



## 2. Rete, Cloud, Server

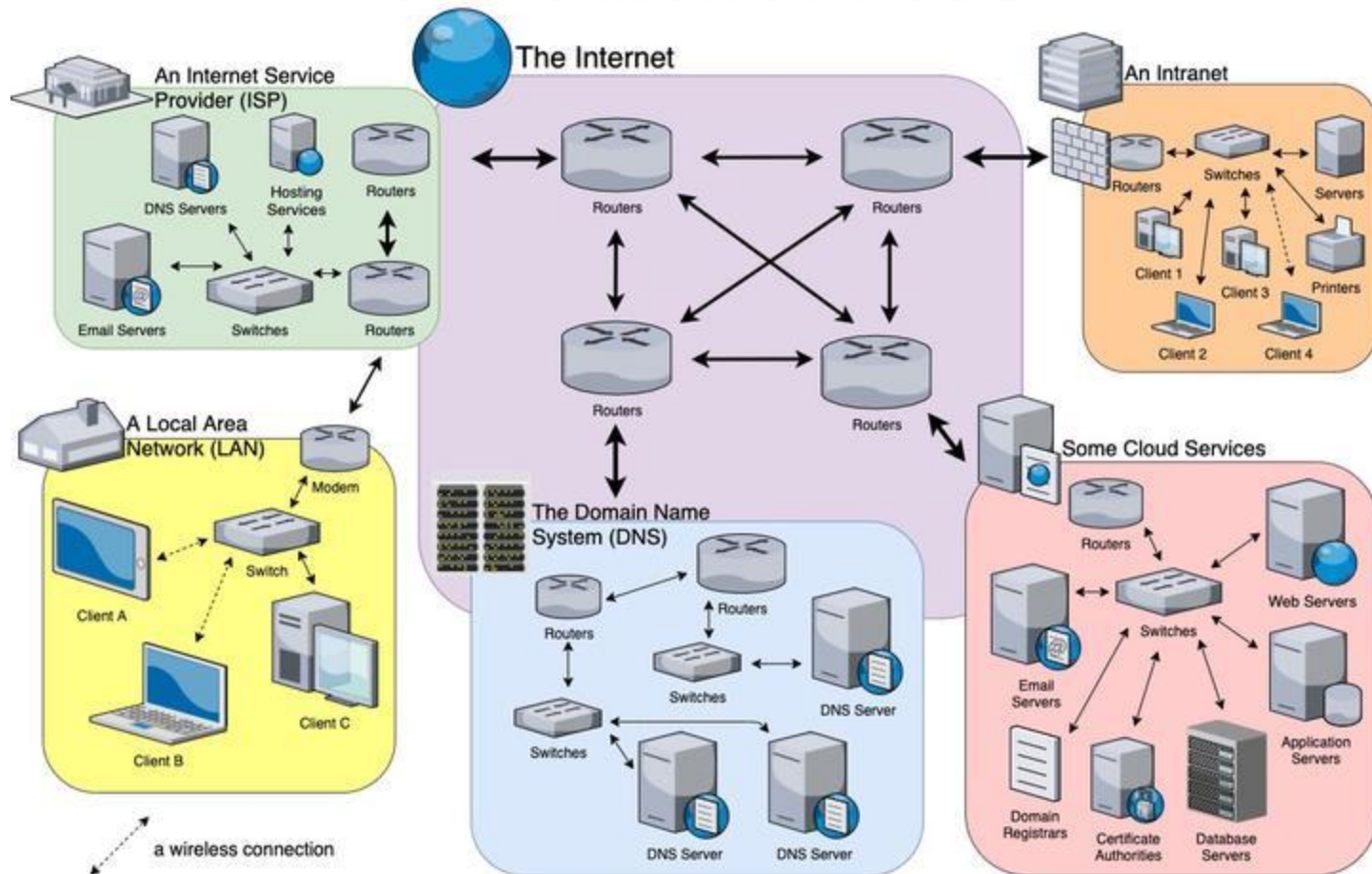
Informazioni generali



# Internet

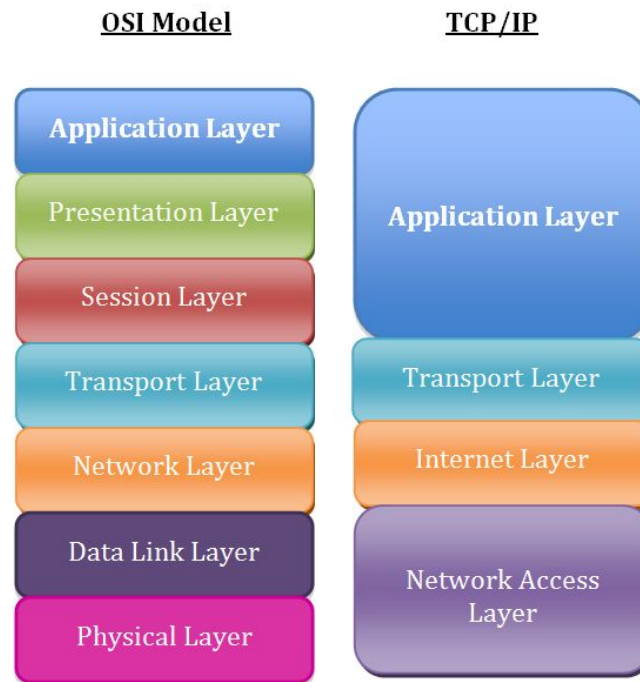
- Nata dalla fusione di reti di agenzie governative americane (ARPANET) e reti di università.
- È una rete di reti, di scala planetaria, pubblica, a commutazione di pacchetto.
- Sistema di comunicazione tra reti e sistemi eterogenei, oltre che geograficamente distribuiti.
- Utilizza protocolli di comunicazione di dominio pubblico derivati dal modello TCP/IP.
- Al giorno d'oggi è accessibile a tutti tramite provider che sono fisicamente collegati a internet.

