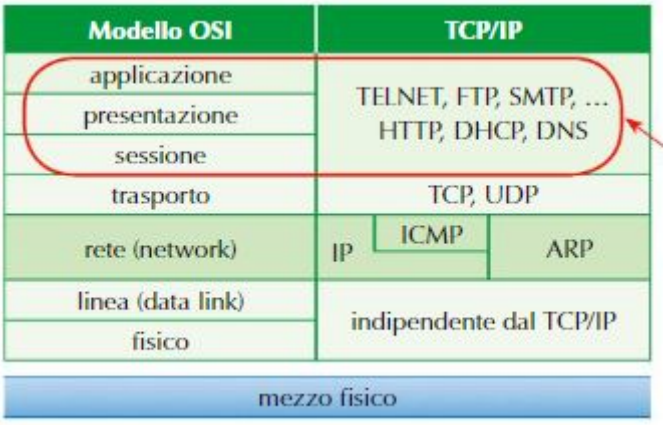
Il livello applicazione e il protocollo HTTP

Il livello applicazione



Il **livello applicazione** è quello più vicino all'utente e agisce da interfaccia tra le applicazioni software e i livelli sottostanti che permettono il transito dei messaggi in rete.

I **protocolli** di livello applicazione sono: **HTTP, DNS, FTP, IMAP, POP, SMTP** e **DHCP**.
In particolare il funzionamento delle applicazioni Web è basato sul protocollo HTTP definito in RFC 2616 che stabilisce le regole di comunicazione tra il client e il server.
HTTP trasmette i dati in **ASCII** servendosi dei protocolli IP, il cui compito è quello di instaurare e mantenere la comunicazione.
Affinché la richiesta del client vada a buon fine è necessario un applicazione sul server, chiamata anche "**demone**" o "**HTTPd**". Quando la richiesta arriva, il demone cerca di soddisfarla inviando al browser i documenti richiesti o la pagina.

Architettura client-server

L'architettura di un'applicazione di rete è basata sullo scambio di informazioni tra client e server. Nel caso di applicazione web, il client fa la richiesta di una risorsa al server, che risponde con un file HTML che il browser interpreta e visualizza.
Il percorso necessario a individuare le informazioni è specificato nell'URL. Il client può essere un qualunque browser e il server coinvolto è un server Web.
La pagina web visualizzata può essere di due tipi:

1. Il file HTML è già presente nella memoria del server che si occupa di recuperarlo e inviarlo al client ➜ **Pagina Web statica**
2. Il file HTML è generato dall'esecuzione di un programma di script eseguito sul server. La sua esecuzione genera un codice HTML che viene inviato al client (la pagina può essere diversa di volta in volta, a seconda dei parametri passati) ➜ **Pagina Web dinamica**

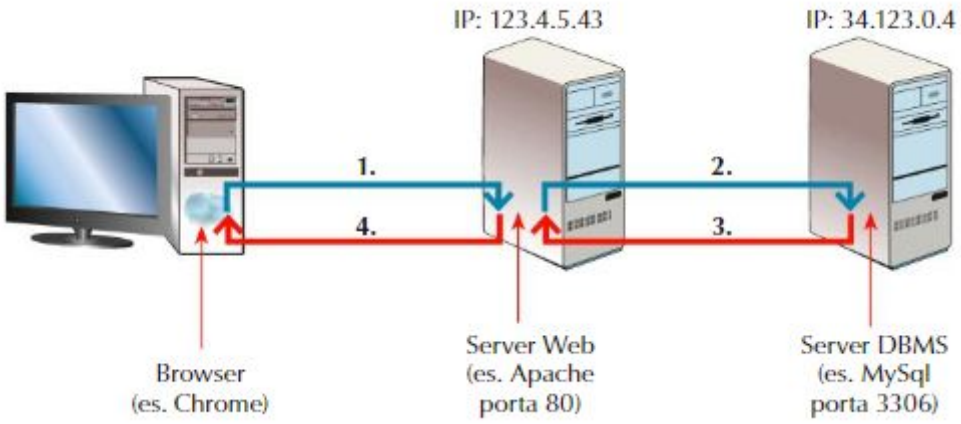
Architettura multi-tier

Uno dei motivi più frequenti per cui viene usato uno script *server side* è per eseguire l'accesso ai dati.
Una completa architettura di un'applicazione Web si configura su tre livelli (**three tier**).

