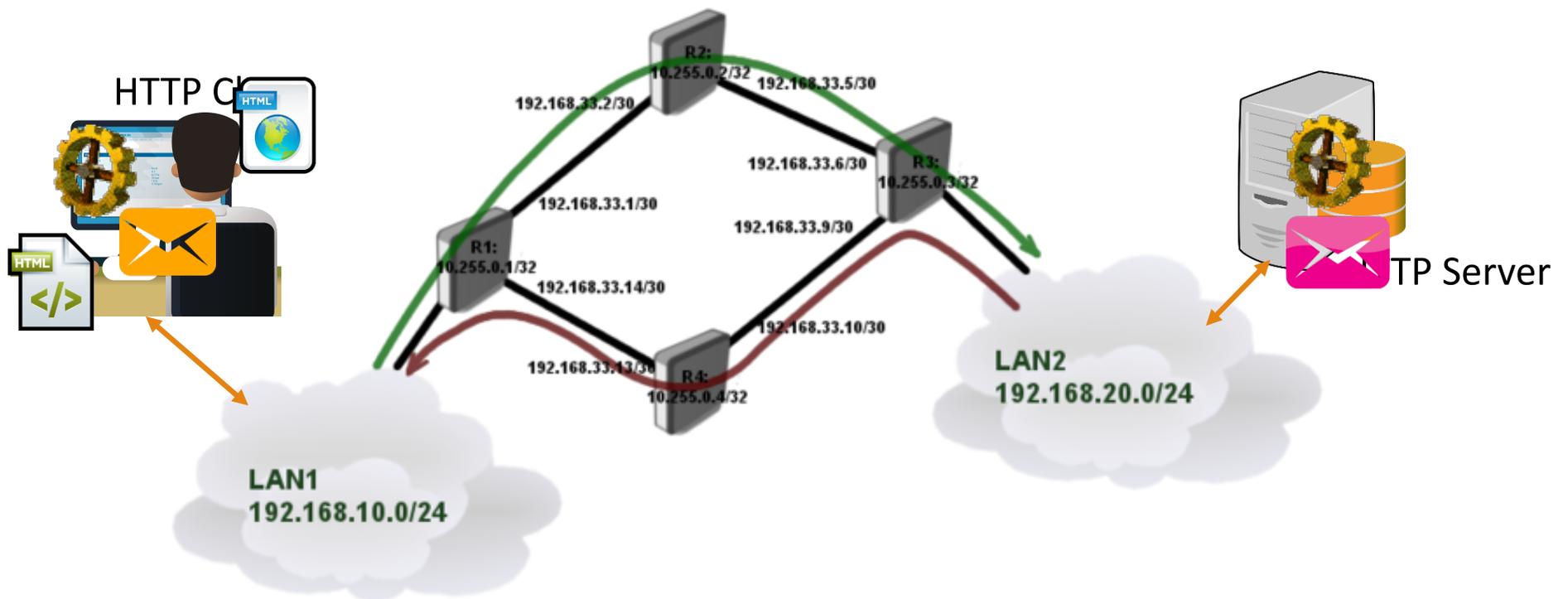
Esistono due approcci diversi per l'implementazione del web dinamico:

- in alternativa c'è un approccio **client-side**: il server HTTP alla ricezione di richieste HTTP, ricerca un documento che soddisfa la richiesta ed lo restituisce al client. Tale documento non ha solo tag HTML, ma anche del codice di scripting, espresso in linguaggi come Javascript. È cura del browser eseguire tale codice ed ottenere un documento in HTML da presentare all'utente.

# Applicazioni web – 5/9

Esistono due approcci diversi per l'implementazione del web dinamico:

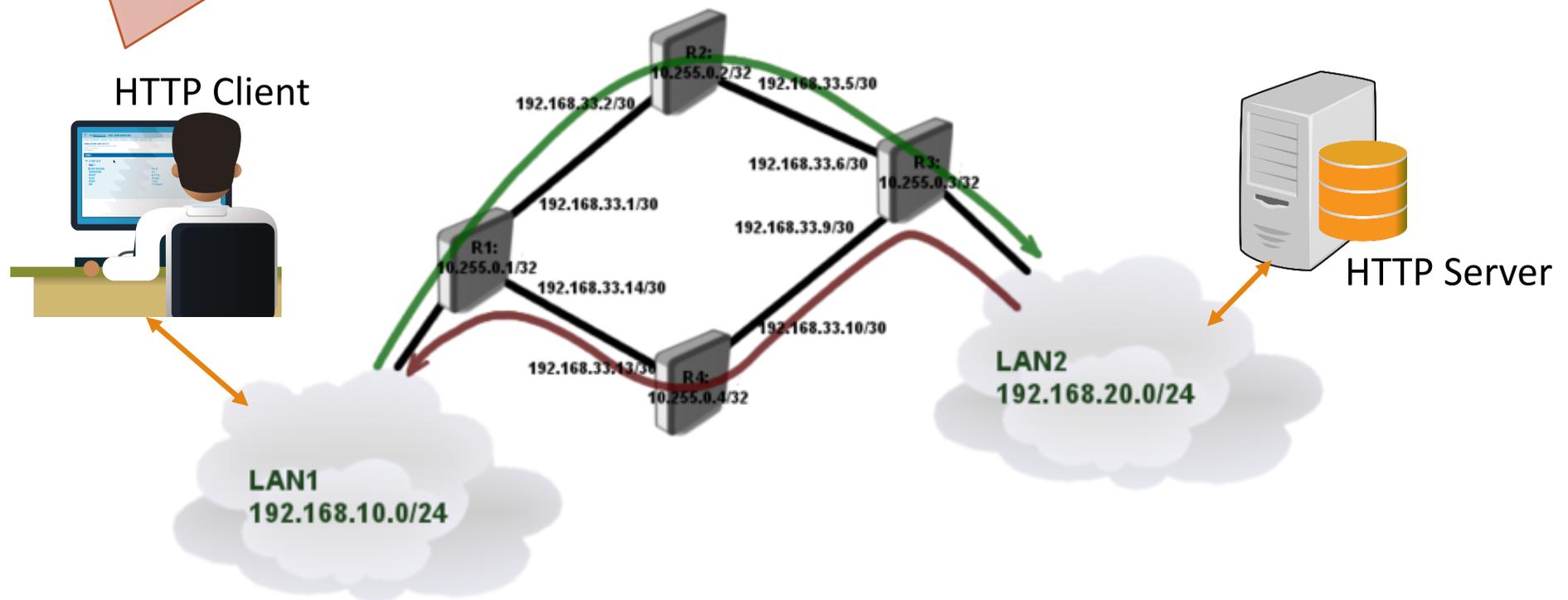
- in alternativa c'è un approccio **client-side**.



# Applicazioni web – 5/9

È necessario che il client disponga di un interprete del linguaggio di scripting selezionato.

si per l'implementazione del web dinamico: approccio client-side.



# Applicazioni web – 6/9

---

Il vantaggio dello scripting server-side è di avere un server centralizzato:

1. più facile da mantenere e tenere aggiornato;
2. più semplice da controllare e gestire gli accessi;
3. avere sia funzionalità avanzate che un accesso a datasets grandi e complessi che non sarebbero trasferibili via Internet.

Gli svantaggi si hanno quando il processore deve eseguire compiti particolarmente pesanti e quando si devono trasferire sulla rete grossi volumi di dati, che può causare:

1. il tempo di risposta può crescere considerevolmente;
2. la capacità di calcolo del client non viene sfruttata;
3. la comunicazione e l'elaborazione da parte del server è necessaria per ogni richiesta, aumentando il traffico sulla rete e il carico del server.