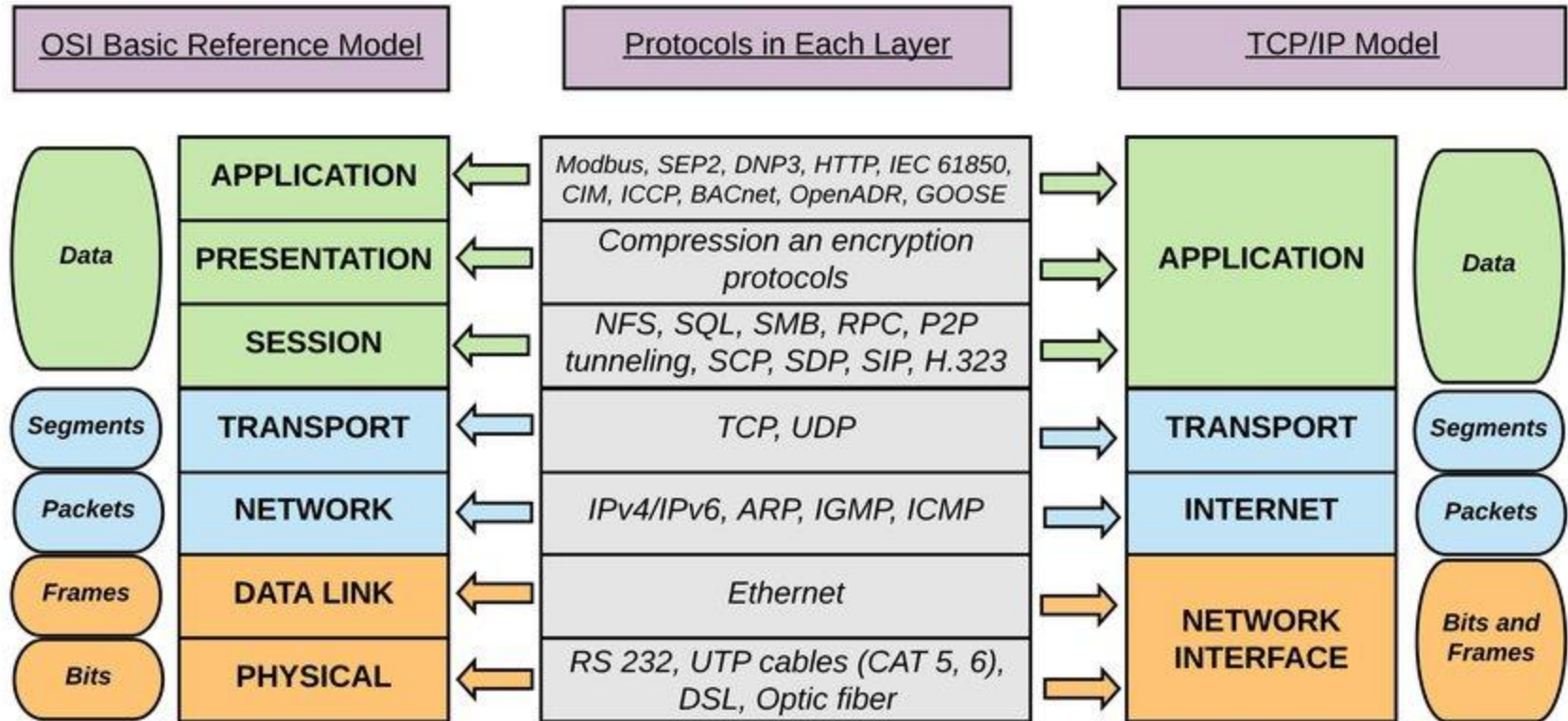
# The Internet Infrastructure: A bird's eye view



# TCP/IP

- Internet usa una suite di protocolli universale, TCP/IP, per far dialogare tra loro hardware e sistemi operativi diversi.
- Indipendenza del protocollo dal modo in cui è fisicamente organizzata la rete.
- Suddivisione in vari livelli.







# Protocolli applicativi

- HTTP
- HTTPS
- DNS
- SSH
- FTP
- SMTP
- IMAP
- POP-3



# Hypertext Transfer Protocol (HTTP)

L'HTTP è il fondamento del World Wide Web, e viene utilizzato per caricare pagine web utilizzando link ipertestuali. È un protocollo di livello applicativo progettato per trasferire informazioni tra dispositivi collegati in rete e funziona sopra altri livelli dello stack di protocollo di rete.

Un tipico flusso su HTTP coinvolge una macchina client che fa una richiesta ad un server, che poi invia un messaggio di risposta.

Richiesta HTTP di esempio con client CURL: `curl -v <url>`

Tipi di richieste:

- **Richiesta sicura:** Una richiesta che richiede solo dati e non modifica alcuna risorsa, è considerata un metodo sicuro.
- **Richiesta idempotente:** un'operazione idempotente è un'operazione che non ha alcun effetto aggiuntivo se viene chiamata più volte con gli stessi parametri di input.



# Richiesta HTTP

Una richiesta HTTP è il modo in cui le piattaforme di comunicazione internet come i browser web richiedono le informazioni necessarie per caricare un sito web.

Ogni richiesta HTTP effettuata attraverso Internet porta con sé una serie di dati codificati che trasportano diversi tipi di informazioni. Una tipica richiesta HTTP contiene:

- Tipo di versione HTTP
- un URL
- un metodo HTTP
- Headers di richiesta HTTP
- Body HTTP opzionale.





# Metodo HTTP

Un metodo HTTP indica l'azione che la richiesta HTTP si aspetta dal server interrogato. Per esempio, due dei metodi HTTP più comuni sono 'GET' e 'POST'.

