INFORMATICA

Dispensa Riassuntiva

5LSOSA

A cura del Prof. Greguoldo Andrea

PROGRAMMAZIONE

Un programma viene sviluppato ***scrivendo il codice sorgente***

Linguaggio Compilato
Linguaggi come il C, C++, Delphi, Visual Basic sono **linguaggi compilati** e seguono questi passi: si scrive il codice sorgente in un editor che viene poi controllato per verificare che non ci siano errori e poi viene compilato, ovvero ogni istruzione viene trasformata nel corrispondente codice in linguaggio macchina che può essere, così, eseguito dal processore

Linguaggio Interpretato
I linguaggi interpretati, invece, seguono una strada diversa, il codice sorgente viene interpretato al volo e vengono, quindi, eseguite le istruzioni così come descritte nel codice sorgente; un esempio su tutti è il PHP il cui codice viene "elaborato" e restituisce una pagina html pura.

Un **linguaggio che è a metà strada tra queste metodologie è Java** che è sia compilato che interpretato; il codice sorgente viene compilato in un formato intermedio (chiamato ***bytecode***), il quale a sua volta viene interpretato dalla Java Virtual Machine (JVM), che ha il compito di interpretare "al volo" le istruzioni bytecode in istruzioni per il processore; la JVM viene sviluppata per ogni Sistema Operativo che rende, in pratica, JAVA altamente portabile.

Si dice ***algoritmo*** la descrizione di un metodo di soluzione di un problema che:

- sia eseguibile
- sia priva di ambiguità
- arrivi ad una conclusione in un tempo finito

Un computer può risolvere soltanto quei problemi per i quali sia noto un algoritmo.

PROGRAMMAZIONE

Cos'è una variabile?
Nei linguaggi di programmazione le variabili sono spazi di memoria usati per memorizzare dei valori numerici, alfanumerici o booleani.

La dichiarazione delle variabili
La dichiarazione di una variabile consiste nel definire il nome e la tipologia della variabile. Ad esempio, nelle seguenti righe dichiaro una variabile alfanumerica (String), due variabili numeriche (int e double) e una variabile booleana (boolean).

```
String nome;
int anno;
double euro;
boolean privacy;
```

L'assegnazione delle variabili
Dopo aver dichiarato una variabile, posso assegnargli un valore.
Ovviamente il valore deve essere dello stesso tipo rispetto alla dichiarazione. Ad esempio, se la variabile è stata dichiarata come intera (int) posso assegnarli solo numeri interi.

```
nome=«Andrea»;
anno=2022;
euro=15.45;
privacy=false;
```

Dichiarazione e assegnazione delle variabili
In alternativa, potrei assegnare il valore iniziale alle variabili già in fase di dichiarazione.

```
String nome=«Andrea»;
int anno=2022;
double euro=15.45;
boolean privacy=false;
```

Grit

PROGRAMMAZIONE

Il modo più semplice e diretto per creare un oggetto di tipo String è assegnare alla variabile un insieme di caratteri racchiusi fra virgolette

```
String titolo = "Lezione sulle stringhe";
```

questo è possibile in quanto il compilatore crea una variabile di tipo String ogni volta che incontra una sequenza racchiusa fra doppi apici; nell'esempio la stringa "Lezione sulle stringhe" viene trasformata in un oggetto String e assegnato alla variabile titolo. La classe String espone anche numerosi metodi per l'accesso alle proprietà della stringa sulla quale stiamo lavorando; uno di questi è il metodo ***length()***, che ritorna il numero di caratteri contenuti nell'oggetto.

```
String descrizione = "Articolo sulle stringhe ...";
int numeroCaratteri = descrizione.length();
System.out.println("Lunghezza: "+ numeroCaratteri);
```

→ stampano come risultato:

```
Lunghezza: 27
```

In questo esempio è interessante notare anche come il compilatore interpreti automaticamente l'espressione *"Lunghezza: "+numeroCaratteri*, creando un oggetto di tipo String ottenuto concatenando la stringa che troviamo in prima posizione e la stringa ottenuta dalla rappresentazione decimale del valore di *numeroCaratteri* (variabile di tipo int).

Per prelevare e manipolare solo una porzione di una stringa possiamo utilizzare il metodo ***subString***

```
String titolo = "I promessi Sposi";
String a = titolo.substring(2);   // a vale "promessi Sposi«
String b = titolo.substring(11);  // b vale "Sposi«
String c = titolo.substring(2,10); // c vale "promessi"
```

Nota: sia l'operazione di concatenamento sia quella di estrazione di una sottostringa (e tutti i metodi che operano sulle stringhe per la verità), sono caratterizzati dal fatto di non modificare la stringa su cui vengono applicate ma di ritornarne una nuova. Ad esempio titolo.substring(12) non modifica titolo ma ritorna una nuova variabile di tipo String che contiene la sottostringa "Sposi";

Come concatenare due o più Stringhe:

```
String str1 = "Nome";
String str2 = "Cognome";
String str3 = str1+str2;
```

str3 conterrà: NomeCognome

Come confrontare due Stringhe:

```
String str1 = «Anna";
String str2 = «Paola";
if (str1.equals(str2))
    {System.out.println(«le due Stringhe sono uguali»);}
```

Estrarre un carattere da una Stringa:

```
String str1 = «Paola";
char ch = str1.charAt(2);
```

ch conterrà: «o»

Grit

PROGRAMMAZIONE

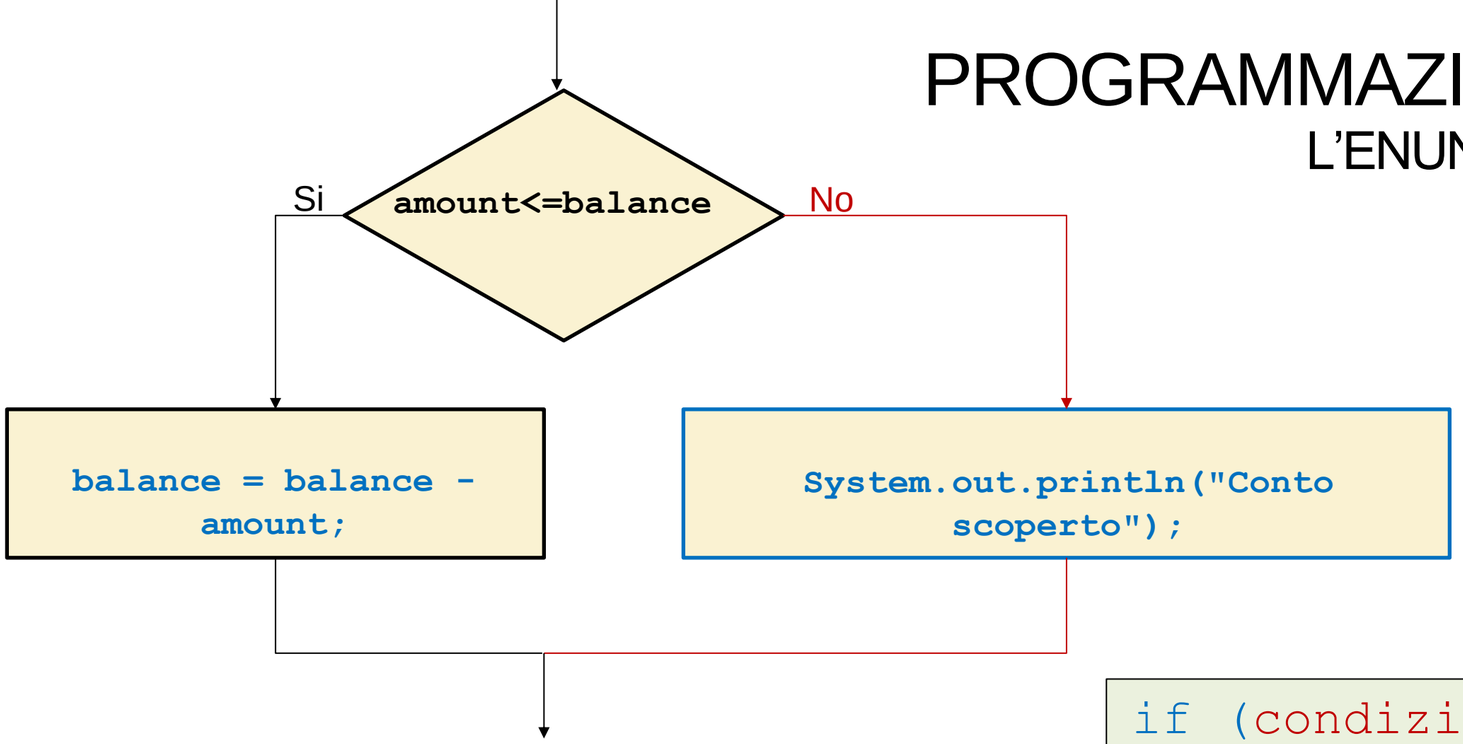
Tipi di variabili in Java

Le variabili locali in Java possono essere:

- ***int*** - Numeri interi a 32 bit. Fino a +/-2 miliardi circa.
- ***short*** - Numeri interi a 16 bit. Fino a +/-32 mila circa.
- ***byte*** - Numeri interi fino a 8 bit. Da -128 a +127.
- ***long*** - Numeri interi a 64 bit. Fino a +/- 9223 milioni di miliardi
- ***float*** - Numeri in virgola mobile a 32 bit.
- ***double*** - Numeri in virgola mobile a 64 bit
- ***boolean*** - Valore booleano vero (true) o falso (false).
- ***char*** - Un carattere singolo.
- ***String*** - Una stringa di caratteri.

PROGRAMMAZIONE

L'ENUNCIATO IF-ELSE



Sintassi:

```
if (condizione)
   enunciato1;
```

```
if (condizione)
   enunciato1;
else
   enunciato2;
```

- ✓ Scopo: eseguire ***enunciato1*** se e solo se la ***condizione*** è **vera**; se è presente la clausola **else**, eseguire ***enunciato2*** se e solo se la ***condizione*** è **falsa**
- ✓ Spesso il corpo di un enunciato **if** o **else** è costituito da ***più enunciati*** da eseguire ***in sequenza***; racchiudendo tali enunciati tra una coppia di parentesi graffe **{ }** si crea un ***blocco di enunciati***, che può essere usato come corpo

```
if (amount <= balance)
   {balance = balance - amount;
   System.out.println("Prelievo accordato");}
else
   {System.out.println("Prelievo non accordato");}
```

Grit

PROGRAMMAZIONE

```
day=console.nextInt();

if (day == 1)
      giorno = "Monday";
   else if (day == 2)
      giorno = "Tuesday";
   else if (day == 3)
      giorno = "Wednesday";
   else if (day == 4)
      giorno = "Thursday";
   else if (day == 5)
      giorno = "Friday";
   else if (day == 6 || day == 7)
      giorno = "Week-end";
   else
      giorno = "errore";
```

Prendiamo ad esempio il codice qui a fianco.
Una serie di if-else per stabilire il giorno della settimana in seguito ad un inserimento da tastiera dell'utente.

Potremmo usare un altro modo, forse, più semplice: l'enunciato switch

Ecco l'equivalente codice tramite l'uso dell'enunciato switch:

```
day=console.nextInt();

switch ( day ) {
   case 1: giorno = "Monday"; break;
   case 2: giorno = "Tuesday"; break;
   case 3: giorno = "Wednesday"; break;
   case 4: giorno = "Thursday"; break;
   case 5: giorno = "Friday"; break;
   case 6: giorno = "Week-end"; break;
   case 7: giorno = "Week-end"; break;
   default: giorno = "errore"; // Tutti gli altri casi.
}
```

Sintassi:

```
switch ( <selettore> )
{           case <valore1> : <istruzioni1> break;
            case <valore2> : <istruzioni2> break;
            ...
            case <valoreX> :
            case <valoreY> : <istruzioniY> break;
             ...
            default: <istruzioniN> break;
}
```

Attenzione: l'enunciato switch si può usare solo in circostanze limitate. I valori con i quali si può confrontare il selettore devono essere costanti di tipo *intero*, *char* o *enumerativo*: non si può usare lo switch ad esempio con un selettore di tipo *double* o *String*.
Anche quando è lecito, non sono molti i casi in cui è opportuno che l'istruzione switch sostituisca l'istruzione if-else.

Grit

PROGRAMMAZIONE

L'**iterazione**, chiamata più comunemente **ciclo** è una struttura di controllo, all'interno di un algoritmo risolutivo di un problema dato, che ordina all'elaboratore di eseguire ripetutamente una sequenza di istruzioni, solitamente fino al verificarsi di particolari condizioni logiche specificate. In Java, come in altri linguaggi di programmazione esistono tre tipi di cicli: FOR, WHILE e DO-WHILE.