1. Interfaccia utente
2. Logica di funzionamento
3. Accesso ai dati

Per **architettura multi-tier** si intende il modo in cui è suddiviso e organizzato il software che costituisce un'intera applicazione. Il software è diviso in livelli che comunicano tra loro.
Una completa architettura di un'applicazione Web, prevede che:

- Il client chiede una pagina al server
- Il server Web chieda i dati (tramite **codice SQL**) a un DBMS server
- Il DBMS server esegua il codice SQL e fornisca i dati al server Web
- Il server generi il codice HTML e lo invia al client
- Il client lo visualizzi sul dispositivo dell'utente.



Il browser svolge il ruolo di client nei confronti del server Web e il server Web svolge il ruolo di client nei confronti del DBMS server.
In un architettura multi-tier, i livelli adiacenti svolgono il ruolo di client e server.

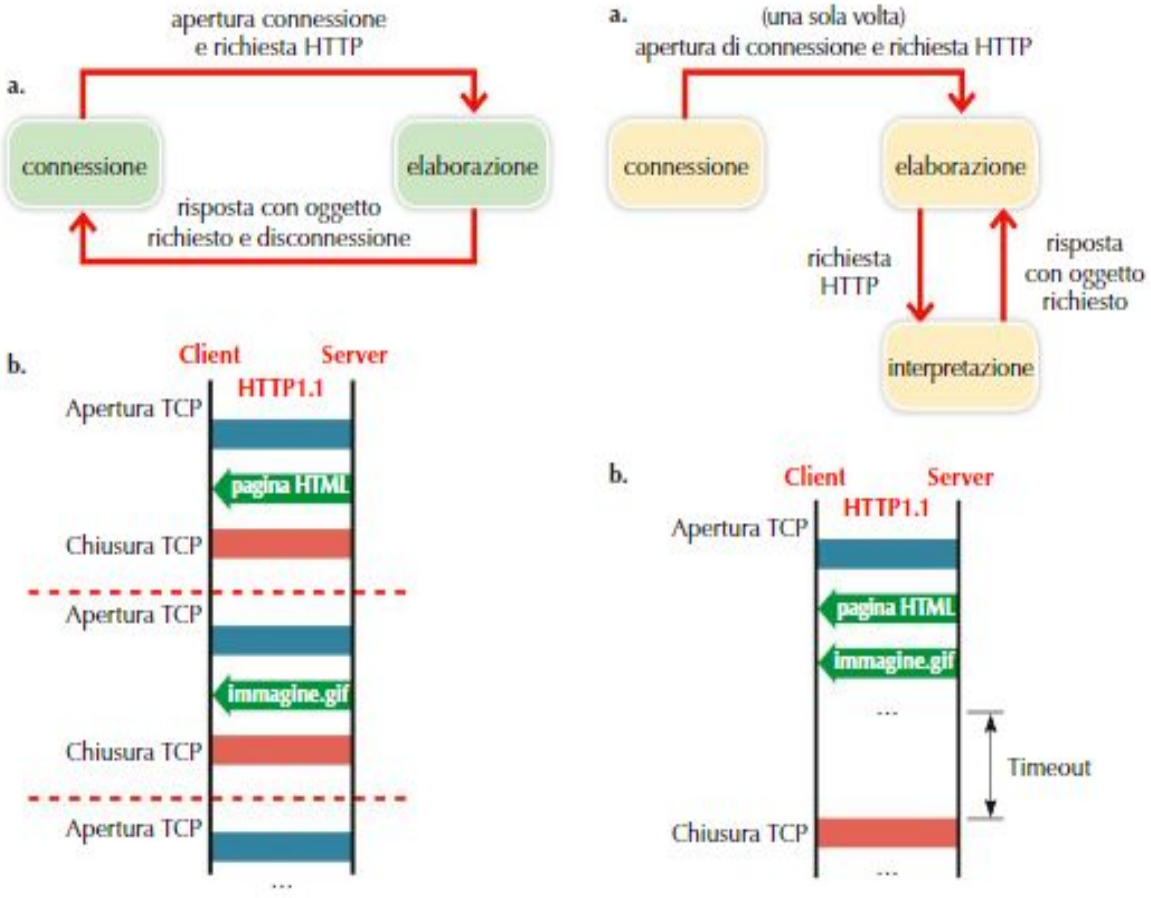
Il protocollo HTTP

Il protocollo HTTP definisce le regole per lo scambio di messaggi tra client e server Web e viene attivato ogni volta che richiamiamo una pagina Web da browser.
Quando il client invia al server una **richiesta**, il demone (HTTPd) presente sul server di destinazione riceverà il messaggio, lo interpreterà e fornirà una **risposta** che sarà interpretata dal client.