# Applicazioni web – 7/9

---

I vantaggi di scripting client-side stanno nei browser come thick clients:

1. riducono il carico del server e decrementano il traffico in rete;
2. danno maggiore autonomia all'utente, ad esempio per map browsing (pan, zoom), controllo della visualizzazione dei layers, input di query spaziali...;
3. permettono il trasferimento di dati in forma vettoriale.

Gli svantaggi del client-side sono:

1. Lento trasferimento del documento: la dimensione del file è proporzionale alle funzionalità e l'utente può non essere disposto ad aspettare;
2. installazione di plug-in del browser, come per applet, e tempo e sforzo extra per l'installazione e il mantenimento (aggiornamenti eccetera);
3. downloading iniziale di datasets anche grandi
4. il computer client non avere grossa potenza di calcolo.

# Applicazioni web – 8/9

---

È possibile avere approcci ibridi combinando scripting server-side e quello client-side. Un esempio è fornito da AJAX (Asynchronous JavaScript and XML) che si basa su:

- HTML e Javascript per la programmazione Web lato client
- scripting server-side con PHP, Servlet, ...

Vantaggi di AJAX:

- sono in grado di inviare al Web Server richieste asincrone (mentre l'utente può continuare ad interagire con la pagina) e parziali (relative solo ai dati necessari);
- consentono un'interazione più veloce (la quantità di dati che è necessario inviare al/ricevere dal server è minore) e in modalità asincrona(senza attesa).

# Applicazioni web – 9/9

---

Funzionamento tipico di un'applicazione AJAX:

- carichiamo nel Browser una pagina web (.html) che contiene degli script client-side (Javascript) che intercettano eventi relativi a parti della pagina, come il click di un bottone o la digitazione di testo in una casella
- in risposta ad un qualche evento, Javascript invia una richiesta HTTP "speciale" (con l'indicazione di una risorsa server-side, es. PHP)
- sul server, un programma (o uno script) server-side elabora la risposta e la invia al client come risposta della richiesta, da cui lo script client-side preleva dei dati e modifica di conseguenza una parte della pagina attualmente caricata nel browser.

# Servizi Web – 1/6

---

Il termine servizio Web è definito secondo il consorzio UDDI come *“applicazioni di business modulari e auto-consistenti che hanno interfacce aperte, Internet-oriented e basate su standard”*:

- Programmi accessibili dal Web, con un’interfaccia stabile;
- Informazioni descrittive dell’interfaccia accessibili pubblicamente;
- Programmi conformi a standard di Internet.

Un’altra definizione è quello fornito dal consorzio World Wide Web (W3C): *“applicazione software identificata da un URI, le cui interfacce e collegamenti possono essere definite, descritte e scoperte come artefatti XML. Un servizio Web supporta interazioni dirette con altri agenti software usando messaggi codificati in XML, scambiati attraverso i protocolli usati in Internet”*. Tale definizione è più accurata e precisa anche come i servizi Web funzionano. Tale funzionamento pone come base di tali servizi l’XML.

# Servizi Web – 2/6

---

Esistono altre definizioni, più tecnologiche, indicando gli standard alla base di tali servizi. Un esempio è quello presente nel dizionario tecnico chiamato Webopedia: *“un modo standardizzato per integrare applicazioni basate sul Web usando gli standard aperti XML, SOAP, WSDL e UDDI al di sopra dei protocolli fondanti di Internet. XML è usato per taggare i dati, SOAP per il loro trasferimento, WSDL per la descrizione del servizio e UDDI per l’identificazione di quali servizi sono disponibili ed accedere alla loro descrizione”*.

I servizi Web sussistono sull’assunzione che l’interazione client/server avvenga per mezzo di un modello a servizio, con le funzionalità del server accessibili come un servizio. In termini della letteratura dei middleware, un servizio è una procedura, una funzionalità o un oggetto con un’interfaccia stabile e pubblicata che può essere invocata dai client. Anche il client è un programma, e la richiesta e l’esecuzione di un servizio consiste in un programma che richiama un altro programma.

# Servizi Web – 3/6

---

Un'architettura a servizi si compone di tre entità:

- Il Service Provider, ovvero il server che ospita il servizio o lo rende accessibile stando in ascolto di possibili richieste in arrivo da Internet.

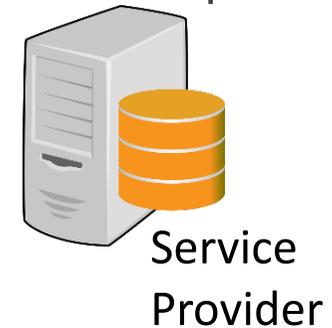


# Servizi Web – 3/6

---

Un'architettura a servizi si compone di tre entità:

- Un Registry, ovvero un server remoto che contiene le descrizioni dei servizi offerti da vari service providers. Tali descrizione contengono una descrizione dei servizi in termini di come accedervi e di quali funzionalità sono offerte e di come invocarle.



Registry

# Servizi Web – 3/6

Un'architettura a servizi si compone di tre entità:

- Un Registry, ovvero un server remoto che contiene le descrizioni dei servizi offerti da vari service providers. Tali descrizioni contengono una descrizione del servizio, di come accedervi e di quali funzioni può svolgere.

**Quando un service provider attiva un servizio può decidere di caricare una descrizione nel registry.**



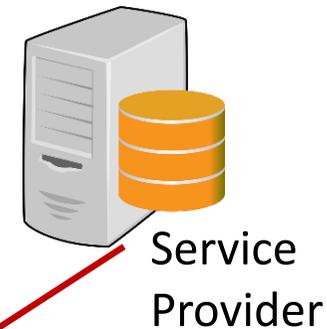
# Servizi Web – 3/6

---

Un'architettura a servizi si compone di tre entità:

- Il Service Requestor, ovvero un applicativo che è interessato ad invocare un servizio offerto da un service provider.

Service Requestor



Registry

Pubblicazione

# Servizi Web – 3/6

Un'architettura a servizi si compone di tre entità:

- Il Service Requestor, ovvero un applicativo che è interessato ad invocare un servizio

Service Requestor



**Un service requestor per determinare il servizio che risponde meglio alle sue esigenze ed ottenerne la descrizione può consultare un registry.**