FOR

Il ciclo *FOR* è paragonabile al *WHILE* ma contiene una condizione implicita che deriva dal fatto che in questo tipo di ciclo il numero di volte che si dovrà eseguire il blocco di istruzioni è noto sin dall'inizio

```
for (inizializzazione; condizione; aggiornamento)
      { enunciati }
```

WHILE

Il ciclo WHILE controlla la condizione e successivamente, se la condizione lo permette, esegue il blocco di istruzioni che contiene, che può anche non essere eseguito nemmeno una volta (ciò avviene nel caso in cui la condizione risulti falsa già al primo controllo).

```
int i = inizializzazione;
while (condizione)
      { enunciati;
      aggiornamento;}
```

DO-WHILE

Il ciclo DO-WHILE termina quando la condizione è falsa, e, siccome la condizione viene controllata dopo aver eseguito il blocco di istruzioni, esegue sempre almeno una volta il ciclo.

```
do
      { enunciati;
      …….
      } while (condizione)
```

Ricordiamo che tra loro i cicli sono intercambiabili, cioè, con i giusti accorgimenti, posso indistintamente utilizzare un ciclo al posto di un altro; però annotiamo le peculiarità di ciascuno:
FOR: di solito si usa quando si sa esattamente il numero di loop che dovrà eseguire il nostro codice.
WHILE: di solito si usa quando non si sa esattamente il numero di loop che dovrà eseguire il nostro codice, potrebbe anche non venire eseguito (dipende dalla condizione).
DO-WHILE: di solito si usa quando non si sa esattamente il numero di loop che dovrà eseguire il nostro codice, ma sappiamo che almeno una volta verrà eseguito.

N.B. per uscire forzatamente da un ciclo si usa il comando «break».

Grit

PROGRAMMAZIONE

Un **array in Java** è un contenitore che permette di gestire una sequenza di lunghezza fissa di elementi tutti del medesimo tipo. Il numero di elementi in un array, detto *lunghezza dell'array*, deve essere dichiarato al momento della sua *allocazione* e non può essere cambiato.

La sintassi per la dichiarazione di una variabile di tipo array è la seguente, nella quale si deve osservare l'uso delle parentesi quadre '[ ]' tra il tipo ed il nome della variabile, per quanto riguarda inizializzazione dovremo allocare la memoria per mezzo della consueta keyword ***new*** che riserva la memoria necessaria per contenere ***n*** elementi di tipo ***Tipo***.

```
Tipo [ ] nome = new Tipo[n];
```

Una volta creato l'array, possiamo **accedere ai singoli elementi** indicandone la posizione (detta *indice*) all'interno contenitore, grazie all'operatore '[ ]'.
Ad esempio:

```
nome [3] = "tipo di dato da inserire";  // indica il quarto elemento della collezione
```

L'inizializzazione degli elementi dell'array è prolissa, poco leggibile e decisamente scomoda da collocare nel codice (soprattutto se vogliamo dichiarare numeroGiorniPerMese come static, provate per fare pratica !!).
Java fortunatamente ammette una sintassi per allocare ed inizializzare gli array in modo più diretto:

```
int [ ] numeroGiorniPerMese = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};
```

UNITÀ DI MISURA INFORMATICA

1bit → La più piccola unità di misura dello spazio in informatica

8bit = 1Byte

1024 Byte = 1 KByte

1024 KByte = 1 MByte

1024 MByte = 1 GByte

1024 GByte = 1 TByte

1024 TByte = 1 PByte (Peta Byte)

1024 PByte = 1 Ebyte (Exa Byte)

La velocità di trasferimento dei dati invece si misura in «*bit per secondo*» (*bps*) o suoi multipli.

80 Mbps	=	10 MByte al secondo
80Megabitper secondo	=	10 MegaByte al secondo

Grit

INTERNET – WEB – RETI

INTERNET - WEB

La nascita di internet

La rete internet nacque all'inizio degli anni '70. In origine si chiamava Arpanet ed era un progetto militare per mettere in comunicazione gli elaboratori elettronici degli enti governativi e militari. Successivamente, nel corso degli anni '80, la rete Internet uscì dagli ambiti militari e iniziò a essere utilizzata nelle università. Alla rete Internet si collegarono altre reti locali anche al di fuori del territorio americano. Per questa ragioni internet è anche conosciuta come Rete delle reti. In questi anni nacquero e si diffusero alcuni protocolli come quelli per gestire la posta elettronica (email) e i newsgroup. Per leggere le informazioni su un server o per scaricare un file si utilizzava la riga comandi e alcuni software di visualizzazione testuale come Telnet.

La nascita del web

Il web nacque nel 1991, circa venti anni dopo la nascita della rete Internet. Il termine Web è l'abbreviazione di World Wide Web (WWW). Si tratta di un sistema di comunicazione basato sul protocollo di comunicazione HTTP (Hyper Text Transfer Protocol) che consente agli utenti di navigare in modo ipertestuale tra i contenuti presenti sulla rete internet. Scompare la riga comandi. Per spostarsi da una pagina all'altra gli utenti possono cliccare sui collegamenti ipertestuali tramite un apposito software client detto browser.

In conclusione

Per riassumere il web non è internet ma soltanto uno dei servizi che lo compongono, al pari della posta elettronica o del trasferimento file via FTP. Il termine Web è spesso associato erroneamente a Internet, talvolta usato come sinonimo, soltanto perché è uno dei servizi più utilizzati dagli utenti.

Il termine INTERNET indica l'infrastruttura di rete, alla quale sono collegate tutte le reti informatiche mondiali, mentre il web è un servizio che si appoggia a Internet.

Se "cade" Internet non funziona il Web, mentre ne caso contrario internet continuerà a funzionare.

Spesso si parla di internet e web come se fossero la stessa cosa. In realtà, non è così. C'è una differenza notevole tra i due concetti.

1. La rete internet è l'infrastruttura tecnologica dove viaggiano i dati. Può essere immaginata come una specie di ferrovia digitale, con i propri binari (canali), stazioni (server) e regole (protocolli).
2. Il web è uno dei servizi internet che permette il trasferimento e la visualizzazione dei dati, sotto forma di ipertesto. Tutto ciò che normalmente vediamo

Esempio. Quando visualizzo una pagina web sul browser sto utilizzando il Web. Quando scarico la posta elettronica sul mio computer, invece, non utilizzo il web. In entrambi i casi uso la rete internet per trasferire i dati ma i protocolli sono differenti.

Grit

COMPONENTI FONDAMENTALI DI UNA RETE

SCHEDA DI RETE

è un'interfaccia digitale, costituita da una scheda elettronica, che svolge tutte le funzionalità logiche di elaborazione necessarie a consentire la connessione del dispositivo ad una rete informatica

MEZZO TRASMISSIVO

il canale a livello fisico entro il quale viaggiano i segnali rappresentativi dell'informazione

PROTOCOLLO

insieme di regole utilizzate dalle due macchine per scambiarsi informazioni e specifica cosa deve essere comunicato, in che modo e quando

LE RETI - TIPOLOGIA

- **PAN: P**ersonal **A**rea **N**etwork (rete personale) - collega strumenti personali quali cellulare, cuffie, orologio, generalmente collegati via Bluetooth

- **LAN**: **L**ocal **A**rea **N**etwork (rete locale) - collega nodi relativamente vicini, tipicamente nell'ambito di uno stesso edificio

- **CAN: C**ampus **A**rea **N**etwork una CAN è conosciuta anche come *corporate area network*. Una CAN è più grande di una LAN ma più piccola di una MAN. Le CAN servono siti come college, università e campus aziendali.

- **MAN**: **M**etropolitan **A**rea **N**etwork (rete metropolitana) - gestisce collegamenti a distanze massime di decine di chilometri

- **WAN**: **W**ide **A**rea **N**etwork (rete geografica) - collega nodi a distanza qualsiasi, anche planetaria

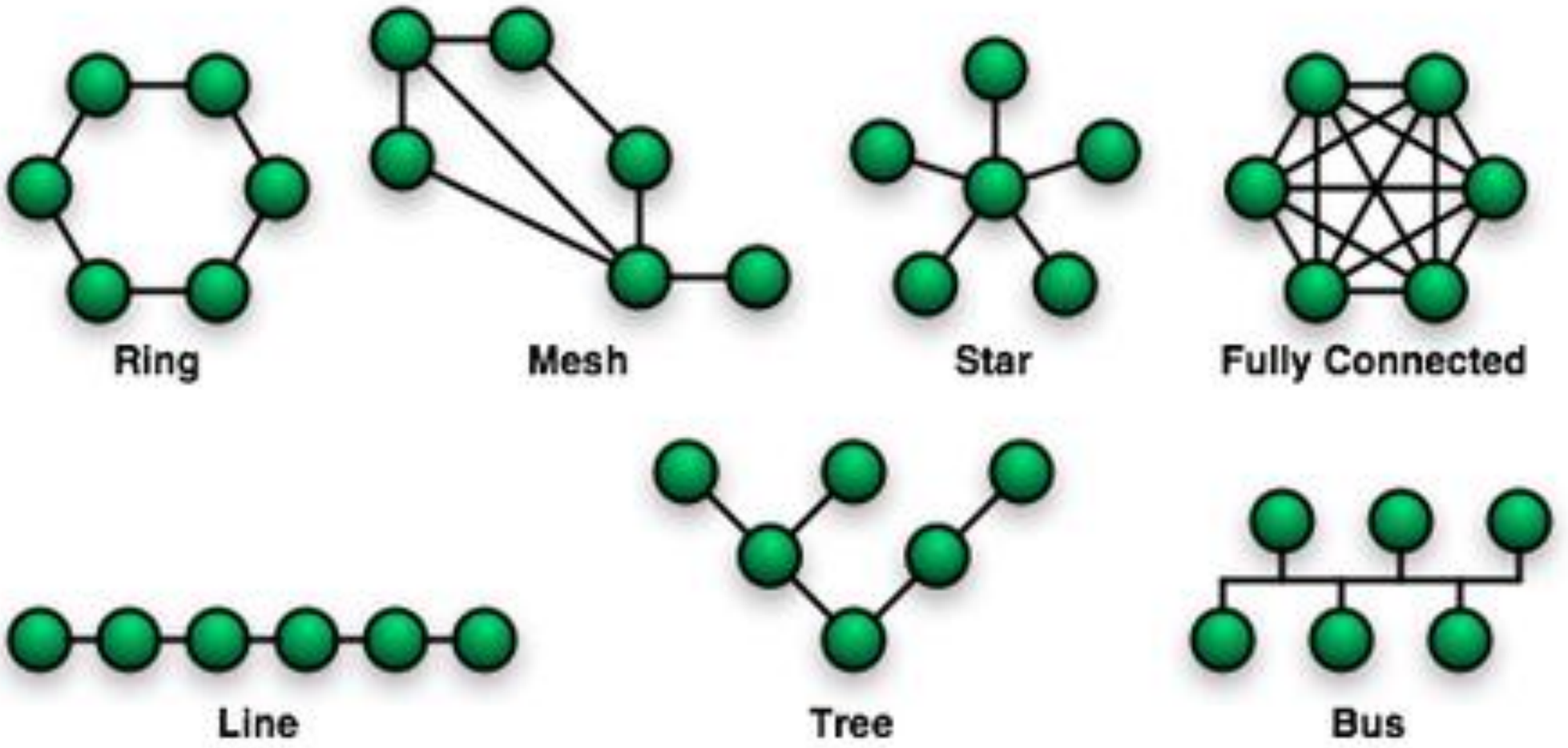
---

- **VPN: V**irtual **P**rivate **N**etwork una VPN è una connessione sicura, point-to-point tra due punti finali di rete. Una VPN stabilisce un canale crittografato che mantiene le credenziali di identità e di accesso di un utente, nonché qualsiasi dato trasferito, inaccessibile agli hacker.

Grit

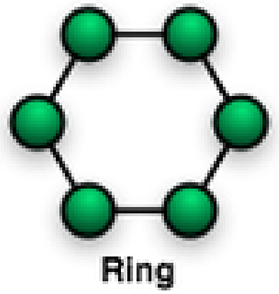
LE RETI - TOPOLOGIA

In telecomunicazioni la **topologia di rete** è il modello geometrico (grafo) finalizzato a rappresentare le relazioni di connettività, fisica o logica, tra gli elementi costituenti la rete stessa (detti anche *nodi*).