Il metodo di richiesta, per la versione 1.1, può essere uno dei seguenti:

- **GET:** richiede una risorsa ad un server.
- **POST:** trasmettere informazioni dal client al server.
- **HEAD:** metodo è simile a GET, ma il server risponde inviando solo gli header relativi, senza il corpo.
- **PUT:** trasmettere delle informazioni dal client al server, sostituendo la risorsa specificata.
- **DELETE:** elimina la risorsa identificata dall'URL specificato.
- **PATCH:** modifica parziale di una risorsa.
- **TRACE:** esegue un test di loop-back del messaggio lungo il percorso della risorsa di destinazione.
- **OPTIONS:** richiede opzioni di comunicazione consentite per un determinato URL o server.
- **CONNECT:** apre una connessione per la comunicazione bidirezionale con una risorsa.



# Risposta HTTP

Una risposta HTTP è ciò che i client web ricevono da un server in risposta a una richiesta HTTP.

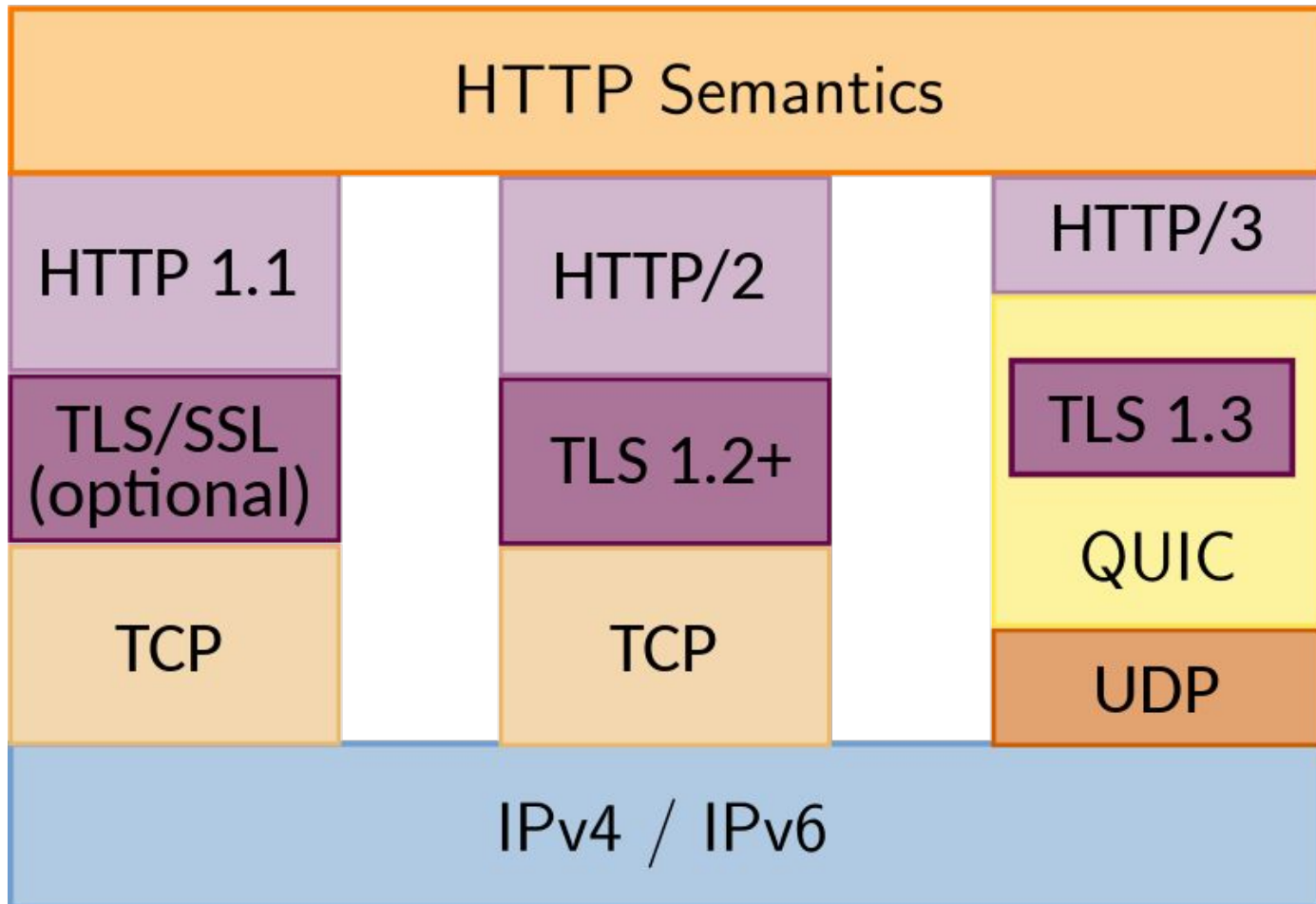
Una tipica risposta HTTP contiene:

- uno status code HTTP
- Headers di risposta HTTP
- Body HTTP opzionale

Codici di stato:

- 1xx Informativo
- 2xx Successo
- 3xx Reindirizzamento
- 4xx Errore del client
- 5xx Errore del server