Service Provider

Scoperta

Pubblicazione

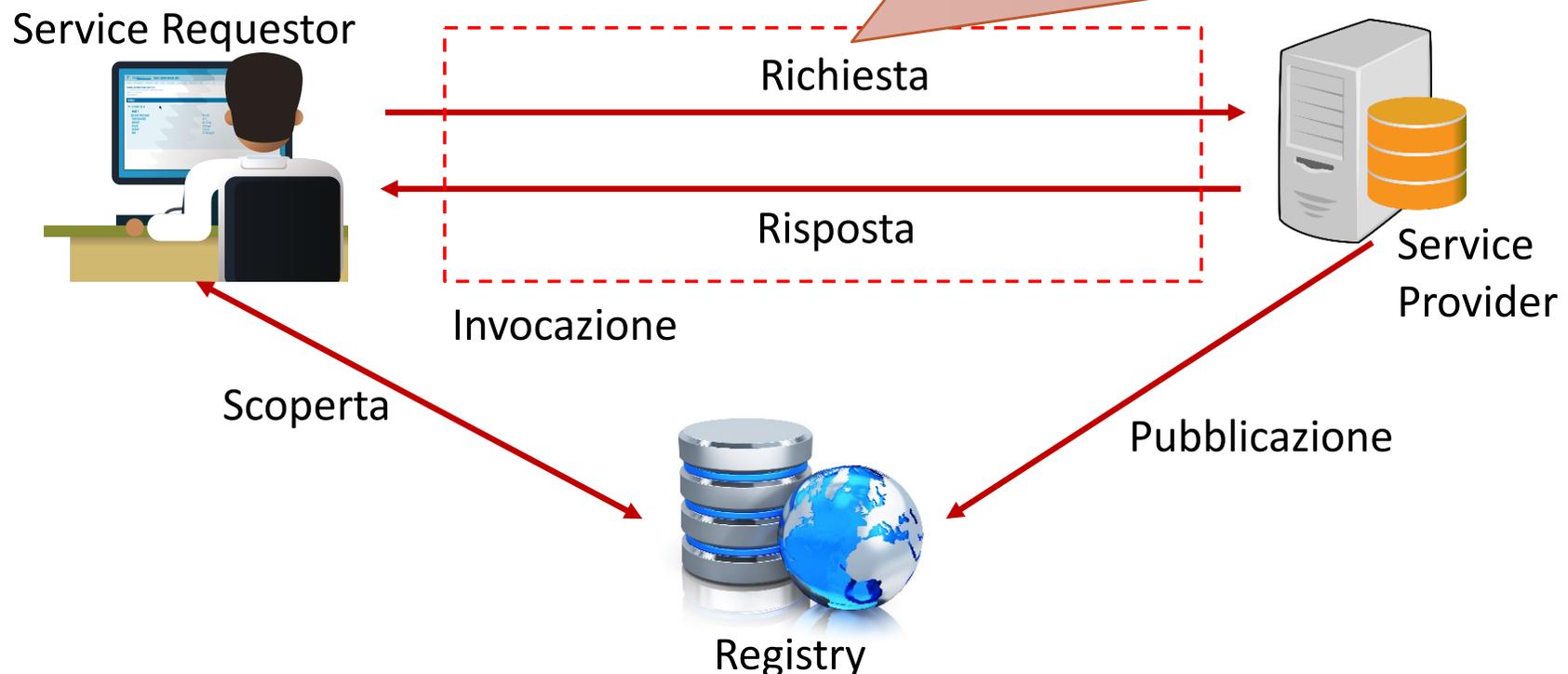


Registry

# Servizi Web – 3/6

Un'architettura a servizi si compone di tre entità:

Ottenuto l'URI del servizio, e come invocarlo, il service requestor può inviare richieste al service provider e ricevere delle risposte.



# Servizi Web – 4/6

Un servizio Web non va confuso con quelle applicazioni fruibili sul web. Inoltre, gli standard alla base dell'architettura dei Web service sono i seguenti, come anche specificato nella definizione Webopedia:



# Servizi Web – 5/6

---

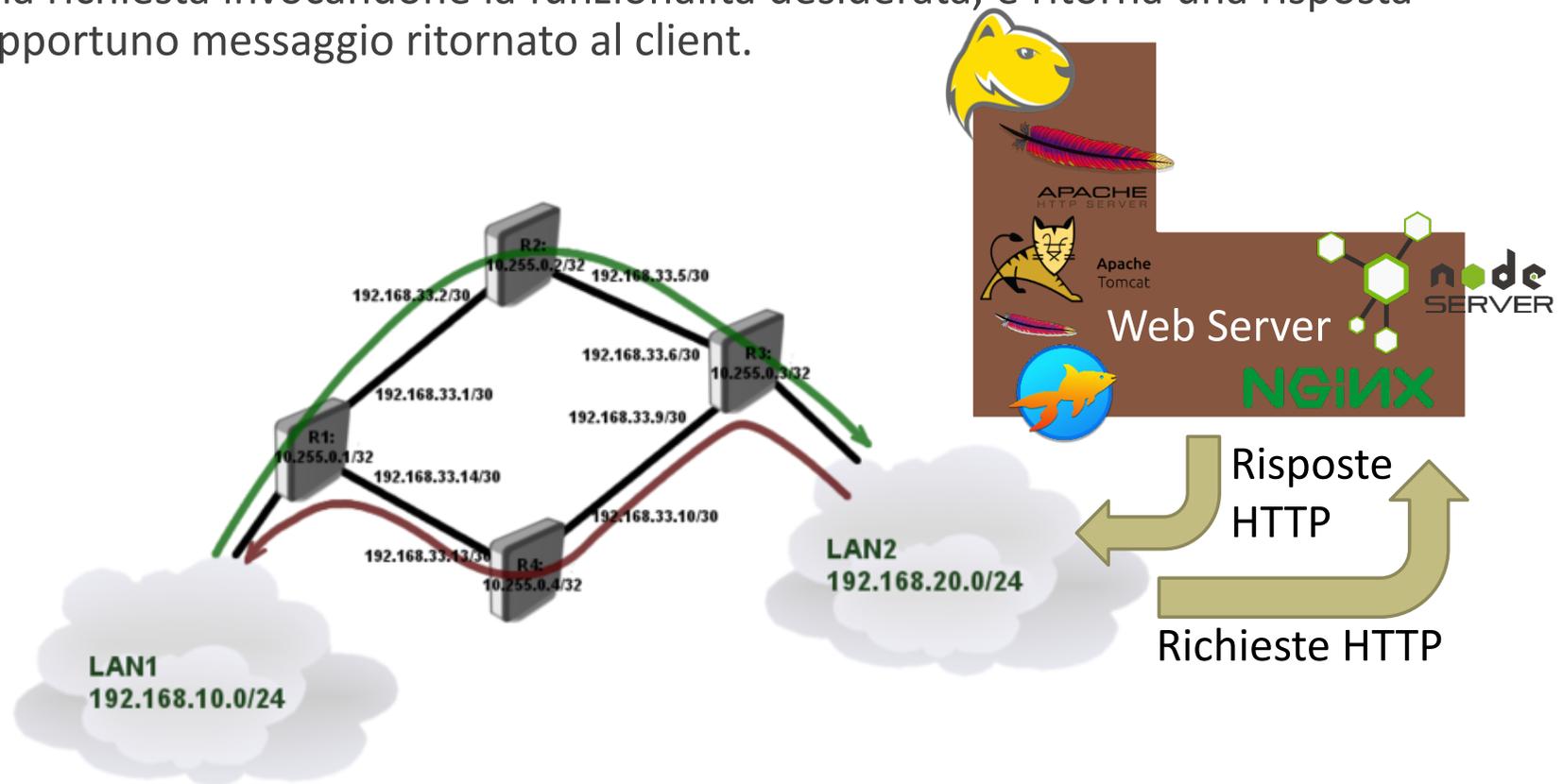
L'architettura alla base dei servizi Web si basa sul concetto di **middleware**, uno strato software che contiene un insieme di applicativi e moduli software che fungono da intermediari tra diverse applicazioni e componenti software. Supportano i sistemi distribuiti complessi, per superare l'eterogeneità delle macchine ed offrire le funzionalità base per la fruizione remota di risorse, procedure ed oggetti.

Nel caso dei servizi Web, si ha bisogno di un middleware che offra le funzionalità di invocazione, pubblicazione e scoperta dei servizi web. Nello specifico questo middleware si compone di due parti:

- Una nel provider per la ricezione di richieste, la loro elaborazione invocando il servizio web più adatto e il ritorno dell'esito, congiuntamente alla pubblicazione delle informazioni del servizio.
- Una sul lato client per supportare l'invocazione remota di servizi e la consultazione di registry per la scoperta di servizi.