IL LIVELLO APPLICATIVO

Sintassi richiesta HTTP

- **HTTP non ha memoria** (è *stateless*): ogni richiesta HTTP è trattata dal server come unica. Ogni immagine o altro oggetto contenuto nel file HTML è richiamata perchè HTTP non ha la nozione di "sessione". e la storia di un utente non può basarsi su HTTP. Questi problemi vengono evitati con i "cookies" e le "sessioni" che memorizzano sul client o sul server, le informazioni necessarie per tenere traccia delle richieste.



- **HTTP è asimmetrico** (*pull based*): è solo il client che chiama il server.
- **HTTP non è un protocollo sicuro**. Le richieste che un browser invia al server possono essere facilmente intercettate e lette. La soluzione a questo problema è **HTTPS** (**Secure HTTP**) che viene utilizzato allo scopo di proteggere i dati tra client e server HTTPS, usa gli stessi metodi del protocollo HTTP, ma fa in modo che richieste e risposte siano crittografate tramite SSL.

Header HTTP

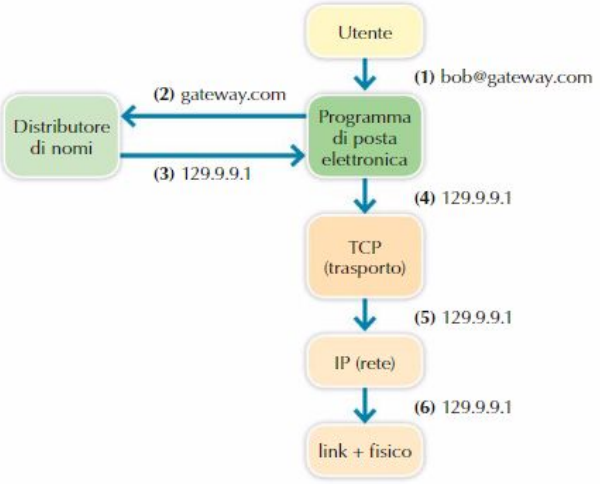
Il vostro browser, ogni volta che si collega ad un web server per aprire una pagina, invia specifiche richieste ed il server gli ritorna delle risposte. Questi messaggi chiamati header HTTP, contengono delle informazioni, alcune necessarie ed altre puramente informative.
Possono essere di diversi tipi, e ognuno di loro si occupa di un compito particolare:

- **General header ➜** Specificano come gestire un messaggio
- **Request header ➜** Specificano quali risorse sono richieste
- **Response header ➜** Specificano informazioni su chi deve rispondere alla richiesta
- **Entity header ➜** Specificano informazioni riguardanti il contenuto del messaggio

Il protocollo DNS

DNS (Domain name system) è un protocollo il compito fondamentale è quello di trasformare il nome di un dominio (es. www.pornhub.com) nel corrispondente indirizzo IP.

I nomi dei server



Un server in Internet è identificato da un indirizzo IP. Per tradurre i nomi dei server usati dagli utenti nei corrispondenti indirizzi. Occorre utilizzare un **sistema di risoluzione dei nomi**, cioè, una procedura che, a fronte di un nome restituisca un valore. Il **DNS** svolge questo compito.
Il DNS comprende un grande archivio distribuito che, oltre a risolvere i nomi dei domini, fornisce:

- Uno **spazio di nomi** strutturato in modo gerarchico che garantisce che il nome sia unico in internet.
- Un **servizio** in cui molti sistemi collaborano in rete per rendere disponibile lo spazio dei nomi.
- Un servizio in cui **ogni server DNS è responsabile solo del suo sottoinsieme.**
- Un **protocollo** a livello applicazione che, si appoggia alla porta 53 di UDP per il trasporto dei dati.

Quando deve essere stabilita una connessione, il programma client "*Resolver DNS*", chiamato da un browser o da un programma di posta elettronica, invoca il DNS che, a

fronte del nome del server da contattare, restituisce l'indirizzo IP corrispondente, con il quale viene stabilita una connessione a livello trasporto.