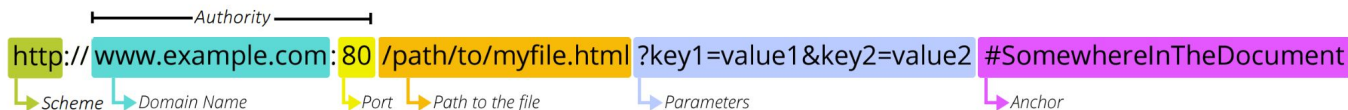
Parameter	http	https
Protocol	This is hypertext transfer protocol	This is hypertext transfer protocol with secure
Security	The data in http is vulnerable to hackers	Https is designed to be secure with the implementation of SSL certificate
Port	By default, http uses port 80	Https uses port 443 by default
Use	The http protocol is suitable for blogs where the platform facilitates the consumption of information	Any website that collects the personal information of users such as credit card data, personal information, etc. should use the https
Scrambling	Scrambling of data does not occur in http, the reason why its highly insecure	In https, data is scrambled before transmission and descrambled at the receivers end
Protocol	Http operates on TCP/IP	Https uses the same protocol but with TLS/SSL encrypted connection
Google Search	Http does not affect the search ranking	Https improves the search ranking
Speed	Faster than https	Slower due to SSL handshake
Domain Name Validation	No requirement of SSL certificate	SSL certification is a must
Data Encryption	Data is not encrypted	Data is encrypted
Vulnerability	Vulnerable to attacks	Secure by design



# Uniform Resource Locator (URL)

L'URL è un meccanismo di indirizzamento globale che consente di specificare in maniera univoca la risorsa alla quale il client WEB è interessato. Un URL indica il tipo di risorsa alla quale si sta accedendo (ex. http, ftp), il server che la possiede, il cammino per ritrovarla ed il suo nome.

Un nome di dominio è in realtà anch'esso costituito da parti più piccole. C'è il “www” (oggi facoltativo) che identifica il world wide web, il nome del sito in questione, e il dominio di primo livello (TLD), che si riferisce al designatore .it .com, .org, .net (tra gli altri) alla fine del nome di dominio.