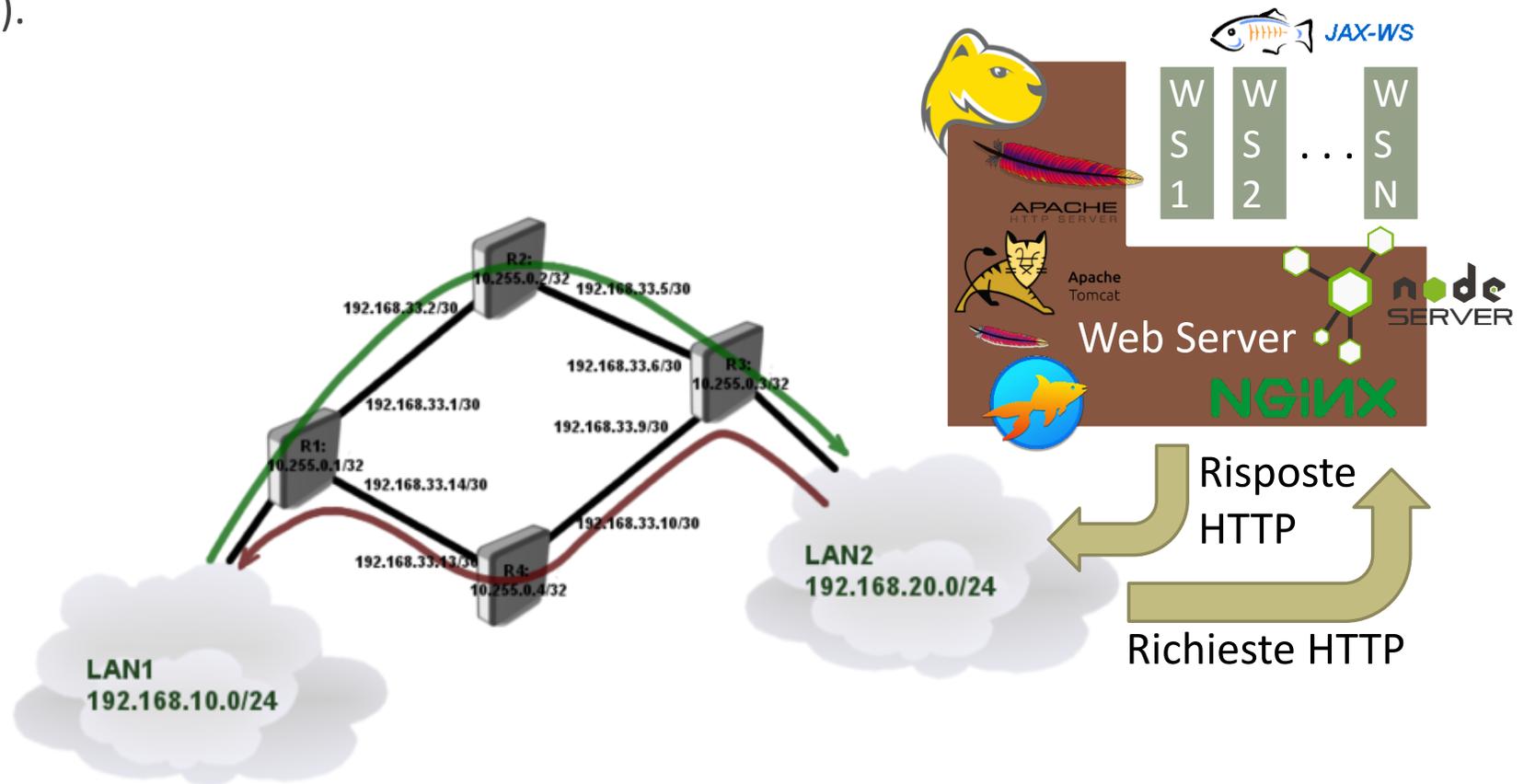
# Servizi Web – 6/6

La parte server è svolta dal web server che gestisce le richieste dei un client: riceve la richiesta mediante il protocollo HTTP, determina a quale servizio passare la richiesta invocandone la funzionalità desiderata, e ritorna una risposta come opportuno messaggio ritornato al client.



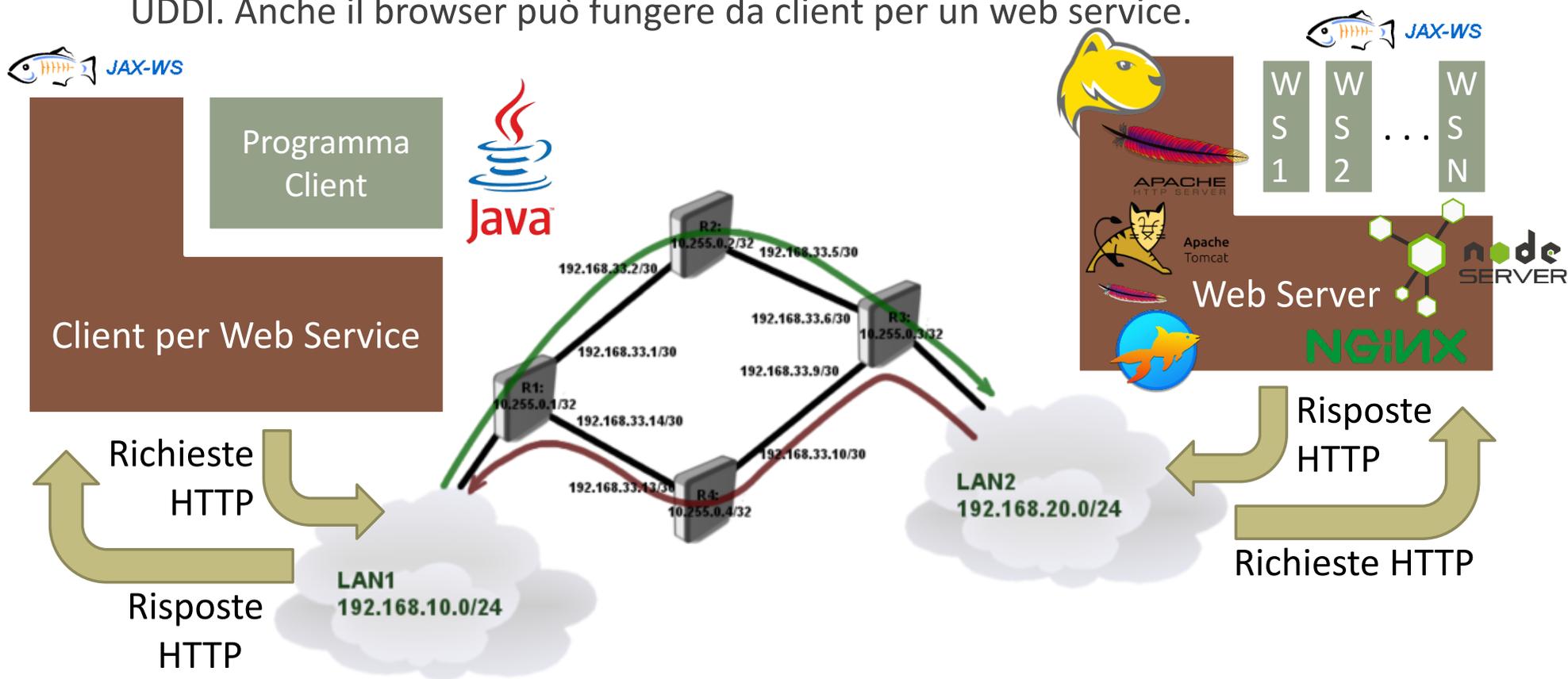
# Servizi Web – 6/6

Il web server ospita al proprio interno i servizi web, realizzate mediante un'apposita tecnologia di sviluppo software, come JAX-WS (Java API for XML Web Services).