Gerarchia di dominio

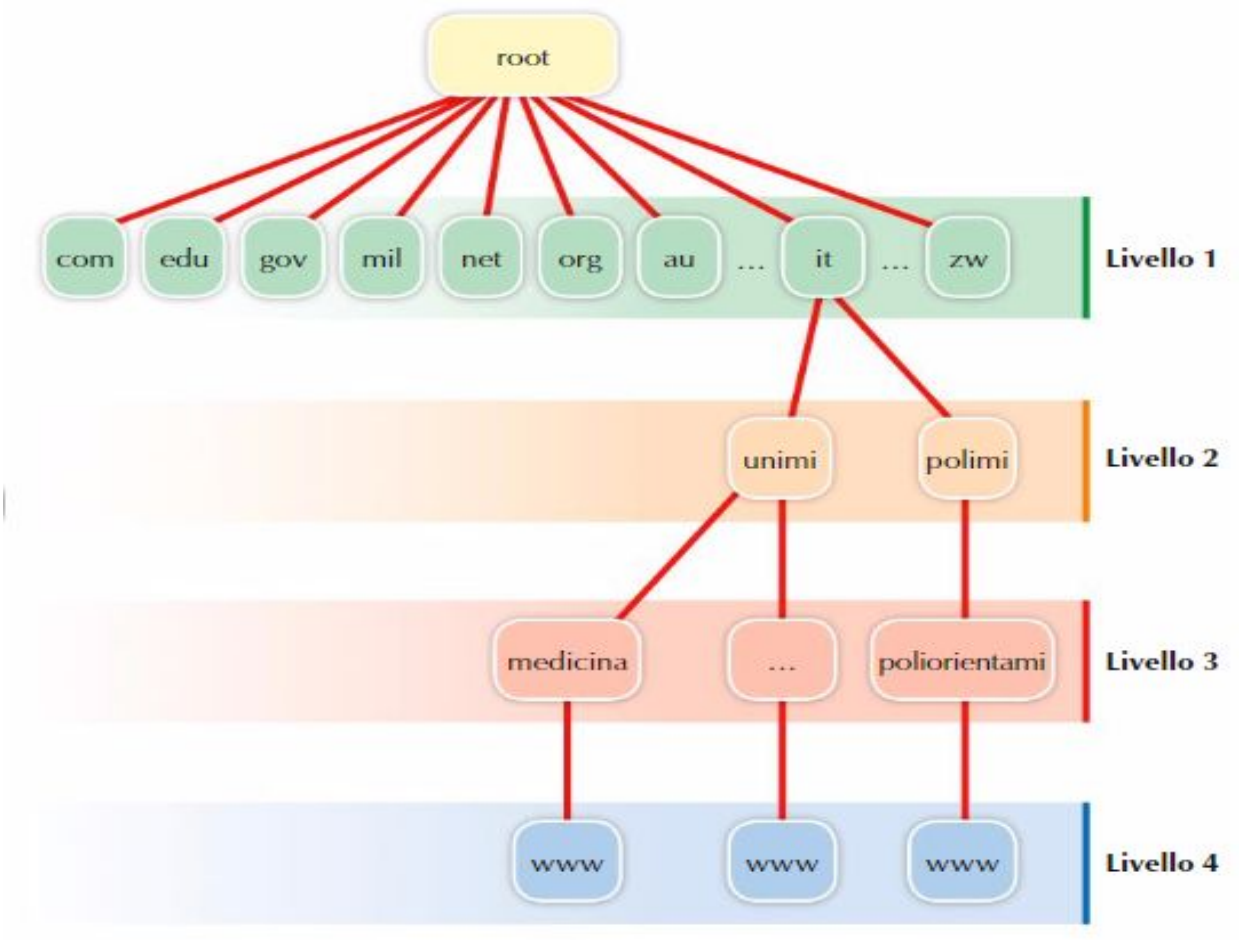
Il DNS assegna uno **spazio gerarchico dei nomi** alle risorse presenti in Internet.
Questa gerarchia può essere visualizzata come un albero in cui a ciascun nodo corrisponde un **dominio** e alle foglie corrispondono i calcolatori, ai quali viene associato un nome.
La gerarchia non è molto estesa. Al primo livello esiste un dominio per ogni Paese a cui si aggiungono i domini generici (.edu, .com, .gov, .org, .net).
Ogni dominio è costituito da una o più etichette separate da un "." e ogni nome di dominio può contenere un massimo di 255 caratteri.
Le etichette sono scritte da destra verso sinistra: l'etichetta a destra è il dominio di primo livello (**TLD**. *Top Level Domain),* a cui seguono i seguenti livelli di dominio.

ES:
www.pornhub.com

- **.com** ➜ Nome del dominio di primo livello
- **.pornhub.** ➜ Nome del dominio di secondo livello
- **www.** ➜ Nome del server.

I domini di secondo livello possono essere classificati secondo due categorie: **Generici** (*generic TLD*), e **nazionali** (*ccTLD*).