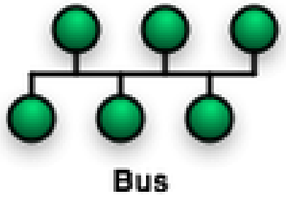


Grit

LE RETI - TOPOLOGIA



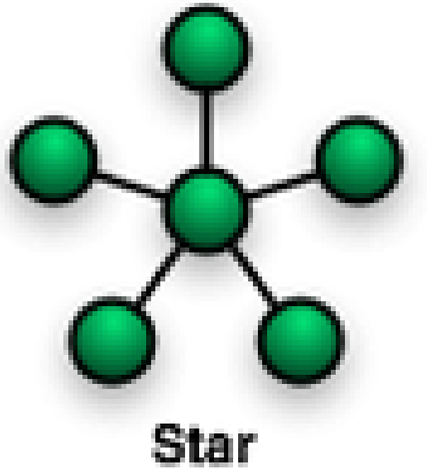
- **AD ANELLO**: La rete ad anello è caratterizzata da dispositivi concatenati uno all'altro, senza interruzioni, secondo una geometria circolare. I dati viaggiano intorno al cerchio (unidirezionale o bidirezionale) e passano attraverso tutti i nodi.
Ciascun nodo esamina il pacchetto che riceve per decidere se è deve acquisirlo o passarlo a sua volta. Passando da un computer all'altro, il segnale dei dati ricevuti e la trasmissione termina quando il pacchetto fa un intero giro e ritorna al nodo trasmittente.
Svantaggi: Scarsa scalabilità: l'aggiunta o la rimozione di un nodo presuppone l'apertura dell'intero anello e inoltre, a seconda delle tecnologie trasmissive e dei protocolli trasmissivi, potrebbe esserci un limite al numero massimo di nodi utilizzabili. per quanto riguarda l'anello unidirezionale se si verifica un problema a un singolo cavo o nodo l'intera rete si blocca



- **A BUS**: Consiste in un singolo cavo che connette in maniera lineare tutti i computers. I dati sono inviati a tutti i calcolatori e vengono accettati solo da quello il cui indirizzo coincide con quello contenuto nel segnale di origine.

- **A LINEA APERTA**: Ogni nodo è collegato con un ramo al nodo adiacente precedente e con l'altro ramo al nodo adiacente successivo. I nodi terminali sono invece adiacenti a un solo nodo. La comunicazione tra due nodi non adiacenti deve attraversare tutti i nodi intermedi, percorrendo i rami relativi: ogni passaggio tra due nodi viene detto salto o hop.
Svantaggi: scarsissima affidabilità: in una rete a Bus se un nodo si guasta o un ramo si interrompe, la rete viene divisa in due sottoreti isolate. Velocità contenuta dovuta al fatto che si utilizza un unico cavo per la trasmissione dei dati, per cui può inviare i dati un solo computer alla volta e, se ne sono tanti, si allungano i tempi di trasmissione dei dati.

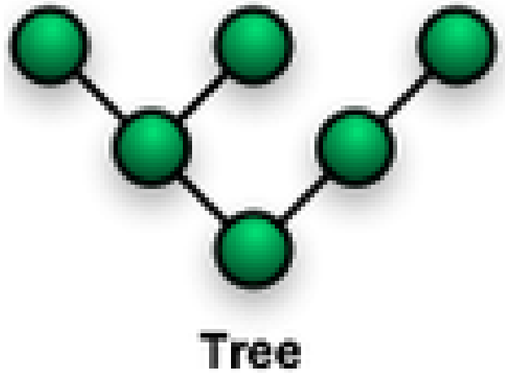
Grit



- **A STELLA**: Nelle reti a stella tutti gli apparecchi sono collegati ad un’unica unità centrale di connessione. Le reti a stella sono la topologia più utilizzata delle reti odierne

Svantaggi: Nella topologia a stella, il guasto dell’hub/switch/server/router comporta la perdita totale della funzionalità di rete, risultando di fatto isolati tutti i nodi componenti.

Lo schema a stella offre però una elevata scalabilità: è possibile aggiungere o togliere nodi e connessioni senza modificare la rete né la sua funzionalità, fino al numero massimo di diramazioni consentite dal nodo padre, inoltre, è molto facile l'accorpamento di più reti in un'unica rete, collegando direttamente tra loro i relativi nodi radice, senza che questo abbia ripercussioni sulle reti preesistenti.



- **AD A L B E R O** : È una variante più complessa di una struttura lineare, caratterizzata dal fatto che da ciascun nodo possono dipartirsi più catene lineari distinte e non intersecantesi, realizzando così una struttura multilivello. Ha un elevato grado di affidabilità.

Svantaggi: l'unico punto debole è costituito dai nodi padre, che, in caso di guasto, rendono impossibile l'accesso alla sottorete che si diparte da essi e che rimane quindi isolata.

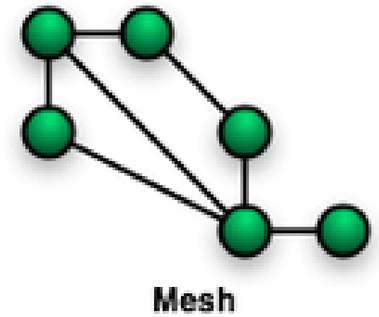
Grit

LE RETI - TOPOLOGIA

- **A MAGLIA**: La topologia completamente magliata o a maglia completamente connessa è quella di complessità più elevata in quanto prevede che ogni nodo sia collegato direttamente con tutti gli altri nodi della rete con rami dedicati.

La caratteristica più importante di questa rete è che, dato un nodo qualsiasi, esiste sempre almeno un percorso che consente di collegarlo a un altro nodo qualsiasi della rete. Il vantaggio principale delle topologie magliate è la robustezza di fronte ai guasti nei collegamenti tra i nodi.

Svantaggi: Il rapporto quadratico tra numero di nodi e numero di rami costituisce un grosso ostacolo per la scalabilità: Aggiungere un nodo a una topologia completamente magliata richiede l'aggiunta di un numero sempre maggiore di rami, aumentando anche la complessità dell'intera rete.



- **A PARZIALMENTE MAGLIATA**: utilizza solo un sottoinsieme di tutti i collegamenti diretti definibili tra i nodi. Similmente alla rete completamente magliata, ci troviamo di fronte ad una rete molto robusta.

Svantaggi: gli stessi della rete completamente magliata

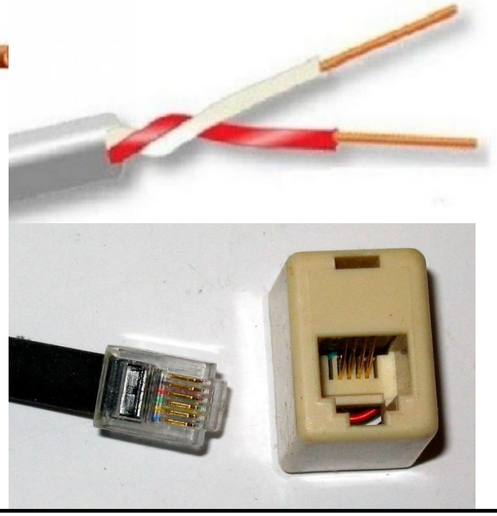
Grit

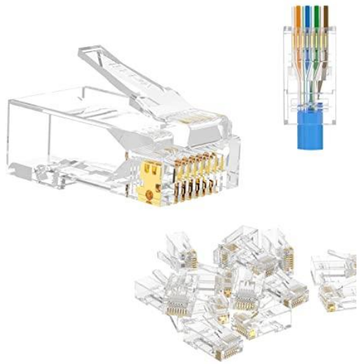
MEZZI TRASMISSIVI

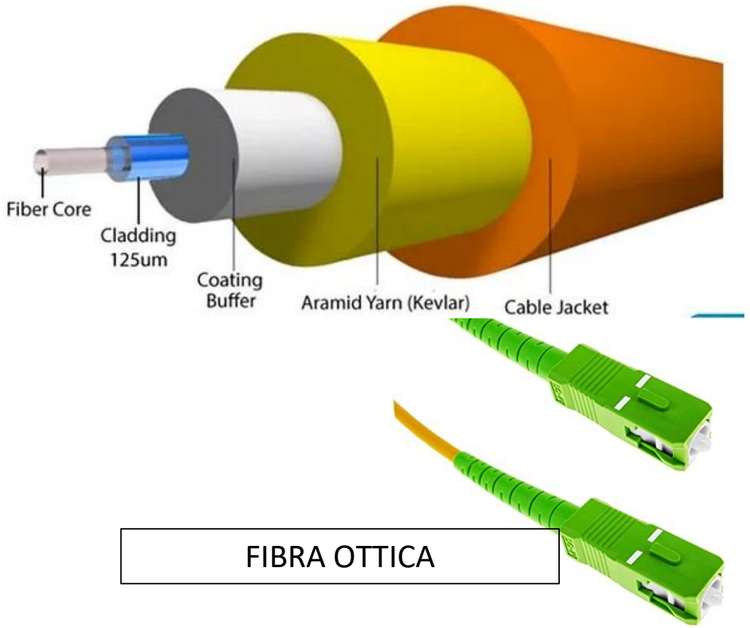
Il mezzo trasmissivo indica nelle telecomunicazioni il canale a livello fisico entro il quale viaggiano i segnali rappresentativi dell'informazione.

Un canale di trasmissione *ideale* dovrebbe possedere una banda sufficientemente larga ed uniforme per contenere lo spettro del segnale di informazione senza distorcerlo e dovrebbe poterlo trasferire a qualsivoglia distanza senza introdurre degradamenti nella qualità elettrica di origine; la realtà risulta diversa in quanto sono presenti fattori di degradazione tipici quali:

- **Attenuazione:** in fisica è la riduzione di intensità di un flusso di qualunque genere che attraversa un mezzo, ovvero la perdita di energia nel tempo e nello spazio da parte di un sistema fisico
- **Rumore di natura elettrica oppure segnali spuri cioè interferenza:** il rumore è l'insieme di segnali imprevisti e indesiderati che si sovrappongono al segnale utile e può provocare una perdita d'informazione o un'alterazione del messaggio trasmesso.
- **Distorsione:** si intende l'alterazione della forma originale di un oggetto, di un'immagine, di un suono, di un'onda o di un'altra forma di informazione o rappresentazione. Di solito è considerata un fenomeno indesiderato.







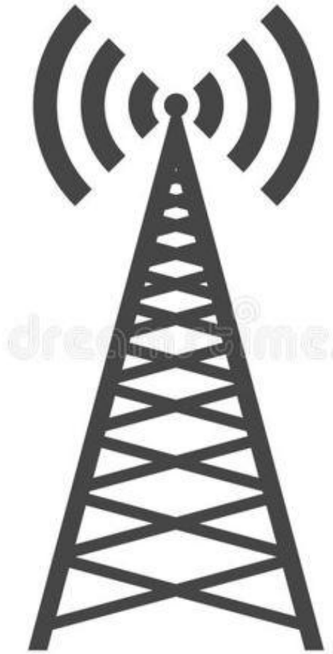
CAVO COASSIALE	DOPPINO TELEFONICO	CAVO ETHERNET	FIBRA OTTICA

SUPPORTI METALLICI IN RAME

MEZZI TRASMISSIVI

ONDE RADIO

In questo caso la trasmissione dei segnali viene effettuata attraverso lo spazio, che viene anche denominato portante radio (o hertziano). Il segnale irradiato deve essere di frequenza elevata e quindi sono necessari opportuni metodi di modulazione e devono essere usati sistemi che consentano di irradiare e captare onde elettromagnetiche (antenne trasmittenti e riceventi)



INDIRIZZO IP

L'*Internet Protocol Address* o indirizzo IP è un codice numerico usato da tutti i dispositivi (computer, server web, stampanti, modem) per navigare in Internet e per comunicare in una rete locale. Un indirizzo IP costituisce quindi la base per una trasmissione corretta delle informazioni dal mittente al ricevente.

TIPI DI INDIRIZZO IP

Ci sono due versioni di indirizzi IP: IPv4 e IPv6. IPv4 è la versione standard attualmente utilizzata ed è composta da 4 serie di numeri separati da punti che vanno da 0 a 255. In genere la prima serie di numeri è compresa tra 1 e 191 mentre le altre tre tra 0 e 255. Uno dei problemi che si potrebbero avere con l'IPv4 riguarda il numero limitato di combinazioni che con il tempo potrebbe rivelarsi insufficiente per il quantitativo di dispositivi presenti nel mondo. Tuttavia al momento la questione è irrilevante in quanto non vengono utilizzati tutti contemporaneamente e il numero di combinazioni è ancora sufficiente.
IPv6 è la versione più recente di un indirizzo IP ed è formata da 8 gruppi di 4 cifre esadecimali separati da due punti. In teoria l'IPv6 permette la creazione di infiniti IP eliminando di fatto il problema che potrebbe verificarsi con l'IPv4.

IP PUBBLICO E PRIVATO

L'indirizzo IP pubblico è un indirizzo visibile e raggiungibile da tutti gli host della rete Internet, mentre quello privato viene usato per identificare in modo univoco un dispositivo appartenente a una rete locale. Gli indirizzi IP privati quindi non possono essere utilizzati per l'accesso a Internet.

INDIRIZZI IP STATICI E DINAMICI

Gli indirizzi IP possono essere dinamici e statici.
I primi sono i più utilizzati e vengono impiegati per la normale navigazione online. Quando un utente si connette a Internet il suo Internet Service Provider gli assegna un indirizzo IP casuale che cambia dopo ogni sessione o a intervalli regolari (ogni 24 ore). Gli IP dinamici garantiscono una maggiore protezione della privacy degli utenti consentendo una navigazione più anonima.
Un indirizzo IP statico invece rimane invariato, tuttavia il proprietario può richiederne la modifica. Gli IP statici sono utilizzati principalmente nelle LAN (reti private) per comunicare, con una stampante o un altro computer della rete locale.