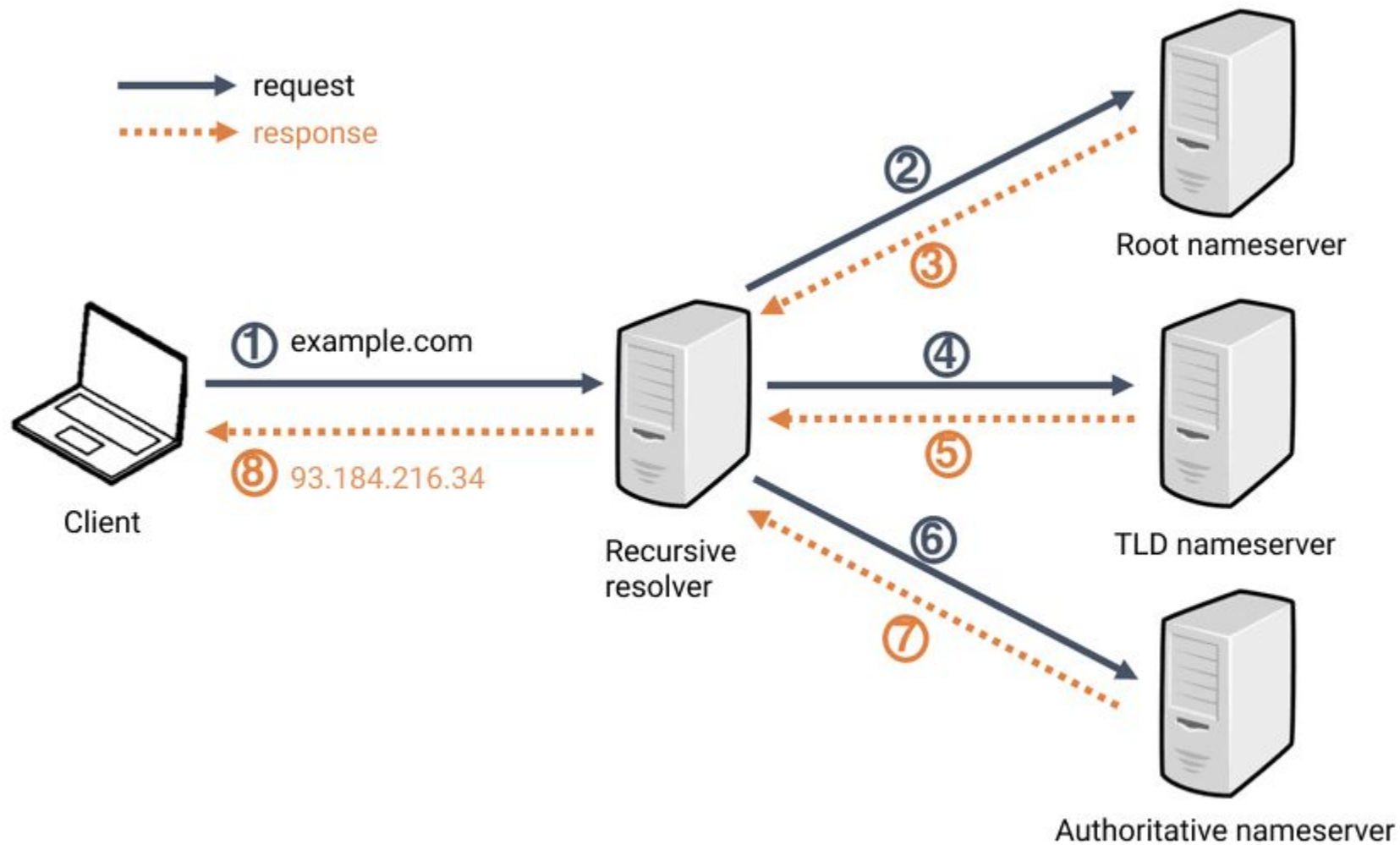


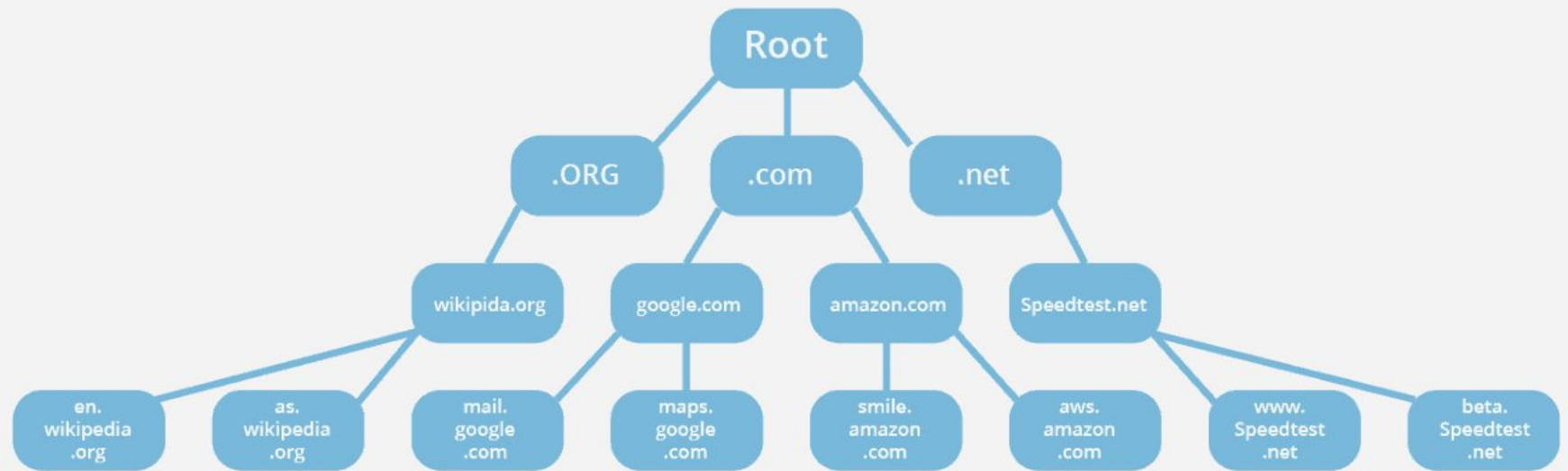
# Domain Name System (DNS)

Il Domain Name System (DNS) è la "guida telefonica" di Internet. Le persone accedono alle informazioni online tramite dei nomi di dominio, come ad esempio nytimes.com o espn.com. I browser Web interagiscono tramite indirizzi Internet Protocol (IP). Il DNS traduce i nomi di dominio in indirizzi IP, in modo che i browser possano caricare le risorse Internet.

Ogni dispositivo collegato a Internet è dotato di un indirizzo IP univoco, che altre macchine usano per trovarlo. I server DNS eliminano la necessità per gli esseri umani di memorizzare gli indirizzi IP, come 192.168.1.1 (in IPv4), o i più recenti e complessi indirizzi IP alfanumerici, come 2400:cb00:2048:1::c629:d7a2 (in IPv6).

Risoluzione di un dominio con DNS (DNS lookup): `host <dominio>`

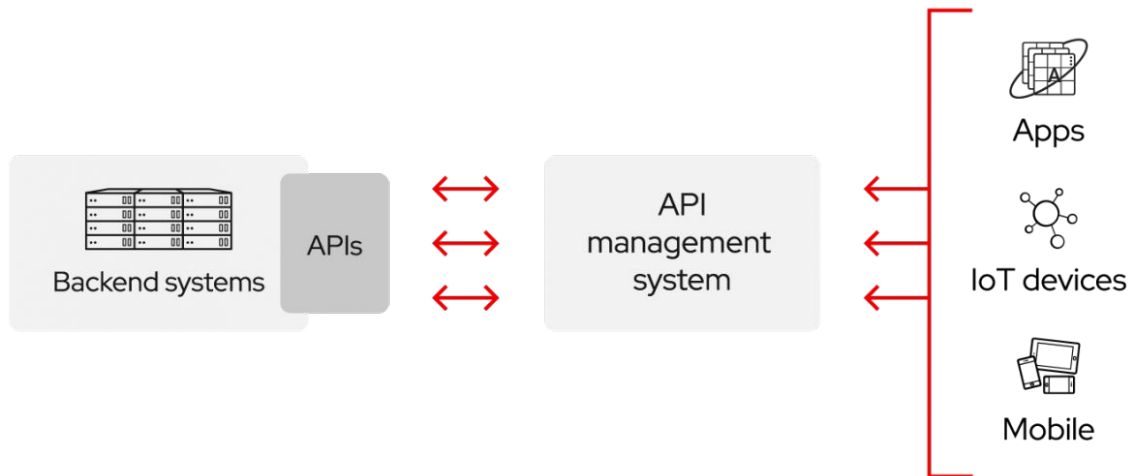






# Application programming interface (API)

Le API permettono ai tuoi prodotti o servizi di comunicare con altri prodotti o servizi, anche se non sai come sono stati implementati, semplificando così lo sviluppo delle app e consentendo un netto risparmio di tempo e denaro.





# JavaScript Object Notation (JSON)

JSON è un semplice formato per lo scambio di dati. Per le persone è facile da leggere e scrivere, mentre per le macchine risulta facile da generare e analizzarne la sintassi.

I tipi di dati supportati da questo formato sono:

- booleani (true e false);
- interi, numeri in virgola mobile;
- stringhe racchiuse da doppi apici ("");
- array (sequenze ordinate di valori, separati da virgole e racchiusi in parentesi quadre []);
- array associativi (sequenze coppie chiave-valore separate da virgole racchiuse in parentesi graffe);
- null.

```
1  {
2    "string": "Hi",
3    "number": 2.5,
4    "boolean": true,
5    "null": null,
6    "object": { "name": "Kyle", "age": 24 },
7    "array": ["Hello", 5, false, null, { "key": "value", "number": 6 }],
8    "arrayOfObjects": [
9      { "name": "Jerry", "age": 28 },
10     { "name": "Sally", "age": 26 }
11  ]
12 }
13
```

# Server

Un server (dall'inglese «serviente, servitore, cameriere») in informatica e telecomunicazioni è un dispositivo fisico o sistema informatico di elaborazione e gestione del traffico di informazioni. Un server fornisce, a livello logico e fisico, un qualunque tipo di **servizio** ad altre componenti (tipicamente chiamate **client**, cioè clienti) che ne fanno richiesta attraverso una **rete di computer**, all'interno di un sistema informatico o anche direttamente in **locale su un computer**.