IL LIVELLO APPLICATIVO



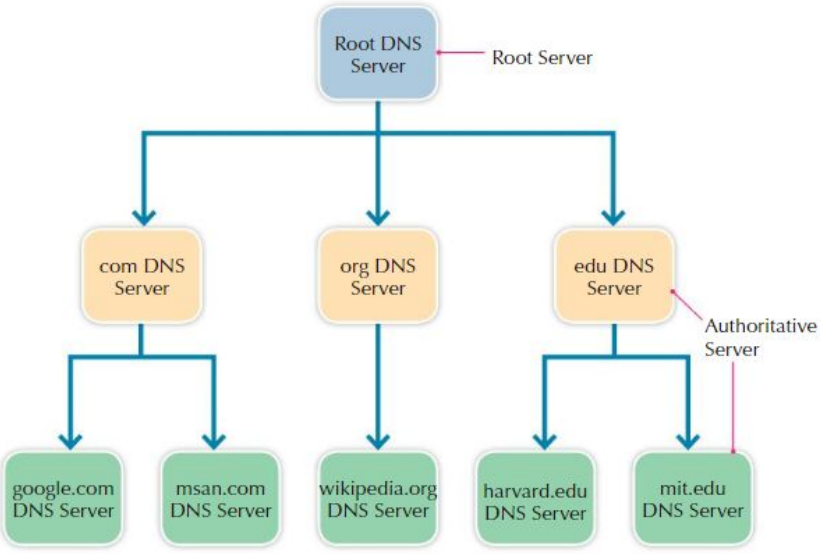
I domini gTLD si suddividono in **sTLD** (sponsored TLD) e **uTLD.** I domini ccTLD si riferiscono ai diversi paesi e alle loro abbreviazioni.

gTLD			
sTLD		uTLD	
aereo	per l'industria dei trasporti aerei	**com**	per le organizzazioni commerciali
asia	per la comunità dell'Asia	**net**	per le infrastrutture di rete
coop	per le cooperative	**org**	per le organizzazioni
jobs	per siti sull'impiego	**info**	per i siti informativi
edu	per siti di educazione scolastica superiore	**biz**	per business
gov	per siti governativi e le loro agenzie negli USA	**name**	per le famiglie e i singoli
mil	per le forze armate USA	**pro**	per alcune professioni
int	per le organizzazioni internazionali		

ccTLD	
it	Italia
uk	Regno Unito
fr	Francia
va	Città del Vaticano
tv	Tuvalu
us	Stati Uniti (raramente usato)
sm	San Marino
za	Sud Africa

Tab. 2 Esempi di domini nazionali.

Struttura e interrogazione del DNS



La struttura del DNS è una struttura ad albero alla cui radice stanno i **root name server** DNS. Questi server contengono l'elenco dei server responsabili (**authoritative server**) dei domini di primo livello (TLD) e lo forniscono come risposta alle richieste di risoluzione dei nomi riguardanti il dominio principale. I server di primo livello (.com, .org) delegano una parte di autorità ai server responsabili per i domini di secondo livello, che rispondono alle interrogazioni relative ai sottodomini e ai singoli host. Spesso i domini sono limitati a tre livelli, ma nulla vieta di averne di più.

Ogni ISP ha un proprio server dei nomi locali (local name server). La richiesta inviata da un host, per prima cosa, viene inoltrata a questo server, che solitamente mantiene una cache memory con i nomi più utilizzati. Se questo non riesce a soddisfare la richiesta, la inoltra a uno dei root name server, che restituirà l'elenco dei server che gestiscono il dominio interessato. Il server locale provvede allora a contattare uno di questi server per ottenere l'indirizzo IP richiesto