PROTEZIONE DATI

Gli indirizzi IP non contengono particolari informazioni o dati sensibili, tuttavia tramite un IP è possibile risalire al provider Internet di un utente o determinare in maniera più o meno precisa la posizione in base alla sua vicinanza al più vicino punto di presenza Internet. Questo tipo di localizzazione diventa molto più precisa nelle aree urbane, mentre per le zone rurali al massimo è possibile determinare la regione o la provincia da cui l'utente si sta connettendo.
Solo gli internet provider possono monitorare e tracciare tramite gli indirizzi IP, il flusso dei dati dei propri clienti e quindi devono impegnarsi a proteggerli in modo adeguato.

Grit

TCP

In telecomunicazioni e informatica il **Transmission Control Protocol** (**TCP**) è un protocollo di rete a pacchetto di livello di trasporto, appartenente alla suite di protocolli Internet, che si occupa di *controllo della trasmissione* ovvero rendere *affidabile* la comunicazione dati in rete tra mittente e destinatario. Definito nella RFC 793, su di esso si appoggia gran parte delle applicazioni della rete Internet, è presente solo sui terminali di rete (host) e non sui nodi interni di commutazione della rete di trasporto

UDP

Lo **User Datagram Protocol** (**UDP**), nelle telecomunicazioni, è uno dei principali protocolli di rete della suite di protocolli Internet. È un protocollo di livello di trasporto a pacchetto, usato di solito in combinazione con il protocollo di livello di rete IP

Confronto con UDP

Le principali differenze tra TCP e UDP (*User Datagram Protocol*), l'altro principale protocollo di trasporto della suite di protocolli Internet, sono:

- TCP è un protocollo orientato alla connessione. Pertanto, per stabilire, mantenere e chiudere una connessione, è necessario inviare segmenti di servizio i quali aumentano l'overhead di comunicazione. Al contrario, UDP è senza connessione ed invia solo i datagrammi richiesti dal livello applicativo; (nota: i pacchetti prendono nomi diversi a seconda delle circostanze a cui si riferiscono: segmenti (TCP) o datagrammi (UDP));
- UDP non offre nessuna garanzia sull'affidabilità della comunicazione, ovvero sull'effettivo arrivo dei segmenti, né sul loro ordine in sequenza in arrivo; al contrario il TCP, tramite i meccanismi di acknowledgement e di ritrasmissione su timeout, riesce a garantire la consegna dei dati, anche se al costo di un maggiore overhead (raffrontabile visivamente confrontando la dimensione delle intestazioni dei due protocolli); TCP riesce altresì a riordinare i segmenti in arrivo presso il destinatario attraverso un campo del suo header: il sequence number;
- l'oggetto della comunicazione di TCP è un *flusso di byte,* mentre quello di UDP sono *singoli datagrammi*.

L'utilizzo del protocollo TCP rispetto a UDP è, in generale, preferito quando è necessario avere garanzie sulla consegna dei dati o sull'ordine di arrivo dei vari segmenti (come per esempio nel caso di trasferimenti di file). Al contrario UDP viene principalmente usato quando l'interazione tra i due host è idempotente o nel caso si abbiano forti vincoli sulla velocità e l'economia di risorse della rete (es. streaming in tempo reale, videogiochi multiplayer).

Il termine *idempotente* viene usato in senso lato per riferirsi a funzioni prive di effetti collaterali.

IL MODELLO ISO/OSI

Il modello ISO/OSI definisce come l'informazione deve essere inviata e ricevuta

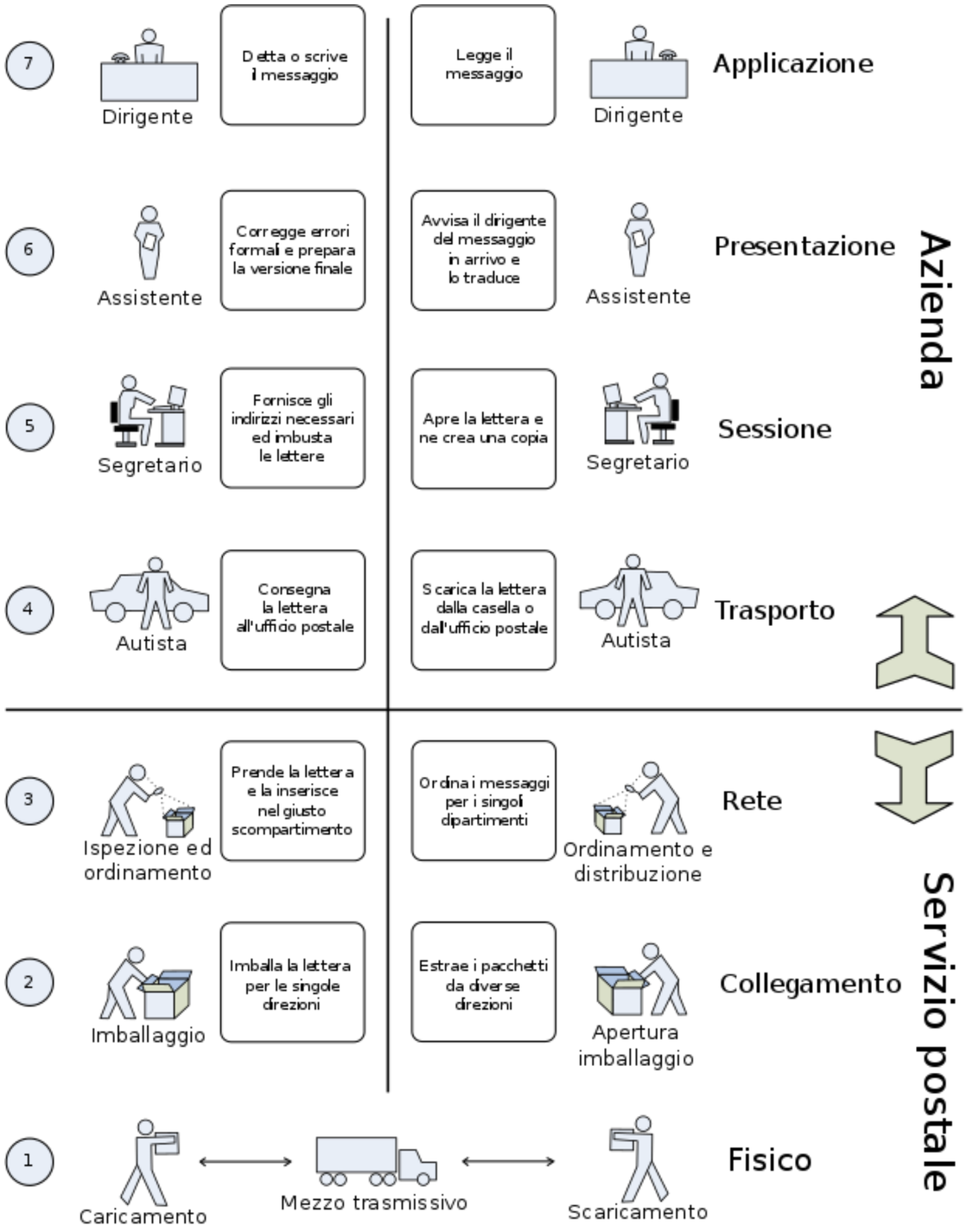
ISO: International Standard Interconnection **OSI**: Open System Interconnection

I livelli sono 7, il computer li usa tutti, il router ne usa 5 (non usa presentazione e sessione)

- **Applicazione**: Vengono gestiti i protocolli (HTTP, HTTPS, FTP, SMTP, IMAP)
- **Presentazione**: (Estensione del livello di applicazione)
- **Sessione**: (Estensione del livello di applicazione)
- **Trasporto**: Trasporta le informazioni tra i punti terminali delle applicazione (i protocolli utilizzati sono TCP e UDP)
- **Rete**: Fornisce il servizio di consegna all'indirizzo richiesto dal livello di trasporto (il protocollo utilizzato è IP)
- **Collegamento**: Livello di scambio informazioni tra il livello rete e il livello trasporto
- **Fisico**: trasforma i dati da frame a dati inviabili nei cavi ad altri PC

IL MODELLO ISO/OSI

Parallelo tra invio di una lettera e il modello ISO/OSI



IL MODELLO ISO/OSI

Livello 1 – Fisico

Supponiamo che siano già state compiute tutte le azioni che vanno dalla numero 7 alla numero 1 nella parte sinistra della figura: l'addetto postale ha appena caricato il pacco contenente la lettera da spedire sul furgone, che deve ora raggiungere l'ufficio postale di zona per recapitare il messaggio.
Come avviene il trasferimento?
Avviene grazie ai protocolli che regolano la trasmissione dei dati tra i due nodi della rete. Qui i bit (l'unità fondamentale dei dati) contenuti in un pacchetto si trasformano in un segnale fisico adatto per il mezzo di trasmissione. Questo segnale può essere trasmesso solo mediante un filo di rame, fibra ottica o per via aerea. I protocolli comuni regolano la comunicazione con il mezzo di trasmissione.
Grazie a protocolli comuni, che potrebbero essere paragonati alle regole di convenzione per regolare gli scambi postali, l'ufficio postale di zona riceve la lettera.

Livello 2 – Collegamento

In questo livello, l'addetto postale apre le scatole e raggruppa le lettere provenienti da diversi uffici postali.
I nostri dati, trasferiti sotto forma di segnale fisico, vengono riorganizzati in pacchetti per viaggiare lungo la dorsale di comunicazione. Il secondo livello ordina i pacchetti e li modifica in modo che abbiano un'intestazione (header – l'inizio del messaggio) e una coda (tail – la fine del messaggio).
È importante sapere che per ogni pacchetto ricevuto il destinatario trasmette al mittente una conferma di ricevuta (segnale di acknowledgement – ACK), facendo così capire al mittente quali pacchetti siano o meno arrivati a destinazione. Nel caso di pacchetti mal trasmessi (corrotti o incompleti) o persi, il mittente deve occuparsi della loro ritrasmissione.

Livello 3 – Rete

L'addetto ordina le lettere e le distribuisce ai diversi destinatari. A questo livello i pacchetti di dati hanno raggiunto la rete Internet e vengono spediti ai dispositivi: ogni dispositivo viene rintracciato tramite un indirizzo IP univoco.
L'attività di spedizione è chiamata routing (o instradamento) e individua il percorso di rete ottimale da utilizzare per la consegna dei pacchetti. Fisicamente, il responsabile di questa funzione è proprio il router. In caso di comunicazioni tra due reti diverse, il livello di rete si occupa anche della conversione dei dati. Ai pacchetti di dati contenenti le informazioni della nostra lettera, si aggiunge un network header, che comprende le informazioni sul routing e il controllo del flusso di dati.

IL MODELLO ISO/OSI

Livello 4 – Trasporto
È il livello intermedio che si occupa del trasporto fisico dei dati. La lettera viene ritirata e trasportata verso l'azienda a cui abbiamo spedito il messaggio.
A questo punto i protocolli stabiliscono tutto ciò che riguarda la connessione tra i due sistemi (sorgente e destinatario). Ai nostri dati viene aggiunto un ulteriore pacchetto, il transport header, che contiene le informazioni relative all'assegnazione dei pacchetti alle applicazioni di pertinenza (nel nostro esempio: a quale azienda e a quale ufficio deve essere recapitata la lettera).
L'obiettivo del livello di trasporto è garantire che i pacchetti arrivino nell'ordine corretto senza alcun errore o perdita di dati. Infatti, i protocolli gestiscono la connessione (che non deve durare più dello stretto necessario per evitare di congestionare la rete) e controllano il sovraccarico dei router di rete (evitando che troppi pacchetti di dati arrivino allo stesso router nello stesso momento).

Livello 5 – Sessione
È detto anche livello di comunicazione perché si occupa di gestire l'intero processo di trasmissione dei dati tra i sistemi che fanno parte della comunicazione (sorgente e destinatario). Questo processo è chiamato sessione. In questa fase si verifica che il significato del messaggio non venga deformato. Per il controllo della comunicazione, questo layer incorpora ulteriori informazioni ai nostri dati aggiungendo un pacchetto chiamato session header.
La nostra lettera è stata appena aperta dalla segretaria in azienda ed è stata letta per capire a chi deve essere indirizzata.

Livello 6 – Presentazione o Traduzione
La segretaria avvisa il manager che è arrivata una lettera e la traduce in modo che il manager possa leggerla facilmente.
I protocolli appartenenti a questo layer consentono di trasformare i dati in un formato standard tramite la crittografia e la formattazione. Viene definito come il modo in cui dovrebbero apparire i pacchetti di dati che compongono la lettera. Per questo motivo, viene aggiunto al pacchetto un presentation header, che contiene informazioni su come l'e-mail deve essere codificata.

Livello 7 – Applicazione
Questo livello è il più "vicino" all'utente finale e opera direttamente sui software come browser o programmi di posta elettronica. Le funzioni tipiche di questi protocolli che interagiscono con i software sono l'identificazione dei partner nella comunicazione, l'identificazione delle risorse disponibili e la sincronizzazione della comunicazione.
A questo punto, grazie all'ultimo livello, il manager può leggere il messaggio inviato dalla prima azienda.