Questa interazione tra server e client è chiamata **architettura client-server**.





# Database

In informatica, una base di dati o banca dati (in inglese database, comune anche in italiano) è una collezione di dati organizzati immagazzinata e accessibile per via elettronica.[1] Piccole banche dati possono essere immagazzinate su un file system, mentre banche dati più grandi sono ospitati su computer cluster o su cloud storage.

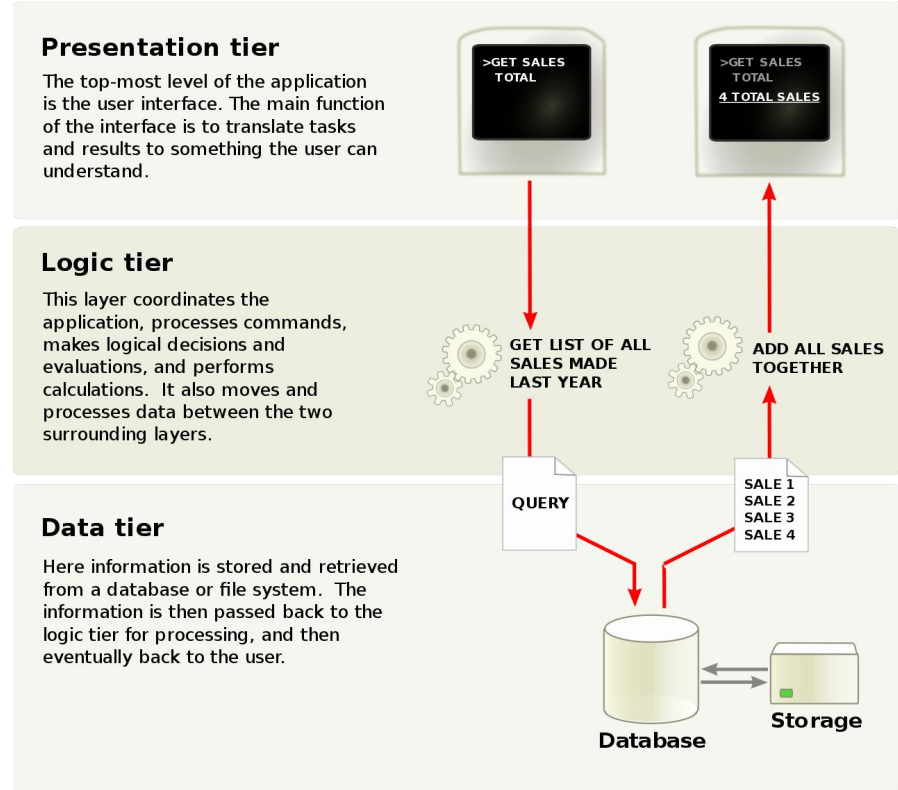
Un sistema di gestione delle banche dati, noto anche come **database management system (DBMS)**, è il software che interagisce con gli utenti finali, le applicazioni e la banca dati stessa per prendere e analizzare i dati. Inoltre, il software di gestione comprende le funzionalità di base necessarie per amministrare la banca dati. L'insieme della banca dati, il sistema di gestione e le applicazioni associate può essere chiamato un database system.

Tipi di DBMS:

- Database relazionali (RDBMS)
- Database orientato al documento
- Database a grafo
- Database a oggetti

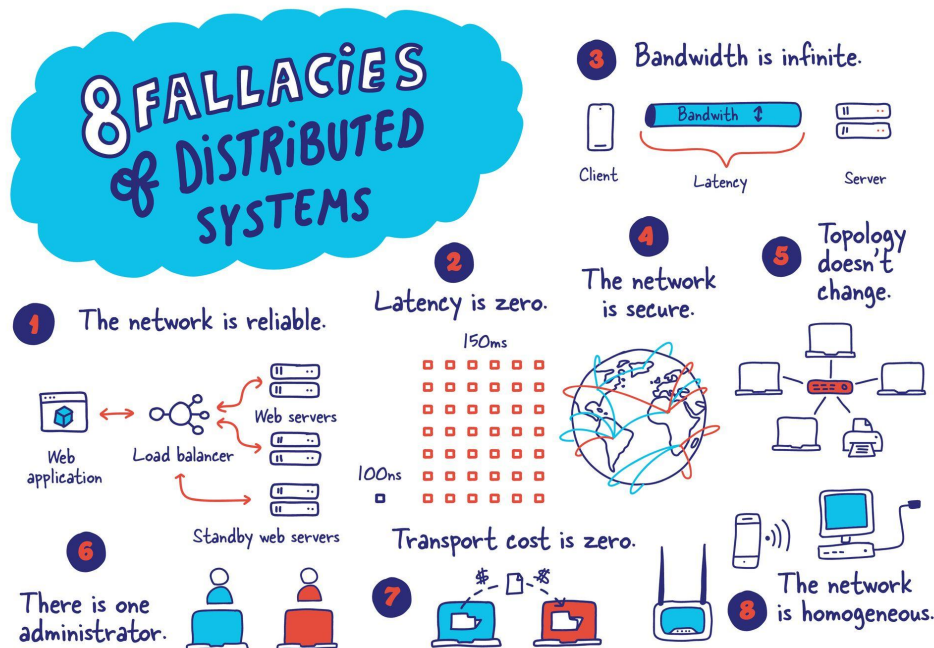
# Architettura three-tier

In ingegneria del software e nell'ambito dei sistemi informatici, l'espressione architettura three-tier ("a tre strati") indica una particolare architettura software e hardware di tipo multi-tier per l'esecuzione di un'applicazione web che prevede la suddivisione dell'applicazione in tre diversi moduli o strati dedicati rispettivamente alla **interfaccia utente**, alla **logica funzionale (business logic)** e alla **gestione dei dati persistenti**.