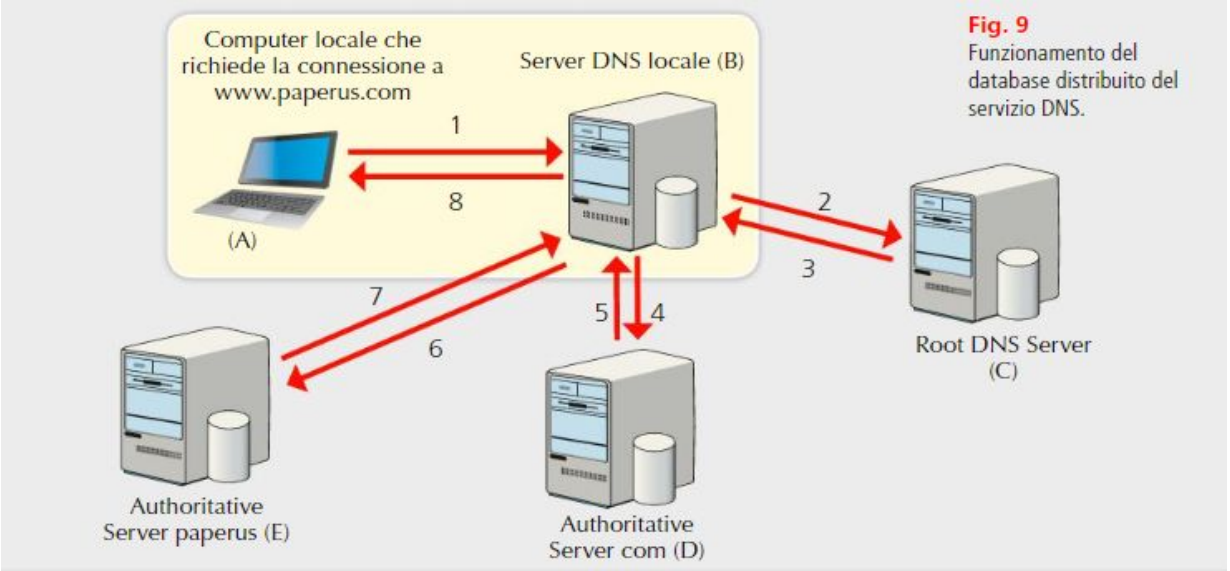
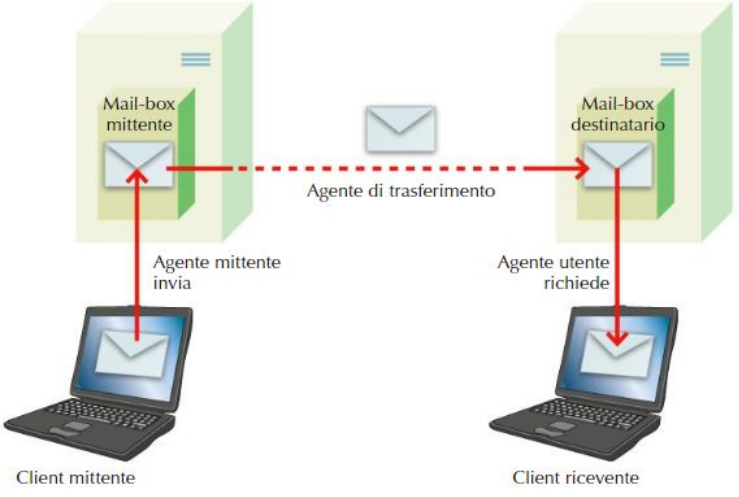


Fig. 9 Funzionamento del database distribuito del servizio DNS.

Le e-mail

La posta elettronica è un mezzo di comunicazione asincrono che, contrariamente a quanto accade per la chat o la videoconferenza, consente una comunicazione anche se mittente e destinatario non sono contemporaneamente connessi, Il sistema di posta elettronica è costituito da due entità:

- **Mail User Agent**
- **Mail Transfer Agent**



I **Mail User Agent** sono i client di posta che si occupano di creare, trasmettere e ricevere dei messaggi che passano dal computer dell'utente al server in cui risiede la casella di posta (mail-box). Si tratta di programmi che offrono all'utente tutti i servizi necessari a gestire la propria posta. Tra i più noti: *Outlook, Thunderbird…*

I **Mail Transfer Agent** trasferiscono i messaggi tra i server di posta con la funzione di ricevere e inoltrare i messaggi, e memorizzarli nelle mail-box degli utenti.