TCP / IP

Cos'è TCP/IP

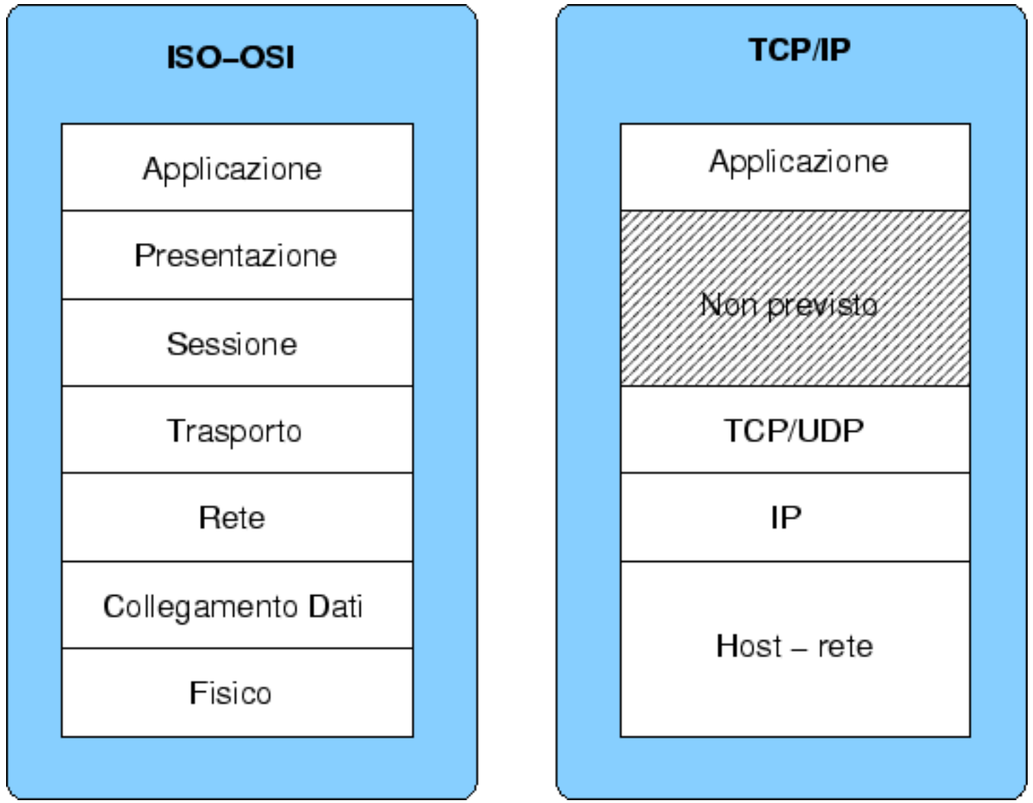
Il protocollo TCP/IP indica l'uso combinato di due protocolli per la trasmissione di dati su internet, TCP (Transmission Control Protocol) e IP (Internet Protocol). Il protocollo TCP crea la connessione tra due host e gestisce la consegna dei pacchetti da un sistema all'altro, mentre il protocollo IP fornisce le istruzioni per il trasferimento dei dati.

Livelli del modello TCP/IP

- **Livello di interfaccia di rete**: questo livello funge da interfaccia tra host e collegamenti di trasmissione e viene utilizzato per la trasmissione di datagrammi. Specifica inoltre quale operazione deve essere eseguita da collegamenti come collegamento seriale e Ethernet classica per soddisfare i requisiti del livello Internet senza connessione.

- **Livello Internet**: lo scopo di questo livello è trasmettere un pacchetto indipendente in qualsiasi rete che viaggia verso la destinazione (potrebbe risiedere in una rete diversa). Include IP (Internet Protocol), ICMP (Internet Control Message Protocol) e ARP (Address Resolution Protocol) come formato di pacchetto standard per il livello.

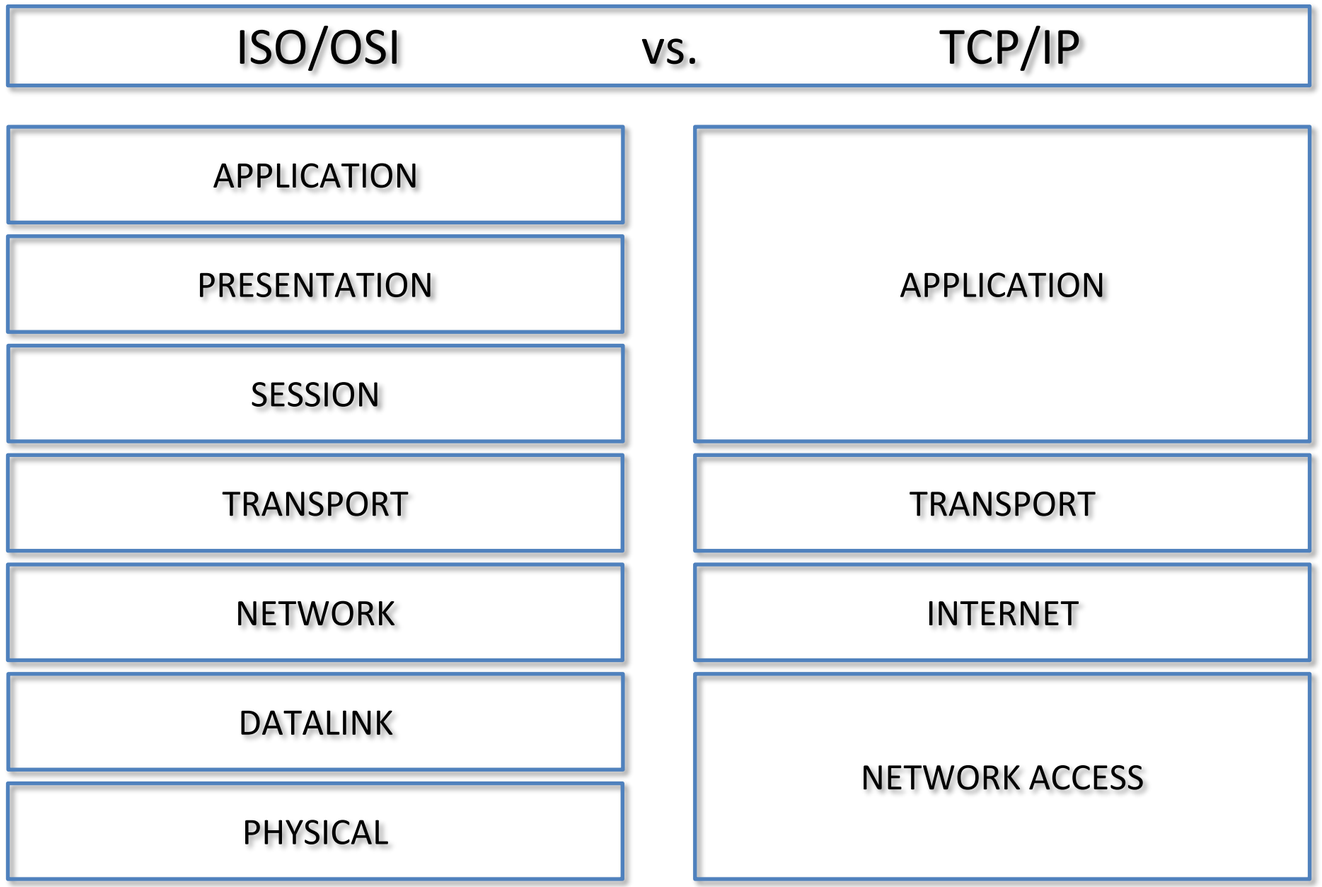
- **Livello di trasporto**: consente una consegna end-to-end senza errori dei dati tra gli host di origine e di destinazione sotto forma di datagrammi. I protocolli definiti da questo livello sono TCP (Transmission Control Protocol) e UDP (User Datagram Protocol).



- **Livello applicazione**: questo livello consente agli utenti di accedere ai servizi di Internet globale o privato. I vari protocolli descritti in questo livello sono il terminale virtuale (TELNET), la posta elettronica (SMTP) e il trasferimento di file (FTP). Alcuni protocolli aggiuntivi come DNS (Domain Name System), HTTP (Hypertext Transfer Protocol) e RTP (Real-time Transport Protocol). Il funzionamento di questo livello è una combinazione di applicazione, presentazione e livello di sessione del modello OSI.

TCP / IP Vs ISO/OSI

- TCP/IP è un modello client-server, cioè quando il client richiede un servizio, viene fornito dal server. Considerando che, OSI è un modello concettuale.
- TCP/IP è un protocollo standard utilizzato per ogni rete inclusa Internet, mentre OSI non è un protocollo ma un modello di riferimento utilizzato per comprendere e progettare l'architettura del sistema.
- TCP/IP è un modello a quattro livelli, mentre OSI ha sette livelli.
- TCP/IP segue l'approccio orizzontale. D'altra parte, il modello OSI supporta l'approccio verticale.
- TCP/IP è tangibile, mentre OSI non lo è.



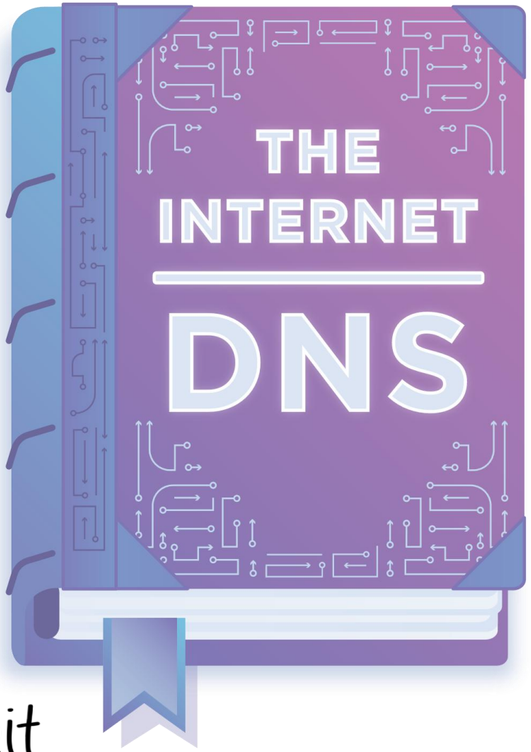
DNS - Domain Name System

DNS (Domain Name System) è il sistema che regola la traduzione dei nomi dominio dei siti internet in indirizzi IP.

Per aprire una pagina Web infatti è necessario l'indirizzo IP, cioè un codice numerico che sta alla base della trasmissione corretta delle informazioni dal mittente al ricevente.

l 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:cb00:2048:4321:c629:d7a2 (in IPv6)

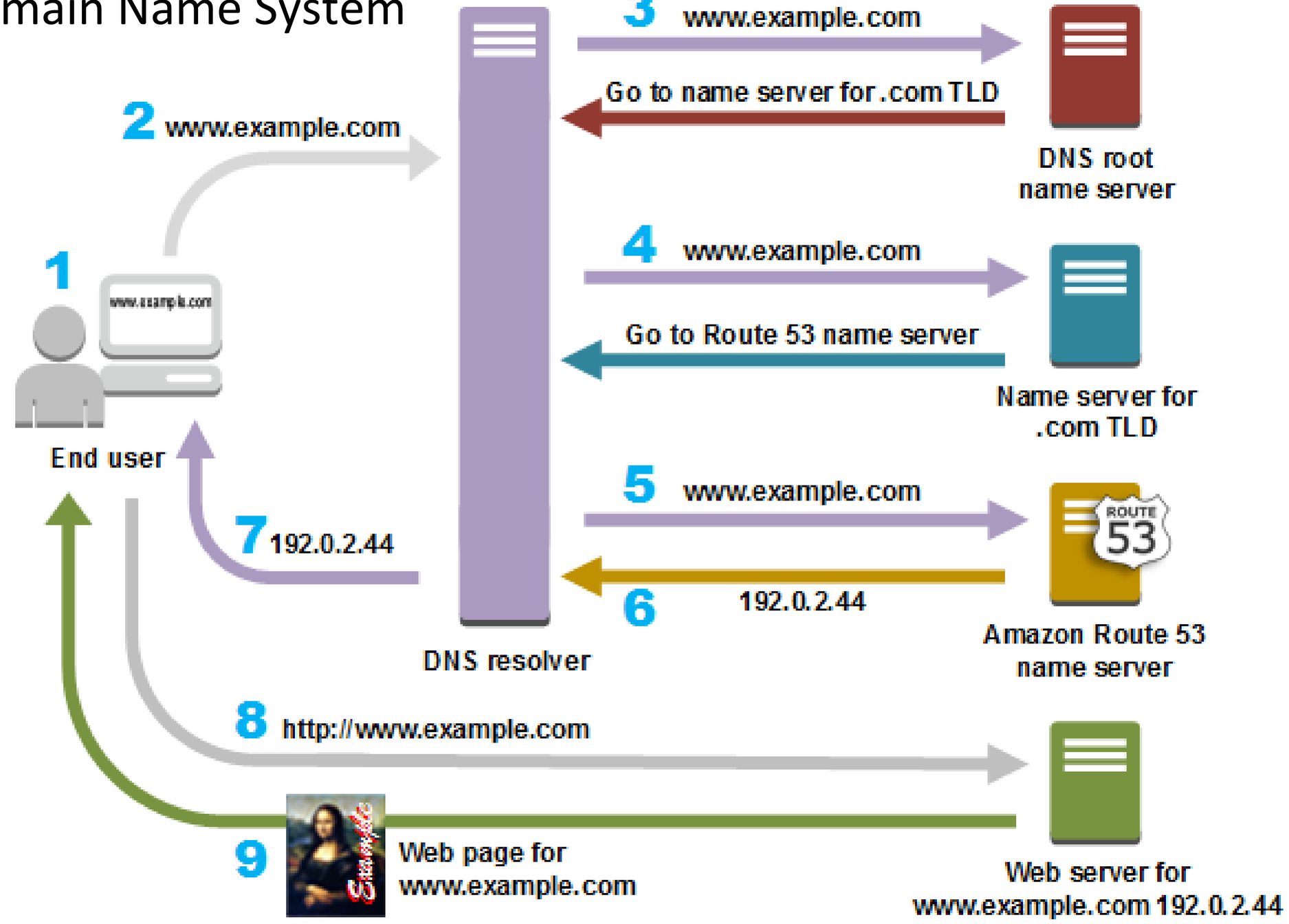


COME FUNZIONA IL DNS?