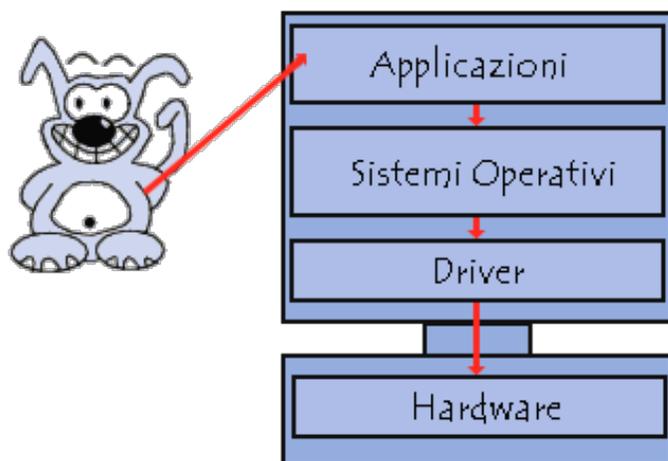
# Servizi Web – 6/6

Dal lato client, JAX-WS fornisce una serie di funzionalità che consentono ad un programma client di invocare remotamente i servizi e di interagire con il registry UDDI. Anche il browser può fungere da client per un web service.



# Sistemi Operativi – 1/9

I componenti fisici di un calcolatore sono resi accessibili agli utenti e ai programmi applicativi attraverso un complesso di strumenti software di base. Tali strumenti sono il **sistema operativo**, e sono forniti a corredo dell'hardware di un elaboratore al momento del suo acquisto.



Quando un programma vuole accedere ad una risorsa hardware, non è necessario che interagisca con la periferica, ma gli basterà inviare richieste al sistema operativo, che si occuperà di trasmetterle alla periferica interessata attraverso il suo **driver**.

Il driver permette al sistema operativo di utilizzare correttamente un dispositivo, astruendo dall'hardware dell'elaboratore.

Un driver è specifico sia riguardo al dispositivo che al sistema operativo.

# Sistemi Operativi – 2/9

---

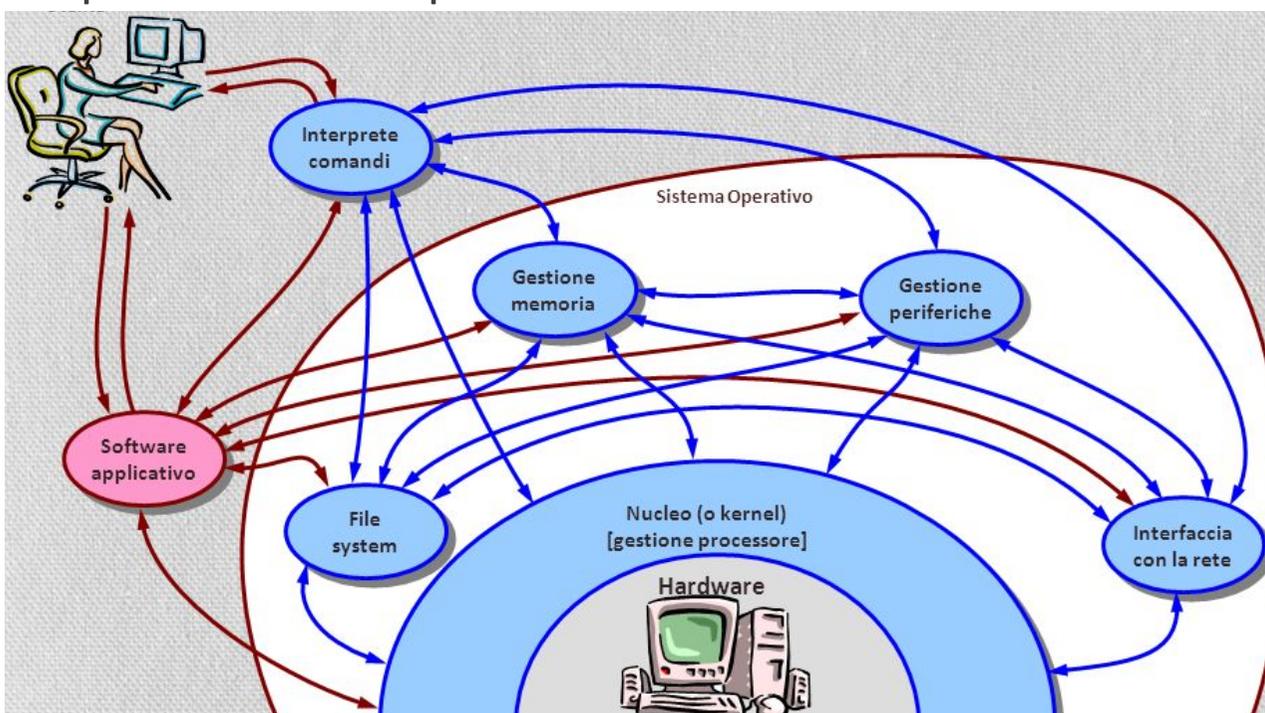
Le funzioni di un sistema operativo sono:

- garantire la correttezza nell'elaborazione e trasmissione delle informazioni;
- Consentire di superare il problema della localizzazione delle risorse;
- Garantire il massimo livello di affidabilità, disponibilità e sicurezza dei sistemi;
- Assicurare l'archiviazione e la riservatezza dei dati;
- Consentire l'interoperabilità tra i dispositivi da diversi produttori;
- Superare i problemi legati alla limitazione del numero di risorse e regolarne l'impiego evitando possibili conflitti di utilizzo;
- Gestione dell'alternanza dei programmi sul processore, e lo spazio di memoria distribuito ad ogni applicazione.

Il sistema operativo rappresenta un'interfaccia tra l'utente e l'hardware del calcolatore, e ne facilita l'uso e la gestione delle risorse.

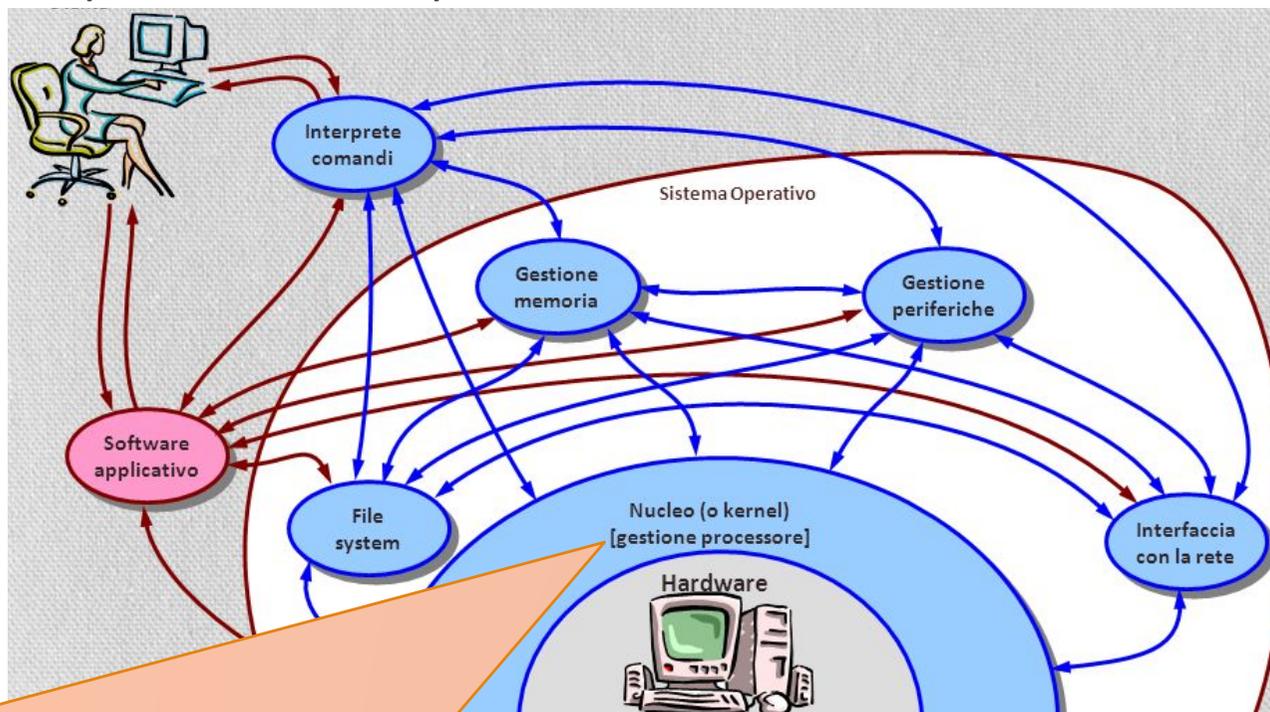
# Sistemi Operativi – 3/9

Un sistema operativo si compone di vari elementi.