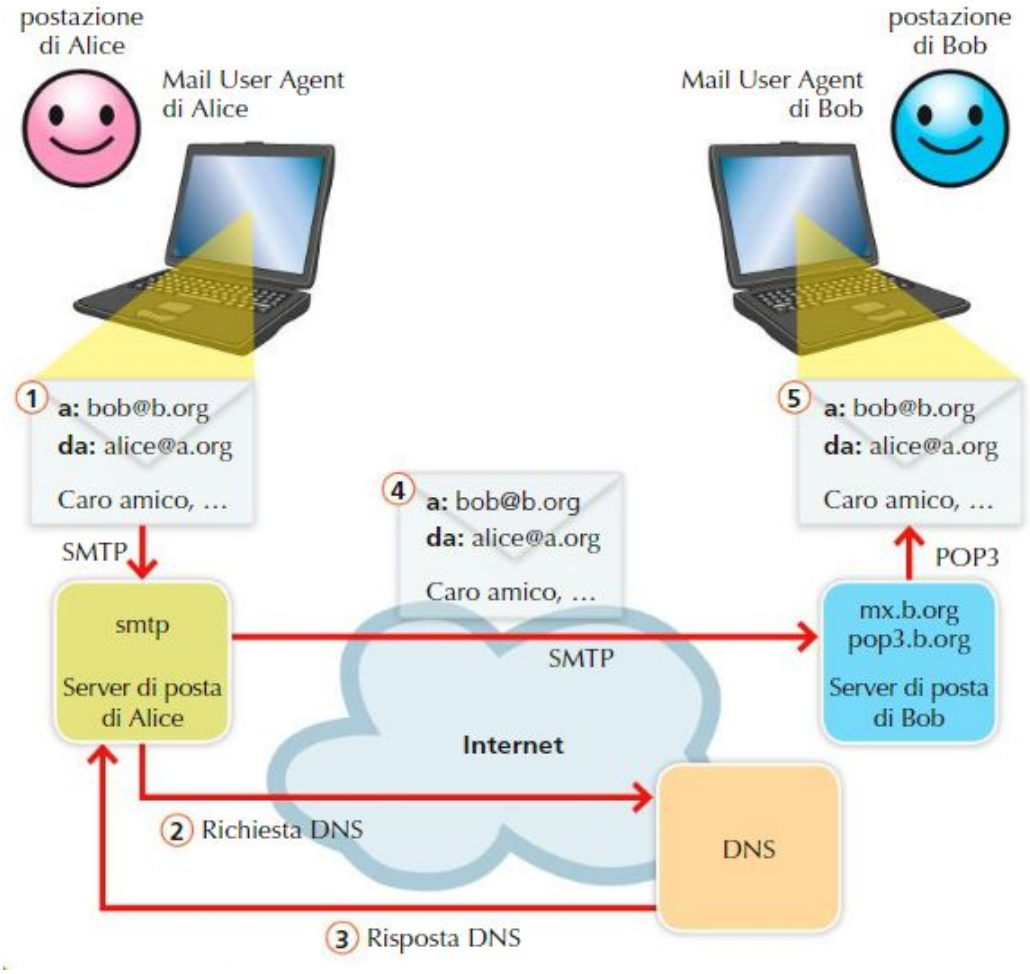


Fig. 11 Il percorso del messaggio di posta elettronica dal computer di Alice a quello di Bob, passando per i server di posta e sfruttando i protocolli SMTP e POP3.

MIME (Multiporpose Internet Mail Extensions)

Inizialmente un messaggio di posta consisteva di soli caratteri ASCII, ma c'era l'impossibilità di esprimere caratteri come le lettere accentate o di alfabeti non latini e di includere file multimediali.
Per questo venne proposto lo standard **MIME** che stabilisce una serie di regole per la codifica dei messaggi di posta, la struttura dei campi e il loro formato. L'idea alla base di MIME è quella di continuare a utilizzare la codifica base originaria in ASCII usando gli stessi protocolli. Vengono invece modificati i programmi di invio e ricezione del messaggio, aggiungendo una struttura al corpo del messaggio e definendo nuove regole per la codifica di messaggi non ASCII.

SMTP (Simple Mail Transfer Protocol)

Il trasferimento di e-mail è basato sul protocollo **SMTP**, che si occupa di **stabilire una connessione** tra una macchina mittente e una macchina destinazione e di **trasferire** il messaggio.
SMTP è un protocollo client/server, affidabile, che lavora sulla porta 25 di TCP. Quando la connessione viene stabilita, il client invia un messaggio al server, che lo inserisce nella casella di posta dell'utente e poi lo inoltra al server destinatario.
Il protocollo SMTP si articola in tre fasi:

1. Apertura della connessione o "handshaking"
2. Trasferimento del messaggio
3. Chiusura della connessione.

Protocollo SMTP con autenticazione

In origine il protocollo SMTP non prevedeva alcuna autenticazione da parte del mittente e ciò dava la possibilità di inviare e-mail con indirizzi falsi, facilitando lo spam e il phishing.
Per cercare di limitare queste operazioni è stata sviluppata un'estensione di SMTP in cui la fase di connessione al server di posta è preceduta da un'autenticazione dell'utente.

POP3 (Post Office Protocol 3)

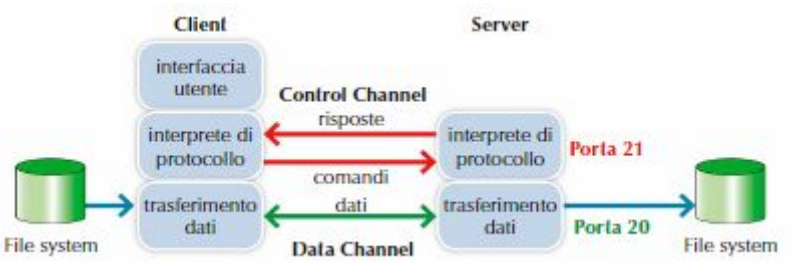
Dato che mittente e destinatario non sono necessariamente connessi simultaneamente, i messaggi vengono salvati sul server di posta del destinatario, in uno spazio di memoria a esso riservato. Quando il destinatario desidera accedere alla sua casella di posta deve collegarsi al server e accedere alla sua mail-box. Questo servizio è svolto da **POP3**.
POP3 si appoggia alla porta 110 di TCP. Come per gli altri protocolli, client e server comunicano scambiandosi comandi che costituiscono i messaggi.
Anche il protocollo POP3 si articola in tre fasi:

1. Apertura della connessione
2. Trasferimento del messaggio
3. Chiusura della connessione

Il trasferimento di messaggi comprende:

- **Autenticazione**: l'utente deve comunicare nome e password per poter accedere alla propria mail-box
- **Transazione**: il client ottiene i messaggi di posta, li segna per l'eliminazione ecc..
- **Aggiornamento**: il server elimina i messaggi precedentemente segnati e la connessione viene terminata.