# Sistema distribuito

La locuzione sistema distribuito, in informatica, indica genericamente una tipologia di sistema informatico **costituito da un insieme di processi interconnessi tra loro** in cui le comunicazioni avvengono solo esclusivamente tramite lo scambio di opportuni messaggi. L'utente percepisce il sistema **come un'unica entità**. I sistemi distribuiti nascono da esigenze sia di tipo economico che tecnologico.





# Cloud

Il cloud computing indica un paradigma di erogazione di **servizi offerti su richiesta** da un fornitore a un utente finale attraverso la rete internet (come l'archiviazione, l'elaborazione o la trasmissione dati), a partire da un insieme di risorse preesistenti, configurabili e disponibili in remoto sotto forma di architettura distribuita.

L'architettura informatica del cloud computing prevede **uno o più server reali**, generalmente in architettura ad alta affidabilità (gruppi di server) e **fisicamente collocati presso il centro dati del fornitore** del servizio.

Il fornitore di servizi espone delle **interfacce per elencare e gestire i propri servizi**; il cliente amministratore utilizza tali interfacce per selezionare il servizio richiesto (per esempio un **server virtuale** completo oppure solo per archiviazione) e per amministrarlo (configurazione, attivazione, disattivazione).

Il cliente finale utilizza il servizio configurato dal cliente amministratore; le caratteristiche fisiche dell'implementazione (server reale, localizzazione del centro dati) sono irrilevanti.





# Tipi di servizi Cloud

- **SAAS (software as a service)** - consiste nell'utilizzo di programmi installati su un server remoto, cioè fuori del computer fisico o dalla LAN locale, spesso attraverso un server web;
- **DAAS (data as a service)** - con questo servizio vengono messi a disposizione via web solamente i dati, ai quali gli utenti possono accedere tramite qualsiasi applicazione, come se fossero residenti su un disco locale; in pratica il DAAS non è basato sul concetto di server ma di storage (conservazione dati);
- **HAAS (hardware as a service)** - con questo servizio l'utente invia dati a un computer, che vengono elaborati da computer messi a disposizione e restituiti all'utente iniziale.
- **PAAS (platform as a service)** - Invece che uno o più programmi singoli, viene eseguita in remoto una piattaforma software che può essere costituita da diversi servizi, programmi, librerie, ecc. Tale servizio è tipico di alcune piattaforme utilizzate per sviluppare altri programmi, quali **Amazon Web Services** o **Microsoft Azure** o **Oracle PaaS**.
- **IAAS (infrastructure as a service)** - oltre alle risorse virtuali in remoto, vengono messe a disposizione anche risorse hardware, quali server, capacità di rete, sistemi di memoria e archivio. La caratteristica dell'IAAS è che **le risorse vengono istanziate su richiesta o domanda al momento in cui una piattaforma ne ha bisogno.**



# Macchina virtuale

Una macchina virtuale è un **ambiente virtuale** creato utilizzando un apposito software che **emula il comportamento di una macchina fisica** (client o server). La creazione della macchina virtuale è possibile destinando temporaneamente ad essa **una parte delle risorse hardware** della macchina fisica (RAM, CPU, GPU, hard disk/SSD). Quando la macchina virtuale viene spenta, tutte le risorse vengono liberate e restituite alla macchina ospitante (**sistema "host"**) fatta eccezione per la totalità o una parte dei dati memorizzati negli **hard disk virtuali**.

Le informazioni relative al sistema operativo e ai programmi installati nella macchina virtuale (così come i dati in essa salvati) saranno conservate nelle unità di memorizzazione del sistema host, tipicamente in un unico file di grandi dimensioni.

L'**hypervisor** è il componente che si occupa di orchestrare il funzionamento delle macchine virtuali ed è pensabile come l'intermediario tra l'hardware del sistema host e le macchine virtuali.

Strumenti come VMware, Virtualbox, Xeon, Hyper-V e KVM sono alcuni esempi di soluzioni che mettono a disposizione hypervisor per la creazione e la gestione di macchine virtuali.



# Container

La principale differenza tra macchine virtuali e container è che questi ultimi **non virtualizzano l'intero hardware "sottostante"** bensì solamente il sistema operativo.

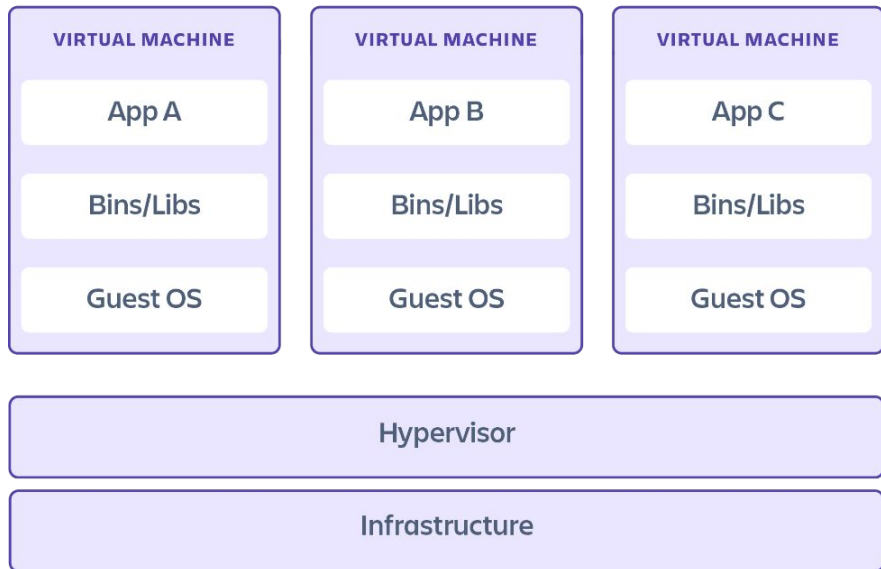
Ogni container **condivide l'utilizzo del kernel del sistema operativo** e, solitamente, anche librerie e binari, sia su Linux che su Windows.