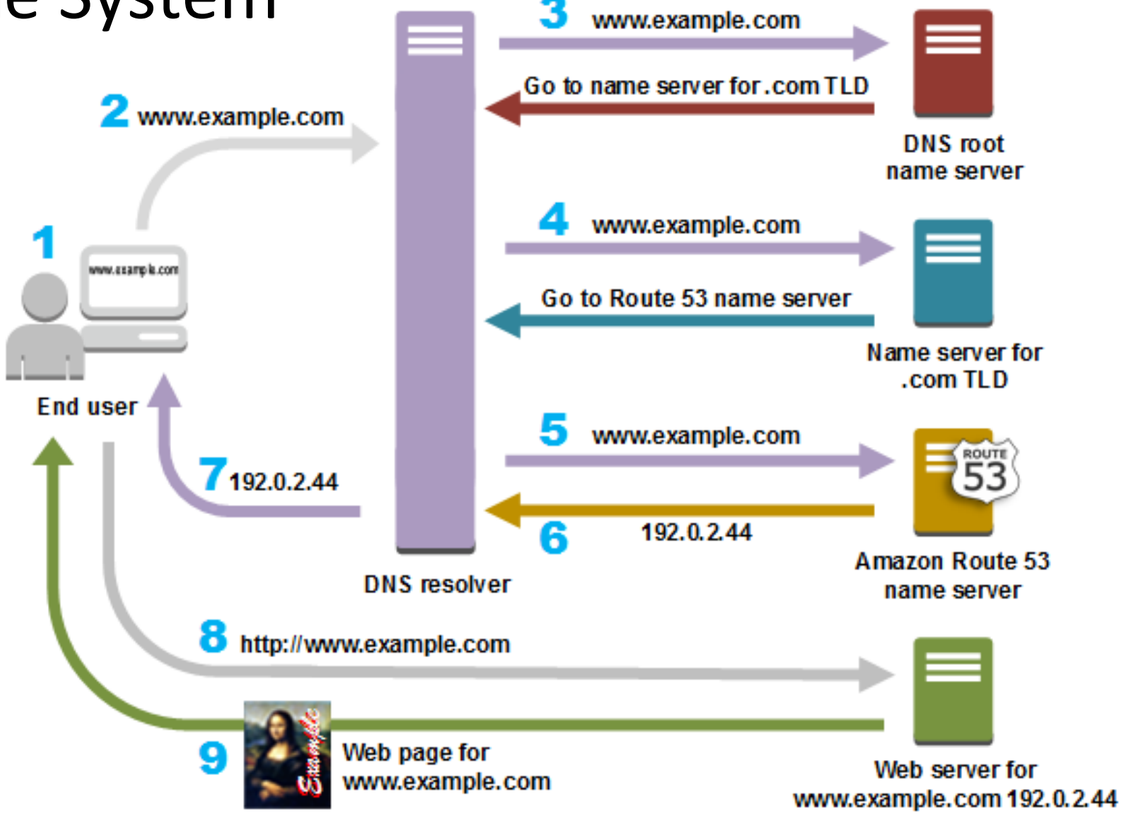
Il processo di risoluzione DNS prevede la conversione di un hostname (www.example.com) in un indirizzo IP compatibile con il computer (come 192.168.1.1). Si assegna un indirizzo IP a ciascun dispositivo su Internet, e tale indirizzo è necessario per trovare il dispositivo Internet appropriato, proprio come per trovare una determinata casa si utilizza un indirizzo stradale. Quando un utente desidera caricare una pagina web, deve avvenire una traduzione tra ciò che un utente digita nel proprio browser web (esempio.com) e l'indirizzo riconoscibile dalla macchina necessario per individuare la pagina esempio.com.

Grit

DNS - Domain Name System



DNS - Domain Name System



1. Un utente apre un browser Web, digita www.example.com nella barra degli indirizzi e preme Invio.
2. La richiesta per www.example.com viene instradata a un resolver DNS, che in genere è gestito da un provider di servizi Internet (ISP), ad esempio un operatore di telefonia che offre connessione via cavo o DSL.
3. Il resolver DNS per l'ISP inoltra la richiesta di www.example.com verso un server DNS dei nomi radice.
4. Il resolver DNS per l'ISP inoltra nuovamente la richiesta di www.example.com, questa volta verso uno dei server dei nomi di primo livello per i domini .com. Il server dei nomi per i domini .com risponde alla richiesta con i nomi dei quattro server dei nomi di Amazon Route 53 associati al dominio example.com.
5. Il resolver DNS per l'ISP sceglie un server dei nomi di Amazon Route 53 e inoltra la richiesta per www.example.com a tale server.
6. Il server dei nomi di Amazon Route 53 cerca il record www.example.com nella zona ospitata example.com, ottiene il valore associato, ovvero l'indirizzo IP di un server Web, 192.0.2.44, e lo restituisce al resolver DNS.
7. Il DNS resolver per l'ISP ha così a disposizione l'indirizzo IP richiesto dall'utente. Il resolver restituisce tale valore al browser Web. Il resolver DNS memorizza inoltre nella cache l'indirizzo IP di example.com per una quantità di tempo specificata, così potrà rispondere più rapidamente al successivo tentativo di connessione ad example.com.
8. Il browser Web invia una richiesta per www.example.com all'indirizzo IP ottenuto dal resolver DNS. Qui si trovano i contenuti, ad esempio un server Web in esecuzione configurato come endpoint di sito Web.
9. Il server Web o la risorsa che si trova all'indirizzo 192.0.2.44 restituisce la pagina Web per www.example.com al browser Web, che visualizzerà così la pagina.

IL MODELLO CLIENT/SERVER

Il modello client/server suddivide i sistemi connessi alla rete in due categorie:
Computer server: che mettono in condivisione delle risorse o offrono servizi per gli altri computer connessi alla rete.
Computer client: i sistemi che utilizzano i servizi e le risorse messe in condivisione da altri computer.

In questa rete ci sono quindi i client che richiedono i servizi e i server che li offrono.
In alcune reti la distinzione tra client e server è netta: un client non può diventare server e né un server può diventare un client (viene detto server dedicato).

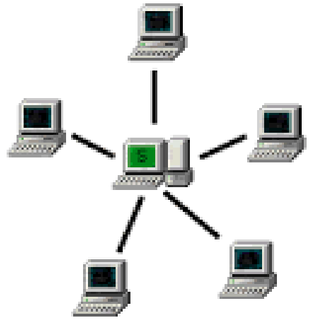
Di solito, nelle reti di grandi dimensioni, vengono utilizzati come server i componenti più potenti, in quanto devono offrire numerosi servizi contemporaneamente.

In questi casi si usa spesso il termine host per indicare sistemi di elaborazione di grandi prestazioni, destinati ad essere centro di distribuzione di informazioni per gli utenti della rete.

In questo modello la comunicazione ha la forma di un messaggio a un server da parte di un client, che richiede l'esecuzione di un lavoro. Il server esegue il lavoro e restituisce la risposta.

Un messaggio è un insieme di caratteri e di dati che devono essere trasferiti da un sistemi ad un altro; e quindi di informazioni organizzate in modo da costruire un'entità completa che può essere trasmessa fra due sistemi di rete. Per favorire la trasmissione dei dati, i messaggi vengono suddivisi in segmenti chiamati pacchetti, secondo l'idea che un messaggio di piccole dimensioni ha più possibilità di trovare una linea libera. Quando un server e un client stabiliscono una comunicazione e inizia l'erogazione del servizio richiesto, si hanno due possibilità:

- esecuzione lato client
- esecuzione lato server

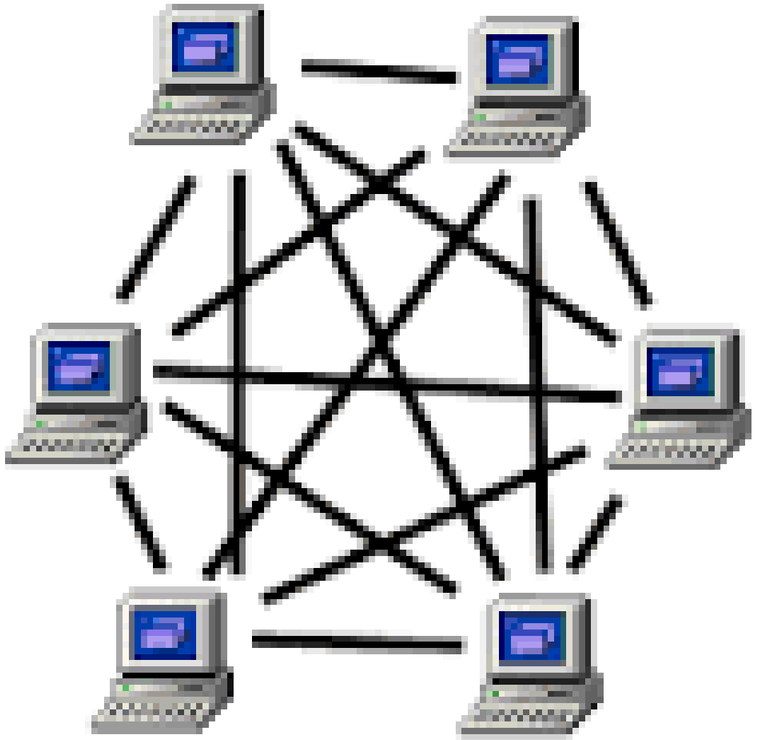


Grit

IL MODELLO PEER TO PEER

Nelle reti peer to peer ogni computer funziona sia come client che come server, mettendo a disposizione dell'intera rete alcune risorse condivise come dischi fissi, stampanti o lettori ottici.

Per esempio Windows permette di condividere cartelle o stampanti in rete, creando automaticamente delle reti di tipo peer to peer.



ALAN MATHISON TURING

Alan Mathison Turing
(Londra, 23 giugno 1912 – Wilmslow, 7 giugno 1954)
è stato un matematico, logico e crittografo britannico, considerato uno dei padri dell'informatica e uno dei più grandi matematici del XX secolo.

Il suo lavoro ebbe vasta influenza sullo sviluppo dell'informatica, grazie alla sua formalizzazione dei concetti di algoritmo e calcolo mediante la macchina di Turing, che a sua volta ha svolto un ruolo significativo nella creazione del moderno computer.
Per questo contributo Turing è solitamente considerato il padre della scienza informatica e dell'intelligenza artificiale, da lui teorizzate già negli anni trenta (quando non era ancora stato creato il primo vero computer).

Fu anche uno dei più brillanti crittoanalisti che operavano in Inghilterra, durante la seconda guerra mondiale, per decifrare i messaggi scambiati da diplomatici e militari delle Potenze dell'Asse.

Turing lavorò infatti a Bletchley Park, il principale centro di crittoanalisi del Regno Unito, dove ideò una serie di tecniche per violare i cifrari tedeschi, incluso il metodo della Bomba, una macchina elettromeccanica in grado di decodificare codici creati mediante la macchina Enigma.

Morì suicida a soli 42 anni, in seguito alle persecuzioni subite da parte delle autorità britanniche a causa della sua omosessualità.