FTP (File Transfer Protocol)



FTP è un protocollo di livello applicativo usato da un client per trasferire file su server remoto utilizzando due canali e due porte TCP: la prima per il trasferimento dati (porta 20), la seconda per l'invio dei comandi (porta 21).
Il suo scopo primario è la condivisione di file e il loro trasferimento in modo affidabile ed efficiente.
A differenza di HTTP, il protocollo FTP non mira all'apertura di pagine da visualizzare su browser, ma permette il collegamento a un server remoto con la possibilità di navigare nel suo file system e di fare tutte le operazioni che in genere vengono effettuate sui file del proprio computer locale (copia, cancella, sposta, invia, ricevi…)

Accesso FTP

Un server FTP è predisposto ad accettare le richieste di connessione fatte da un client, con il quale comunica tramite il protocollo FTP. Un client FTP, per connettersi al server, deve possedere i requisiti di accesso (login e password). A differenza di Telnet, che permette di eseguire comandi interattivi sul server remoto, FTP trasferisce (upload/download) e manipola i file.
I server a cui è possibile accedere in modo anonimo sono chiamati "*anonymous FTP servers*". La connessione in modalità anonima avviene digitando "*anonymous*" alla richiesta del login, senza immettere alcuna password. L'accesso anonimo pone dei limiti alle operazioni consentite: è possibile, per esempio, fare il download dei file ma non modificarne il contenuto.

Funzionamento del protocollo FTP

Una sessione FTP si articola in quattro fasi:

1. Il client effettua una richiesta a un server FTP in ascolto sulla porta 21
2. Il server effettua l'autenticazione del client
3. Il client effettua il trasferimento dei dati sulla porta 20
4. La connessione viene chiusa

Per distinguere il trasferimento dei comandi dal trasferimento dei file vengono eseguiti su due processi diversi

- **PI (Protocol Interpreter)** ➜ Con cui il client invia i comandi e riceve le risposte sulla porta 21.
- **DTP (Data Transfer Process)** ➜ In cui vengono scambiati i file sulla porta 20.

Il protocollo IMAP

Il **protocollo IMAP** (***Internet Message Access Protocol***), in particolare, consente al client di posta elettronica di accedere ad un server di posta gestendo la *sincronizzazione* tra i messaggi archiviati sul computer locale e il server di posta stesso. In pratica, l'IMAP permette al client di scaricare la posta dal server e quindi di accedere, leggere e cancellare le email attraverso altre macchine, sincronizzando con il server le operazioni effettuate.

Il funzionamento del protocollo IMAP

Ma come funziona il protocollo IMAP? Quando il client si connette al server di posta, IMAP controlla lo stato dei messaggi – se ci sono nuovi messaggi in arrivo, messaggi cancellati o email in bozza – senza scaricare le email, ovvero memorizzando i risultati in cache; se l'utente apre, cancella o archivia i messaggi, le modifiche vengono successivamente inviate al server. In questo modo, IMAP gestisce la sincronizzazione dei messaggi direttamente sul server.

Differenze con POP3

POP3 e IMAP svolgono sostanzialmente la stessa funzione, ovvero quella di gestire il flusso della posta elettronica in arrivo, ma lo fanno in maniera diversa. Come abbiamo visto, infatti, **IMAP gestisce la sincronizzazione dei messaggi dal server**, mentre POP3 scarica direttamente la posta in locale.

Vantaggi e svantaggi IMAP

La differente gestione della sincronizzazione dei messaggi di posta tra POP3 e **IMAP offre diversi vantaggi** a chi utilizza l'uno o l'altro protocollo. IMAP consente di eliminare i messaggi dal server senza doverli prima scaricare – e dunque di risparmiare spazio su disco – e di gestire accessi simultanei, rendendo accessibile la posta elettronica da qualunque dispositivo. POP3, però, scaricando i messaggi in locale li rende sempre accessibili, permettendo di leggerli anche senza connessione ad Internet. Quale scegliere tra POP3 e IMAP è una decisione che spetta a voi.