

# DNS (Domain Name System)

- È un **protocollo** di **livello applicativo**
- **Si appoggia** sul protocollo **UDP**
- Usa la **porta 53**
- I DNS sono un **file di testo**, **non** sono dei **database**
- Viene usato da altri protocolli applicativi (HTTP o FTP) per tradurre i nomi di domini nei loro rispettivi indirizzi IP
- **Paradigma client-server**

## Funzionalità fondamentali:

- **Risoluzione diretta:** traduce i **nomi** di **domini** nei loro rispettivi **indirizzi IP**.
- **Risoluzione inversa:** dato l'indirizzo IP, si restituisce il nome del dominio
- **Host aliasing:** dà un alias agli host che hanno un nome complicato
- **Distribuzione locale:** I siti con molto traffico, vengono replicati su più server, e ciascuno ha un indirizzo IP diverso.
- **Mail server aliasing:** per mandare una mail a un server di posta elettronica di cui non si conosce l'indirizzo IP, si fa una richiesta al server DNS del server di posta elettronica chiedendo il suo indirizzo IP per poi aprire la connessione e fargli arrivare la mail.

Avere un **sistema centralizzato** in cui il DNS contiene tutte le corrispondenze è **svantaggioso** per la **bassa scalabilità** nell'aumentare il numero di host:

- Si guasta il server DNS, ne soffre l'intera Internet
- Sarebbe solo il server DNS a gestire tutte le richieste
- Il server DNS si trova distante dagli host
- Manutenzione costante

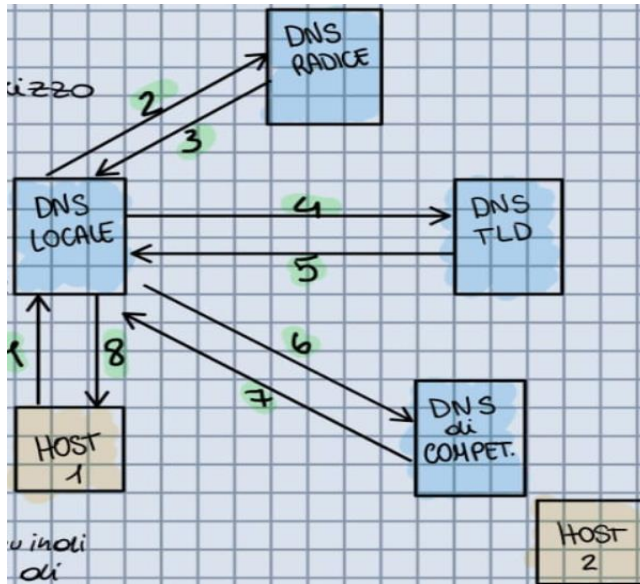
## Risoluzione del problema della scalabilità:

Tutte le corrispondenze vengono distribuite in diversi server DNS organizzati in un **sistema distribuito** (o **gerarchico**). Esistono 3 classi di server DNS:

- **Server radice:** in Internet esistono 13 di questa classe (etichettati dalla A-M)
- **Server TLD (Top Level Domain):** si occupano dei domini di alto livello (org, net, edu, gov e net) e di tutti i domini locali (it, uk, fr)
- **Server di competenza (authoritative name server):** sono autoritativi su un dominio, cioè un server ha la responsabilità di risolvere i nomi di un dominio

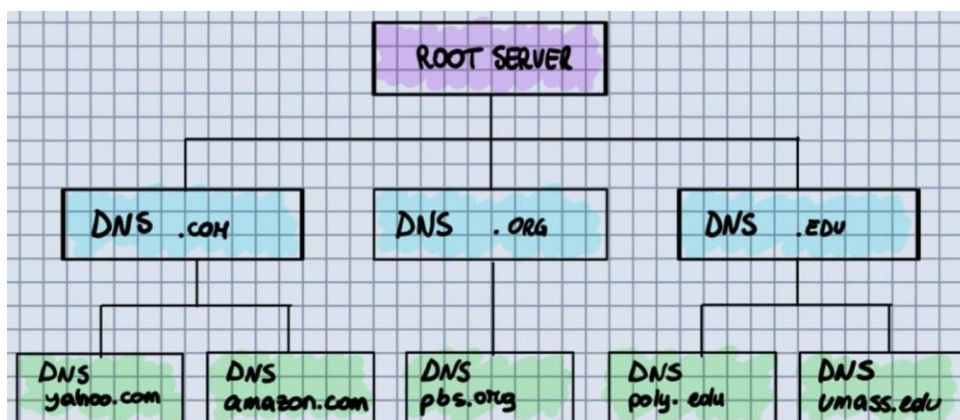
Esiste anche il **server locale DNS** che non appartiene strettamente alla gerarchia.

## Come funziona?



1. Un client deve accedere a [www.portaleargo.it](http://www.portaleargo.it) e non ha l'indirizzo IP, quindi lo chiederà al DNS locale.
2. Il DNS locale di norma non lo conosce, quindi inoltra il messaggio al server di radice.
3. Ma il server di radice non lo sa, però gli dà l'indirizzo IP di un server DNS TLD per il dominio .it
4. Viene fatta la richiesta dal DNS locale al DNS TLD di dominio .it.
5. Il DNS TLD risponde che non lo sa perché sotto il dominio .it ci sono tantissimi altri sottodomini. Ma questo server è obbligato a conoscere l'indirizzo IP del server DNS di competenza di portaleargo
6. Il server DNS locale fa la richiesta al server DNS di competenza
7. Il server DNS di competenza risolve il problema del nome del dominio mandando l'indirizzo IP al server DNS locale
8. Il server DNS locale manda l'indirizzo IP al client

Anche se questo esempio può succedere nella vita reale, risulta molto lento.



Schema fatto da Rizzy

Resource Record:

I resource record forniscono le corrispondenze tra nomi e indirizzo. Ogni messaggio DNS trasporta uno più resource record. I resource record è formato da:

- **Name e Value:** dipenderà dal tipo di resource record
- **Type:**

- A: risolve il nome di dominio per indirizzi IPv4

TYPE	NAME	IP ADDRESS	TTL
AAAA	example.com	2501:0:53b::3 330:c2f4	7200

- NS: fornisce il nome del server name (authoritative) all'interno di un dominio

TYPE	VALUE	NAME	TTL
NS	ns1.example.com	example.com	7200
NS	ns2.example.com	example.com	7200

- CNAME: associa un nome di dominio o di sottodominio a un altro nome di dominio (esempio.com www.esempio.com).

TYPE	NAME	ALIAS TO	TTL
CNAME	www.example.com	example.com	7200

- MX: specifica il server di posta responsabile per la ricezione delle email per un determinato dominio.

TYPE	PRIORITY	NAME	HOST	TTL
MX	10	example.com	mail1.example.com	7200

- **TTL (Time to Live):** tempo di vita del record resources

[DNS Records Explained](#)