Grit

MACCHINA DI TURING

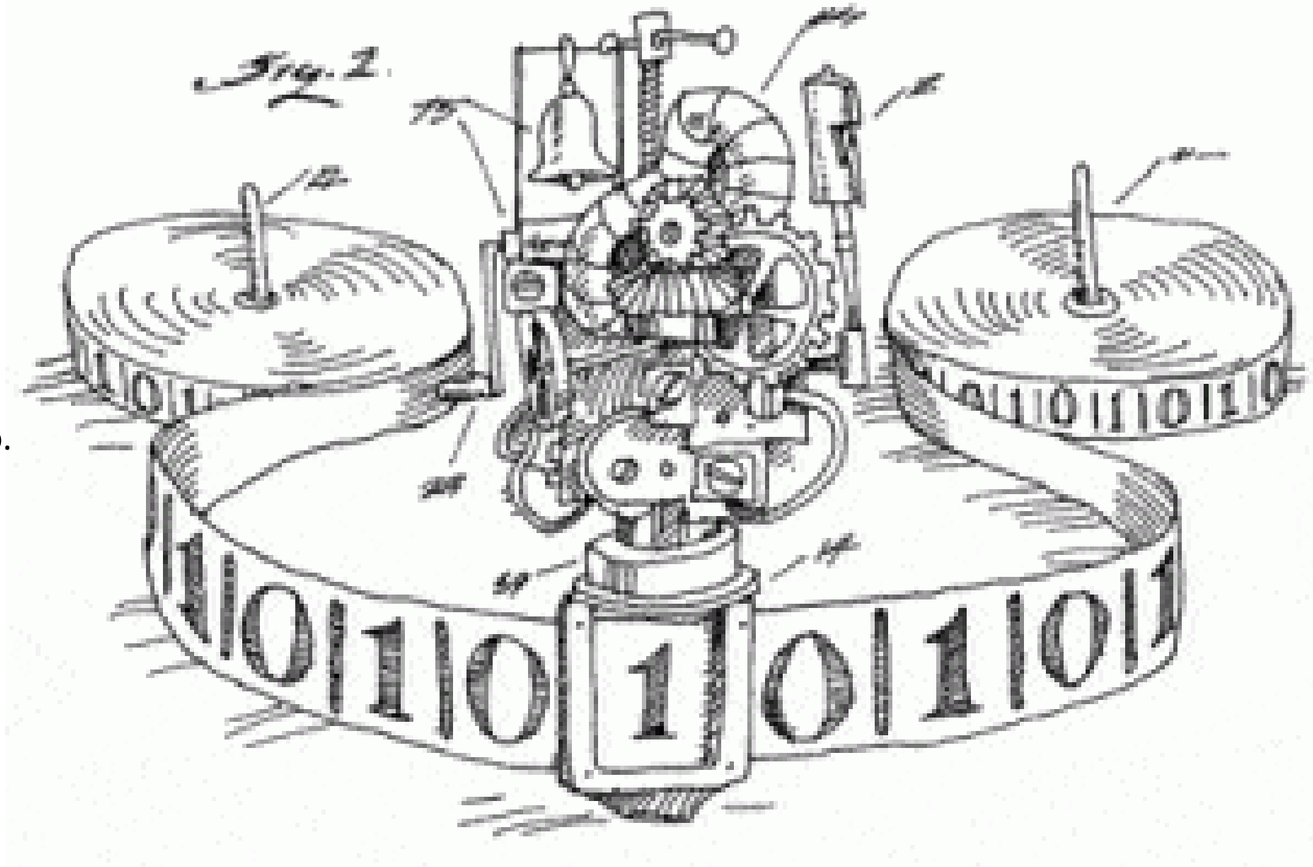
La **macchina di Turing** (o più brevemente **MdT**) è una macchina ideale che manipola i dati contenuti su un nastro di lunghezza potenzialmente infinita, secondo un insieme prefissato di regole ben definite. In altre parole, si tratta di un modello astratto che definisce una macchina in grado di eseguire algoritmi e dotata di un nastro potenzialmente infinito su cui può leggere e/o scrivere dei simboli.

Essa ha la particolarità di essere retta da regole di natura molto semplice, ovvero di potersi descrivere come costituita da meccanismi elementari molto semplici; inoltre è possibile presentare a livello sintetico le sue evoluzioni mediante descrizioni meccaniche piuttosto intuitive. D'altra parte, essa ha la computabilità (*una funzione si dice computabile se esiste un algoritmo che la calcola*) che si presume essere la massima: si dimostra, infatti, che essa è in grado di effettuare *tutte* le elaborazioni effettuabili dagli altri modelli di calcolo noti all'uomo.

La MdT dal punto di vista informale è formata da:

- Un nastro
- Una testina di lettura/scrittura
- Unità di memoria interna
- Unità di calcolo
- Unità logica

Il nastro è lo strumento che contiene le informazioni, è diviso in celle che possono contenere un singolo simbolo appartenente all'**alfabeto di lavoro**.



Grit

MACCHINA DI TURING

La soluzione proposta da Turing consiste nell'utilizzo di un modello matematico (*un **modello matematico** è una rappresentazione quantitativa di un fenomeno naturale*) capace di simulare il processo di calcolo umano, scomponendolo nei suoi passi ultimi.

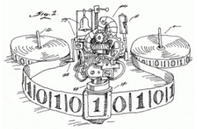
La macchina è formata da una testina di lettura e scrittura con cui è in grado di leggere e scrivere su un nastro potenzialmente infinito partizionato, in maniera discreta, in caselle. Ad ogni istante di tempo t_1, la macchina si trova in uno stato interno s_1 ben determinato, risultato dell'elaborazione compiuta sui dati letti.
Lo stato interno, o configurazione, di un sistema è la condizione in cui si trovano le componenti della macchina ad un determinato istante di tempo *t*.

Le componenti da considerare sono:
- il numero della cella osservata
- il suo contenuto
- l'istruzione da eseguire

Tra tutti i possibili stati, si distinguono:
- una configurazione *iniziale*, per $t=t_0$ (prima dell'esecuzione del programma)
- una configurazione *finale*, per $t=t_n$ (al termine dell'esecuzione del programma)
- delle configurazioni *intermedie*, per $t=t_i$ (prima dell'esecuzione dell'istruzione o_i)

Implementare un algoritmo in questo contesto significa effettuare una delle quattro operazioni elementari
- spostarsi di una casella a destra
- spostarsi di una casella a sinistra
- scrivere un simbolo preso da un insieme di simboli a sua disposizione su una casella
- cancellare un simbolo già scritto sulla casella che sta osservando
- oppure fermarsi

Eseguire un'operazione o_1, tra gli istanti di tempo t_1 e t_2, vuol dire passare dallo stato interno s_1 al s_2. Più formalmente questo si esprime in simboli come: $\{s_1,a_1,o_1,s_2\}$ da leggersi come: *nello stato interno s_1 la macchina osserva il simbolo a_1, esegue l'operazione o_1 e si ritrova nello stato interno s_2*.
Turing poté dimostrare che un tale strumento, dalle caratteristiche così rigidamente definite, è in grado di svolgere un qualsiasi calcolo, ma non si fermò qui; egli capì che la *calcolabilità* era parente stretta della *dimostrabilità* e dunque le sue macchine potevano definitivamente chiudere la questione dell'*Entscheidungsproblem (problema della decisione)*.



MACCHINA DI TURING

L'importanza della MdT deriva dal fatto che permette di compiere *tutte* le elaborazioni effettuate mediante le macchine (elettroniche o meccaniche) apparse nella storia dell'umanità, incluse le elaborazioni che oggi si eseguono con le tecnologie più avanzate e gli odierni computer, e perfino le dimostrazioni matematiche che l'umanità ha raccolto nel corso della sua storia.
Infatti, tutte le macchine che si conoscono possono essere ricondotte al modello estremamente semplice di Turing. Ad esempio, anche i più complessi computer odierni possono essere ricondotti a questo modello

Con questo ragionamento si ottengono macchine formali che, in linea di principio, si possono ricondurre alla MdT introdotta inizialmente, ma che possono essere programmate molto più agevolmente, e soprattutto che possono essere realizzate con le tecnologie disponibili oggi.

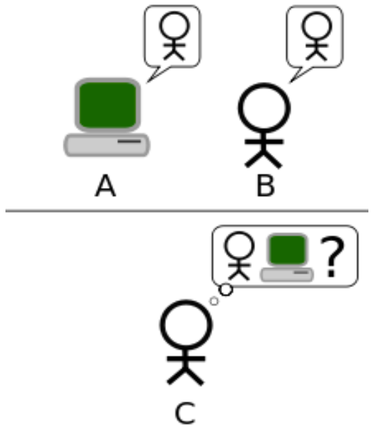
Dimostrare che una di queste macchine può risolvere un certo problema vuol dire dimostrare che anche la MdT può risolverlo. La conclusione è che tutte le computazioni effettuabili dalle macchine a noi note sono effettuabili anche dalla MdT.

Una macchina che permetta di risolvere *tutti* i problemi risolvibili anche dalla MdT si dice "Turing-equivalente".

La conclusione è che tutte le computazioni effettuabili dalla MdT sono effettuabili anche da tutte le macchine di cui si è in grado di dimostrare l'equivalenza con la MdT.

Quindi, l'importanza della MdT è duplice: non solo è il modello teorico di macchina più "potente" che si conosca, ma può essere usato anche per verificare la potenza di nuovi modelli teorici.

TEST DI TURING



Il test di Turing è stato il primo passo nell'intelligenza artificiale. Turing si è fatto una domanda: i computer posso pensare? Per rispondere a questa domanda fece un test chiamato anche «*Gioco dell'imitazione*". Questo test si basa su 3 operatori chiamati A, B e C. A e B si parlano a vicenda, in mezzo tra A/B e C è situato un muro che impedisce la visione.

La comunicazione tra A/B e C è in formato scritto. A e C sono uomini e B è una donna. Inizialmente A e B fanno domande a C (simile a una conversazione), a un certo punto A si trasforma in una macchina (Bot) e B aiuta A nel gioco dell'imitazione.

Se C continua a pensare che A sia un uomo il test di Turing ha funzionato e quindi A è una intelligenza artificiale.

A.I.

Finora non abbiamo dato alcuna definizione di di intelligenza artificiale (IA), e in realtà non è affatto facile darne una di una qualche validità. Si potrebbe dire che l"IA" è il settore dell'informatica che si occupa di creare macchine intelligenti ovvero che simulano di pensare come l'essere umano.
In genere, si intende per intelligenza la capacità di completare compiti e risolvere problemi nuovi (problem solving), eventualmente sorti durante il completamento del compito originale, nonché la capacità di adattarsi all'ambiente e ai suoi mutamenti.
Si vuole aggiungere a questa definizione alquanto pragmatica ed operativa anche la capacità della comprensione della realtà, includendo in questa anche la comprensione del linguaggio.

La definizione di Intelligenza Artificiale si basa sul Test di Turing.

Esempi attuali di I.A. sono Siri di Apple, Alexa di Amazon e Cortana di Microsoft, che comunque non hanno superato al 100% il Test di Turing.
Probabilmente, l'unico fattore limitante al pieno utilizzo di strumenti in grado di imparare da soli è il timore dell'uomo che le macchine possano diventare troppo intelligenti, togliendogli facoltà di scelta e libertà.

Un timore che, come afferma il professore Pedro Domingos dell'Università di Washington, esperto di machine learning e data mining non esiste visto che **"la gente ha paura che i computer diventino troppo intelligenti e dominino il mondo, ma il vero problema è che pur essendo ancora troppo stupidi lo hanno già conquistato"**.

Grit

MACHINE LEARNING

Che cos'è il machine learning?

Quando si parla di **machine learning** (in italiano **apprendimento automatico**), si parla di una particolare branca dell'informatica che può essere considerata una parente stretta dell'intelligenza artificiale. Definire in maniera semplice le caratteristiche e le applicazioni del machine learning non è sempre possibile, visto che questo ramo è molto vasto e prevede differenti modalità, tecniche e strumenti per essere realizzato. Inoltre, le differenti tecniche di apprendimento e sviluppo degli algoritmi danno vita ad altrettante possibilità di utilizzo che allargano il campo di applicazione dell'apprendimento automatico rendendone difficile una definizione specifica.
Si può tuttavia dire che quando si parla di machine learning si parla di differenti **meccanismi che permettono a una macchina intelligente di migliorare le proprie capacità e prestazioni nel tempo**. La macchina, quindi, sarà in grado di imparare a svolgere determinati compiti migliorando, tramite l'esperienza, le proprie capacità, le proprie risposte e funzioni. Alla base dell'apprendimento automatico ci sono una serie di differenti algoritmi che, partendo da nozioni primitive, sapranno prendere una specifica decisione piuttosto che un'altra o effettuare azioni apprese nel tempo.

Un po' di storia.