# Sistemi Operativi – 3/9

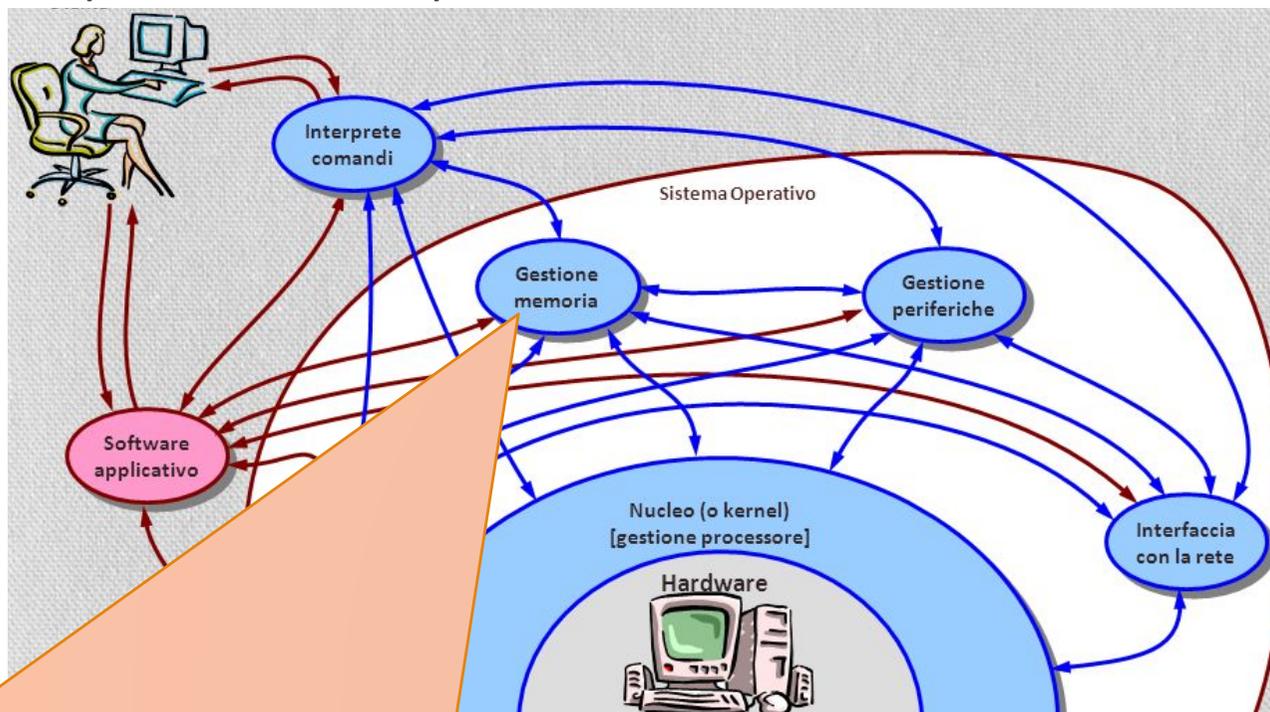
Un sistema operativo si compone di vari elementi.



Un sistema di gestione del processore definisce quali programmi sono da eseguire e quali compiti assegnare alla CPU di volta in volta e traduce i programmi in un formato direttamente comprensibile dal processore.

# Sistemi Operativi – 3/9

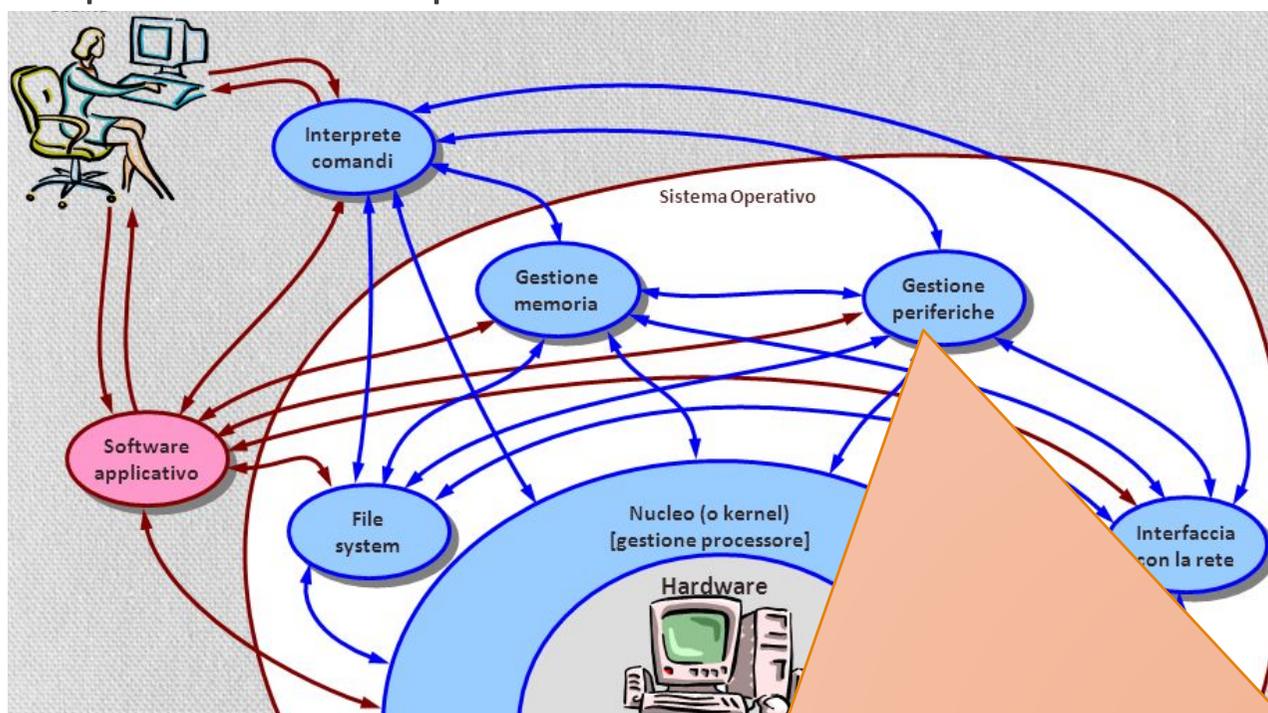
Un sistema operativo si compone di vari elementi.



Un sistema di gestione della memoria controlla l'allocazione della memoria di lavoro ai diversi programmi che sono contemporaneamente in esecuzione, garantendo zone riservate di memoria.