I componenti del sistema operativo vengono condivisi in sola lettura e la condivisione di componenti software come le librerie riduce in maniera significativa la necessità di duplicare il codice del sistema operativo mantenendone in esecuzione più istanze in contemporanea.

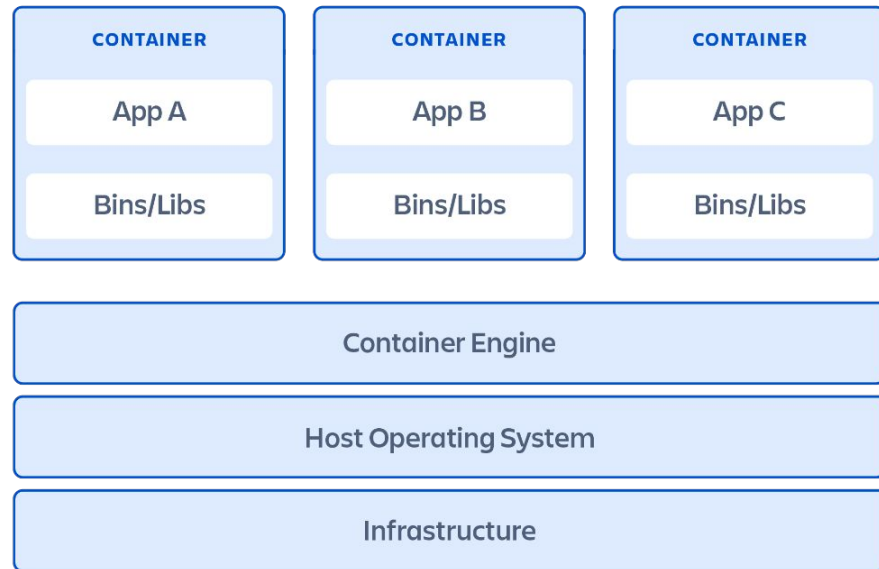
I container sono quindi **eccezionalmente leggeri**, pesano pochi megabyte e impiegano pochi secondi per avviarsi al contrario, evidentemente, delle macchine virtuali.

Diversamente dalle macchine virtuali, poi, il quantitativo di risorse che i container richiedono per il loro funzionamento è esattamente lo stesso del sistema operativo e delle applicazioni installate.

## Virtual machines

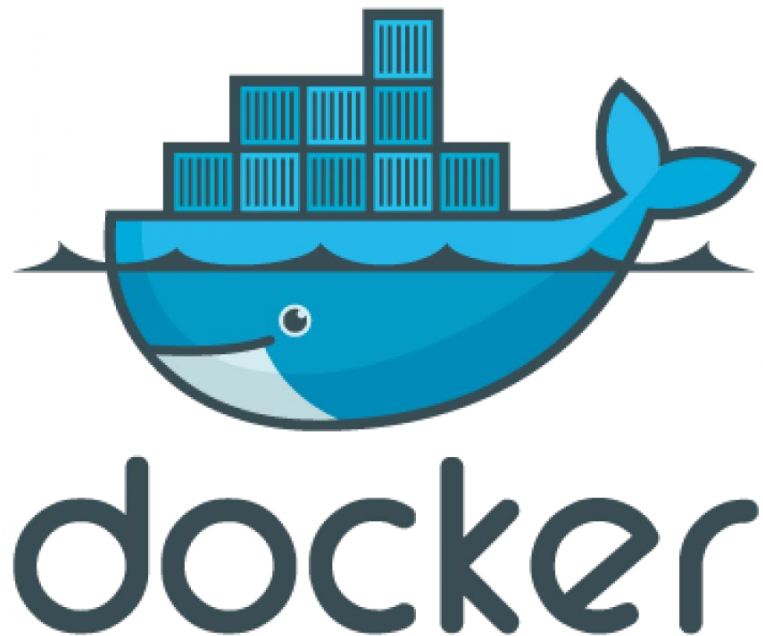


## Containers



# Docker

Docker è il runtime di container più popolare e più ampiamente utilizzato. Docker Hub è un enorme repository pubblico di popolari applicazioni software containerizzate. I container su Docker Hub possono essere scaricati e distribuiti istantaneamente in un runtime Docker locale.





# Applicazione web

Una applicazione web (web application in inglese, abbreviato **web app**), in informatica ed in particolare nella programmazione web, indica genericamente tutte le **applicazioni distribuite ovvero applicazioni accessibili/fruibili via web per mezzo di un network**, come ad esempio una Intranet all'interno di un sistema informatico o attraverso internet, cioè in una architettura tipica di tipo client-server, offrendo determinati **servizi all'utente client**.

Tipi di applicazioni web: e-commerce, motore di ricerca, social network, ecc.

Tecnologie utilizzate:

- HTML
- CSS
- JavaScript (JS)
- WebAssembly (Wasm)

```
{  
  message: "Grazie per l'attenzione!",  
  lecture: 2.0  
}
```