Anche se oggi parlare di apprendimento automatico, di intelligenza artificiale, di computer e macchine intelligenti sembra quasi la normalità, per arrivare ai risultati odierni la strada è stata molto complessa, perché divisa tra **sperimentazioni e scetticismo**. Le prime sperimentazioni per la realizzazione di macchine intelligenti risalgono agli inizi degli anni Cinquanta del Novecento, quando alcuni matematici e statistici iniziarono a pensare di utilizzare i metodi probabilistici per realizzare macchine che potessero prendere decisioni proprio tenendo conto delle probabilità di accadimento di un evento.
Il primo grande nome legato al machine learning è sicuramente quello di **Alan Turing**, che ipotizzò la necessità di realizzare algoritmi specifici per realizzare **macchine in grado di apprendere**. In quegli stessi anni, anche gli studi sull'intelligenza artificiale, sui sistemi esperti e sulle reti neurali vedevano momenti di grossa crescita alternati da periodi di abbandono, causati soprattutto dalle molte difficoltà riscontrate nelle possibilità di realizzazione dei diversi sistemi intelligenti, nella mancanza di sussidi economici e dallo scetticismo che circondava spesso chi provava a lavorarci.
A partire dagli anni Ottanta, una serie di interessanti risultati ha portato alla rinascita di questo settore della ricerca: una rinascita che è stata resa possibile da nuovi **investimenti nel settore**.
Alla fine degli anni Novanta l'apprendimento automatico trova nuova linfa vitale in una serie di innovative tecniche legate ad elementi statistici e probabilistici: si trattava di un importante passo che permise quello sviluppo che ha portato oggi l'apprendimento automatico ad essere un **ramo della ricerca riconosciuto e altamente richiesto**.

Grit

MACHINE LEARNING

I diversi apprendimenti di una macchina

La strada per realizzare macchine intelligenti è stata lunga, ma ha portato, oggi, ad avere **differenti modalità di apprendimento**, tutte efficaci, che differiscono non solo per gli algoritmi utilizzati, ma soprattutto per lo scopo per cui sono realizzate le macchine stesse.

A seconda del tipo di algoritmo utilizzato per permettere l'apprendimento alla macchina, ossia a seconda delle modalità con cui la macchina impara ed accumula dati e informazioni, si possono suddividere tre differenti sistemi di apprendimento automatico: **supervisionato**, **non supervisionato** e **per rinforzo.**

I tre modelli di apprendimento sono utilizzati in maniera differente a seconda della macchina su cui si deve operare, garantendo così sempre la massima performance e il migliore risultato possibile per la risposta agli stimoli esterni.

L'apprendimento supervisionato:

Consiste nel fornire al sistema informatico della macchina una serie di nozioni specifiche e codificate, ossia di modelli ed esempi che permettono di costruire un vero e proprio **database di informazioni e di esperienze**.

In questo modo, quando la macchina si trova di fronte ad un problema, non dovrà fare altro che **attingere alle esperienze inserite nel proprio sistema**, analizzarle, e decidere quale risposta dare sulla base di esperienze già codificate. Questo tipo di apprendimento è, in qualche modo, fornito già confezionato e la macchina deve essere solo in grado di scegliere quale sia la migliore risposta allo stimolo che le viene dato.

Gli algoritmi che fanno uso di apprendimento supervisionato vengono utilizzati in molti settori, da quello medico a quello di identificazione vocale: essi, infatti, hanno la capacità di effettuare **ipotesi induttive**, ossia ipotesi che possono essere ottenute scansionando una serie di problemi specifici per ottenere una soluzione idonea ad un problema di tipo generale.

Grit

MACHINE LEARNING

L’apprendimento non supervisionato o senza supervisione:
Prevede invece che le informazioni inserite all’interno della macchina non siano codificate, ossia la macchina ha la possibilità di **attingere a determinate informazioni senza avere alcun esempio del loro utilizzo** e, quindi, senza avere conoscenza dei risultati attesi a seconda della scelta effettuata.
Dovrà essere la macchina stessa, quindi, a catalogare tutte le informazioni in proprio possesso, organizzarle ed imparare il loro significato, il loro utilizzo e, soprattutto, il risultato a cui esse portano.
L’apprendimento senza supervisione offre **maggiore libertà di scelta alla macchina** che dovrà organizzare le informazioni in maniera intelligente e imparare quali sono i risultati migliori per le differenti situazioni che si presentano.

L’apprendimento per rinforzo:
Rappresenta probabilmente il sistema di apprendimento più complesso, che prevede che la macchina sia dotata di sistemi e strumenti in grado di migliorare il proprio apprendimento e, soprattutto, di comprendere le caratteristiche dell’ambiente circostante.
In questo caso, quindi, alla macchina vengono forniti una serie di elementi di supporto, quali sensori, telecamere, GPS eccetera, che permettono di rilevare quanto avviene nell’ambiente circostante ed **effettuare scelte per un migliore adattamento all’ambiente** intorno a loro.
Questo tipo di apprendimento è tipico delle **auto senza pilota**, che grazie a un complesso sistema di sensori di supporto è in grado di percorrere strade cittadine e non, riconoscendo eventuali ostacoli, seguendo le indicazioni stradali e molto altro.

Grit

RETI NEURALI

Le **reti neurali**, la cui implementazione risultava impensabile fino a pochi decenni fa, farebbero affermare a un Julius Verne redivivo che la fantascienza finalmente ha fatto il suo ingresso nella realtà. I circuiti neurali artificiali sono la base di sofisticate forme di intelligenza artificiale, sempre più evolute, in grado di **apprendere sfruttando meccanismi simili (almeno in parte) a quelli dell'intelligenza umana**.

Risultato: prestazioni impossibili per altri algoritmi.

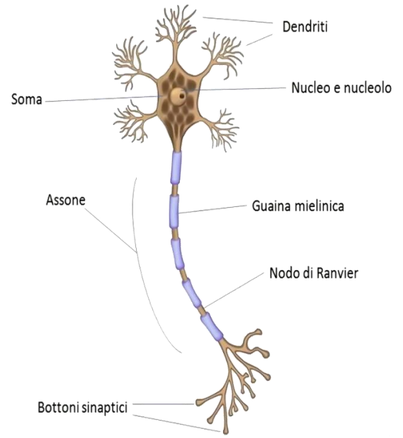
Le reti neurali artificiali riescono oggi a risolvere determinate categorie di problemi avvicinandosi sempre più all'efficienza del nostro cervello, e trovando perfino soluzioni inaccessibili alla mente umana. Dalla nascita del concetto di neurone artificiale ad oggi è stata fatta molta strada. In moltissimi ed eterogenei settori scientifici, dalla biomedicina al data mining, **le reti neurali hanno ormai un impiego quotidiano**.

Si tratta di un trend in crescita.

I continui progressi permettono di ottenere circuiti sempre più sofisticati.
Tutto lascia prevedere, insomma, che le reti neurali e il machine learning saranno parte notevole delle fondamenta del mondo ipertecnologico in cui ci stiamo addentrando.

Grit

RETI NEURALI



Il modello di riferimento: le reti neurali biologiche
Il prototipo delle reti neurali artificiali sono quelle biologiche.
Le reti neurali del cervello umano sono la sede della nostra capacità di comprendere l'ambiente e i suoi mutamenti, e di fornire quindi **risposte adattive calibrate sulle esigenze che si presentano**.

Sono costituite da insiemi di cellule nervose fittamente interconnesse fra loro. Al loro interno troviamo:

- i **somi neuronali**, ossia i corpi dei neuroni. Ricevono e processano le informazioni; in caso il potenziale di azione in ingresso superi un certo valore, generano a loro volta degli impulsi in grado di propagarsi nella rete;
- i **neurotrasmettitori**, composti biologici di diverse categorie (ammine, peptidi, aminoacidi), sintetizzati nei somi e responsabili della modulazione degli impulsi nervosi;
- gli **assoni o neuriti**: la via di comunicazione in uscita da un neurone. Ogni cellula nervosa ne possiede di norma soltanto uno;
- i **dendriti**: la principale via di comunicazione in ingresso; sono multipli per ogni neurone, formando il cosiddetto albero dendritico;
- le **sinapsi**, o giunzioni sinaptiche: i siti funzionali ad alta specializzazione nei quali avviene il passaggio delle informazioni fra neuroni. Ognuno di questi ne possiede migliaia. A seconda dell'azione esercitata dai neurotrasmettitori, le sinapsi hanno una funzione eccitatoria, facilitando la **trasmissione dell'impulso nervoso**, oppure inibitoria, tendente a smorzarlo. Le connessioni hanno luogo quando il neurotrasmettitore viene rilasciato nello spazio intersinaptico, raggiungendo così i recettori delle membrane post-sinaptiche (ossia del neurone successivo), e, alterandone la permeabilità, trasmettendo l'impulso nervoso.

Un singolo neurone può **ricevere simultaneamente segnali da diverse sinapsi**. Una sua capacità intrinseca è quella di misurare il potenziale elettrico di tali segnali in modo globale, stabilendo quindi se è stata raggiunta la soglia di attivazione per generare a sua volta un impulso nervoso. **Tale proprietà è implementata anche nelle reti artificiali.**

La configurazione sinaptica all'interno di ogni **rete neurale biologica** è dinamica. Si tratta di un fattore determinante per la loro efficienza.
Il numero di sinapsi può incrementare o diminuire a seconda degli stimoli che riceve la rete. Più sono numerosi, maggiori connessioni sinaptiche vengono create, e viceversa.

In questo modo, **la risposta adattiva fornita dai circuiti neurali è più calibrata**, e anche questa è una peculiarità implementata nelle reti neurali artificiali.

Grit

RETI NEURALI

Algoritmi di apprendimento:
Nelle reti artificiali ovviamente il processo di apprendimento automatico è semplificato rispetto a quello delle reti biologiche.
Non esistono analoghi dei neurotrasmettitori, ma lo schema di funzionamento è simile.
I nodi ricevono dati in input, li processano e sono in grado di inviare le informazioni ad altri neuroni. Attraverso cicli più o meno numerosi di input-elaborazione-output, in cui gli input presentano variabili differenti, diventano in grado di generalizzare e fornire output corretti associati ad input non facenti parte del training set.
Gli **algoritmi di apprendimento** utilizzati per istruire le reti neurali sono divisi in 3 categorie. La scelta di quale usare dipende dal campo di applicazione per cui la rete è progettata e dalla sua tipologia (feedforward o feedback).
Gli algoritmi sono:

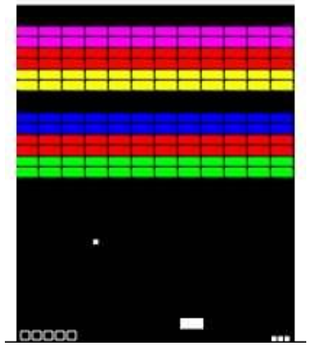
- supervisionato
- non supervisionato
- di rinforzo

Apprendimento supervisionato
Nell'**apprendimento supervisionato** si fornisce alla rete un insieme di input ai quali corrispondono output noti (training set). Analizzandoli, la rete apprende il nesso che li unisce. In tal modo impara a generalizzare, ossia a **calcolare nuove associazioni corrette input-output** processando input esterni al training set. Man mano che la macchina elabora output, si procede a correggerla per migliorarne le risposte variando i pesi. Ovviamente, aumentano i pesi che determinano gli output corretti e diminuiscono quelli che generano valori non validi.
Il meccanismo di apprendimento supervisionato impiega quindi l'**Error Back-Propagation**, ma è molto importante l'esperienza dell'operatore che istruisce la rete. Il motivo risiede nel non facile compito di trovare un rapporto adeguato fra le dimensioni del training set, quelle della rete e l'abilità a generalizzare che si tenta di ottenere.
Un numero eccessivo di parametri in ingresso e una troppo potente capacità di elaborazione, paradossalmente, rendono difficile alla **rete neurale** imparare a generalizzare, perché gli input esterni al training set vengono valutati dalla rete come troppo dissimili ai sofisticati e dettagliati modelli che conosce.
D'altro canto, un training set con variabili scarse porta per la via opposta alla stessa conclusione: la rete, in questo caso, non ha sufficienti parametri per apprendere a generalizzare.
Il giusto compromesso, insomma, è un compito che necessita di **molta preparazione ed esperienza**.
Le reti feedforward come il MLP utilizzano l'apprendimento supervisionato.

RETI NEURALI

Apprendimento non supervisionato
In una rete neurale ad **apprendimento non supervisionato**, la medesima riceve solo un insieme di variabili di input. Analizzandole, la rete deve creare dei cluster rappresentativi per categorizzarle. Anche in questo caso i valori dei pesi è dinamico, ma sono i nodi stessi a modificarli.
Esempi di reti ad apprendimento supervisionato sono **SOM** e la **rete di Hopfield.**

Apprendimento per rinforzo
Nelle reti neurali che apprendono mediante l'**algoritmo di rinforzo**, non esistono né associazioni input-output di esempi, né un aggiustamento esplicito degli output da ottimizzare.
I circuiti neurali imparano esclusivamente dall'interazione con l'ambiente. Su di esso, eseguono una serie di azioni.
Dato un risultato da ottenere, è considerato rinforzo l'azione che avvicina al risultato; viceversa, la rete apprende a eliminare le azioni negative, ossia foriere di errore.
Detto in altri termini, **un algoritmo di apprendimento per rinforzo mira a indirizzare la rete neurale verso il risultato sperato con una politica di incentivi (azioni positive) e disincentivi (azioni negative)**.



Usando tale algoritmo, una macchina impara a trovare soluzioni che non è esagerato definire creative. Una rete neurale così implementata, per esempio, è stata utilizzata per giocare ad **Arcade Breakout**. Risultato: dopo sole 4 ore di continuo miglioramento, i circuiti hanno individuato una strategia di gioco mai ideata da un essere umano in questo videogame.

Grit