# Sistemi Operativi – 3/9

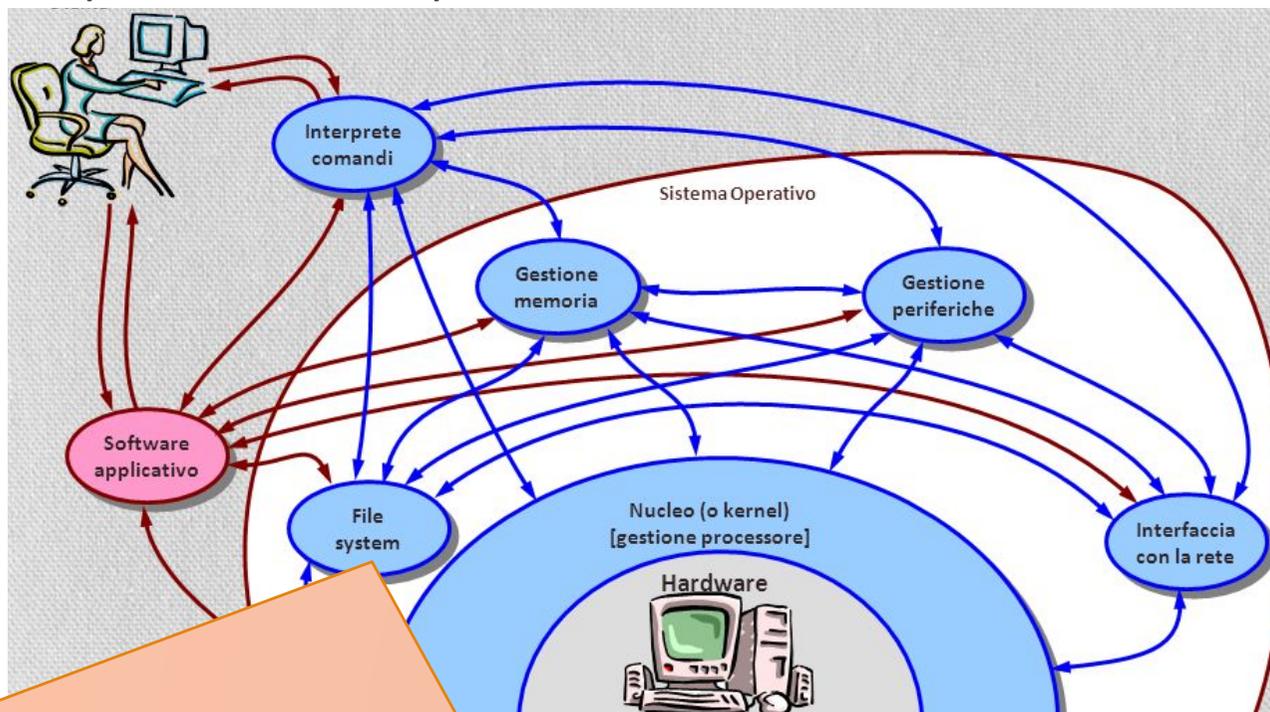
Un sistema operativo si compone di vari elementi.



Un sistema di gestione delle periferiche garantisce l'accesso ai dispositivi di ingresso/uscita, mascherandone i dettagli di basso livello e possibili conflitti di più programmi che intendono impiegare contemporaneamente un determinato dispositivo.

# Sistemi Operativi – 3/9

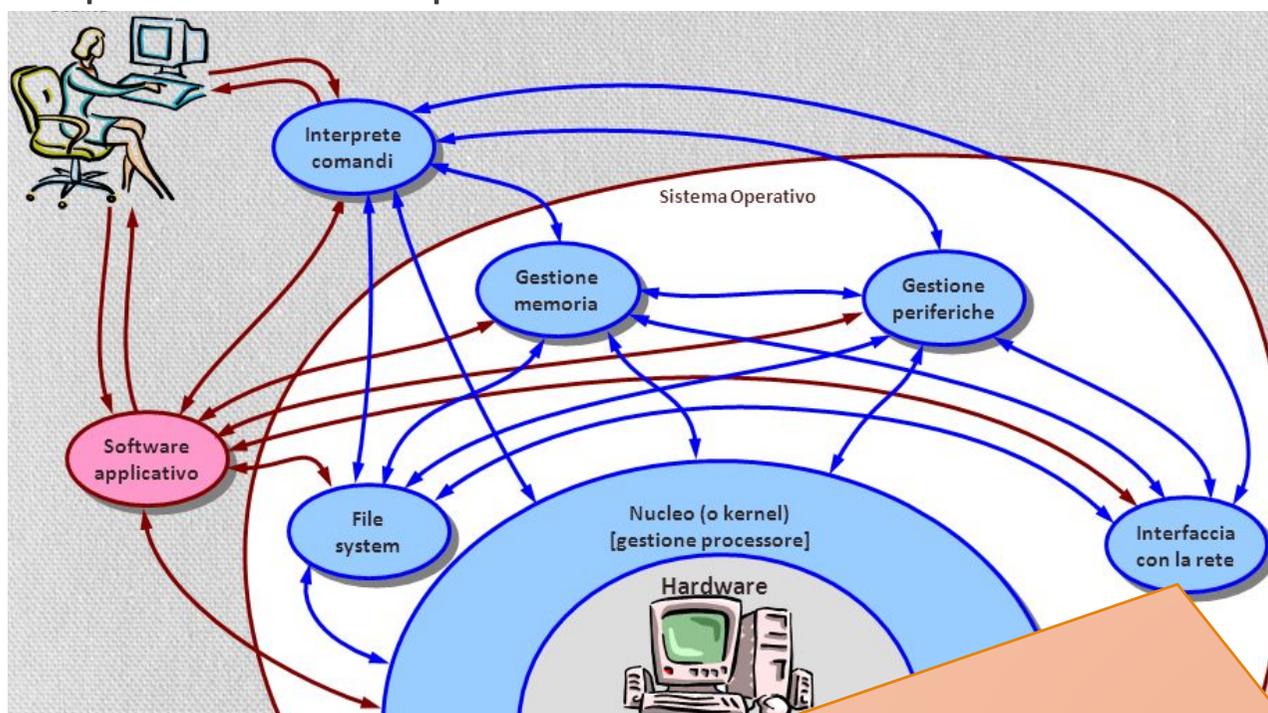
Un sistema operativo si compone di vari elementi.



Un sistema di gestione dei file consente l'archiviazione e il reperimento di dati, sfruttando le memorie di massa.

# Sistemi Operativi – 3/9

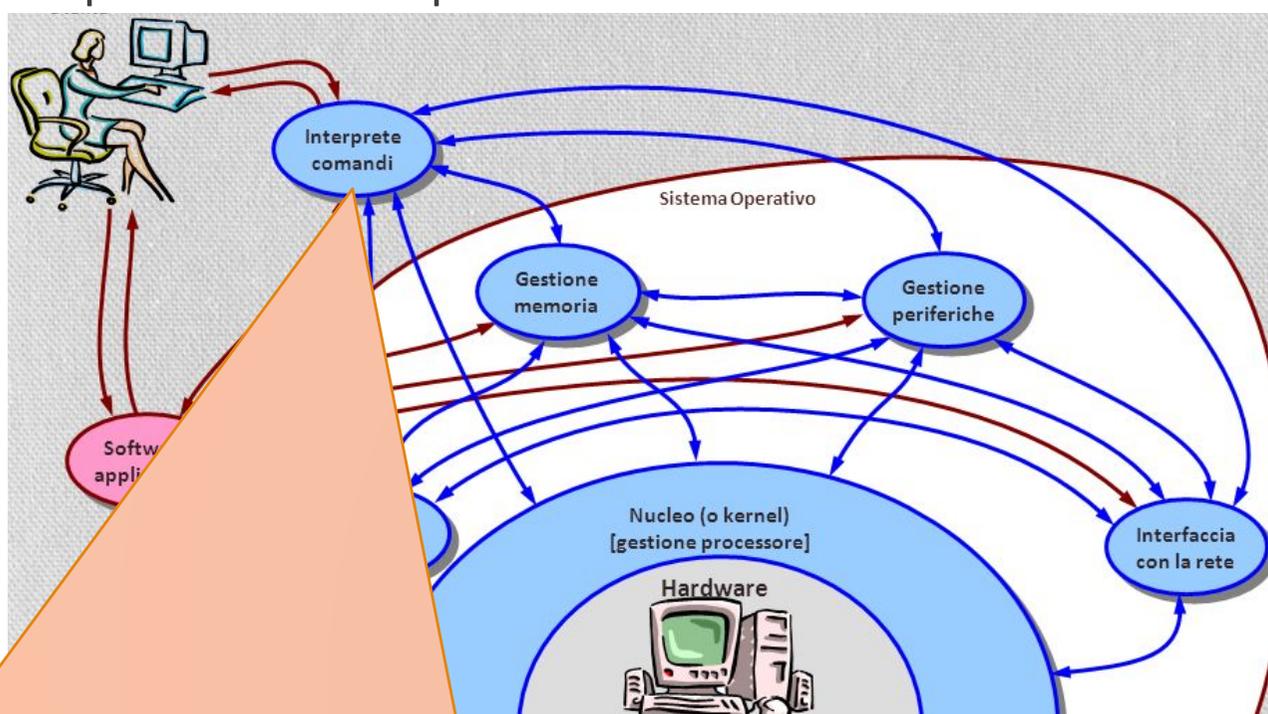
Un sistema operativo si compone di vari elementi.



Un sistema di gestione della rete che supporta l'implementazione dei principali protocolli di rete del modello ISO70SI e la gestione delle schede di rete.

# Sistemi Operativi – 3/9

Un sistema operativo si compone di vari elementi.



L'interprete dei comandi si interfaccia direttamente con l'utente, e permette loro di accedere in maniera semplice ed intuitiva (testuale o grafica) alle funzionalità offerte dagli altri elementi del sistema operativo.

# Sistemi Operativi – 4/9

---

L'insieme delle funzionalità offerte da un sistema operativo consentono di astrarre, o **virtualizzare**, l'hardware dell'elaboratore, consentendo a un utente di saper usare ogni possibile macchina indipendentemente dai suoi dettagli hardware. Non sono tutte comunemente messe a disposizione degli utenti e del software applicativo:

- in modalità utente, solo un sotto-insieme di funzionalità è impiegabile;
- in modalità supervisore o amministratori è possibile impiegare tutte le funzionalità.

I programmi applicativi non hanno diretto accesso alle risorse, ma richiedono al sistema operativo l'esecuzione di alcuni servizi. Il sistema operativo decide se e in che ordine soddisfare le richieste che gli giungono.

# Sistemi Operativi – 5/9

---

L'architettura di von Neumann è basata sul principio di esecuzione sequenziale delle istruzioni, spesso dobbiamo portare avanti dei programmi in parallelo. Il parallelismo a livello di dati e di istruzioni è possibile solo disponendo della ridondanza di CPU, come nelle attuali architetture. Il grado di tali ridondanza, però, non tiene il passo delle esigenze. Il sistema operativo è in grado di sopperire a tale mancanza, riducendo il tempo di inutilizzo delle risorse e di fornire l'illusione di un parallelismo mediante l'esecuzione concorrente dei programmi su una o più CPU.

I programmi in esecuzione, e i relativi dati o contesto, su una macchina prendono il nome di processi. Il sistema operativo simula il parallelismo tra processi, scegliendo di volta in volta il processo che prende possesso della CPU, ed avendo gli altri in una coda. Quando un processo ha terminato il tempo ad esse dedicato o deve attendere un evento, libera la CPU a beneficio di un altro processo ed entra nella coda.

# Sistemi Operativi – 6/9

---



Il problema dei filosofi a cena è un esempio, descritto nel 1965 da Edsger Dijkstra, che illustra un comune problema di controllo della concorrenza in informatica. Cinque filosofi siedono ad una tavola con un piatto di spaghetti davanti, e due forchette ai due lati. Ogni filosofo alterna momenti in cui riflette ed altri in cui deve mangiare con due forchette, prese una per volta. Il filosofo mangia per un po', poi lascia le forchette e ricomincia a pensare.

Il problema consiste nello sviluppo di un algoritmo che impedisca

- lo stallo (deadlock), ovvero ciascuno dei filosofi tiene in mano una forchetta senza mai riuscire a prendere l'altra;
- la morte d'inedia (starvation), ovvero se uno dei filosofi non riesce mai a prendere entrambe le forchette e non mangia mai.

La presa di forchette è analoga al blocco di risorse di un programma.

# Sistemi Operativi – 7/9

---

Per evitare, o eventualmente risolvere, tali problemi sono state pensate delle apposite tecniche che si avvalgono di strumenti messi a disposizione dal sistema operativo per la comunicazione interprocesso (IPC o InterProcess Communication) che permettono ai processi di alternarsi nell'accesso ad una risorsa condivisa.

Le più comuni di queste soluzioni sono

- Mutex: variabili binarie che permettono di gestire l'accesso ad una risorsa condivisa mediante accesso esclusivo di uno dei contendenti.
- Semafori n-ari: variabili n-arie che possono essere incrementate e decrementate. Il processo che decrementa il semaforo si bloccherà appena raggiunto lo zero della variabile. Un processo che incrementa il semaforo invece risveglia tutti i processi che si erano bloccati in fase di decremento.
- Primitive wait per porre in attesa un processo, e signal per risvegliare uno o più processi in attesa.

# Sistemi Operativi – 8/9

---

Una delle principali funzionalità di un sistema operativo è la gestione dei dati da memorizzazione in maniera persistente nella memoria di massa. Tale compito è svolto dal file system, che ha l'obiettivo di presentare all'utente il contenuto dell'hard disk e di effettuare opportune operazioni su di esso.

I dati in memoria sono organizzati in file, ovvero contenitori logici identificati mediante un nome. Tali nomi sono composti da un nome vero e proprio seguito dall'estensione, con un punto che li divide. L'estensione identifica il programma che ha generato il file (e/o il tipo del file). Ogni sistema operativo pone dei vincoli sulla lunghezza dei nomi dei file e i caratteri che possono essere impiegati al suo interno.

I file vengono suddivisi in più contenitori logici, dette cartelle o directory, organizzati secondo una struttura da albero, con una directory detta radice. Una directory può contenere al suo file oppure altre directory. Pertanto ogni file o directory è identificata da un percorso.

# Sistemi Operativi – 9/9

---

Il file system può avere un'interfaccia grafica, oppure una testuale per mezzo del prompt dei comandi. Ogni sistema operativo ha il suo insieme di comandi:

- **cd** – per accedere ad una cartella;
- **dir (ls in Linux)** – per visualizzare il contenuto di una directory;
- **md (mkdir in Linux)** - crea una directory;
- **move (mv in Linux)** – per spostare uno o più file in un'altra posizione;
- **rd (rm in Linux)** – per cancella una directory;
- **ren (mv in Linux)** – per rinominare uno o più file;
- **del (rm in Linux)** – per cancellare uno o più file.

# Riferimenti

---

- **Libro di testo**
  - Capitoli 5, 7 e 8