# Lo strato Applicativo
# DNS

**Reti di Calcolatori**
**AA. 2023-2024**

Docente: Federica Paganelli
Dipartimento di Informatica
federica.paganelli@unipi.it

# Obiettivo: Identificare il processo

- Ogni processo

  di livello applicativo ha necessità di individuare il processo omologo con il quale vuole comunicare

- Il processo omologo

  risiede su una particolare macchina (remota), anch'essa da individuare (sappiamo che usa lo stesso protocollo!)

# Nomi e Indirizzi

- **Nome**
- un nome identifica un oggetto
  - identificatore che consiste di una sequenza di caratteri scelti da un alfabeto finito
- Es. nome di host o servizio
  - www.yahoo.com
  - di.unipi.it
- Identificativo di livello applicativo
  - Nome logico
  - Mnemonico (non sempre)

- **Indirizzo:** Identifica dove tale oggetto è situato
- I dispositivi connessi in rete vengono individuati mediante i loro indirizzi IP
- Indirizzi IP:
  - 4 byte (32 bit)
  - usato per instradare i datagrammi (livello di rete)
- Formato indirizzo IP progettato per garantire efficienza nell'instradamento

**`rever.nmsu.edu <-> 128.123.3.18`**

Q: Come associare indirizzo IP e nome?

# DNS: Domain Name System

- Inizialmente l'associazione tra nomi logici e indirizzi IP era statica:
  1. tutti i nomi logici e i relativi indirizzi IP erano contenuti in un file (host file)
  2. periodicamente tutti gli host prelevavano una versione aggiornata del file (master host file) da un server ufficiale
- Date le dimensione attuali di Internet, questo approccio è impraticabile
  - Non è possibile che ogni host abbia una copia (aggiornata) di un elenco del genere [come agli inizi delle reti]
    - dimensione elenco, volume traffico …
  - Non è possibile neppure centralizzare un elenco del genere
    - unico punto di fallimento, volume di traffico, DB distante, … non scala!
- si utilizza pertanto il DNS (Domain Name System)

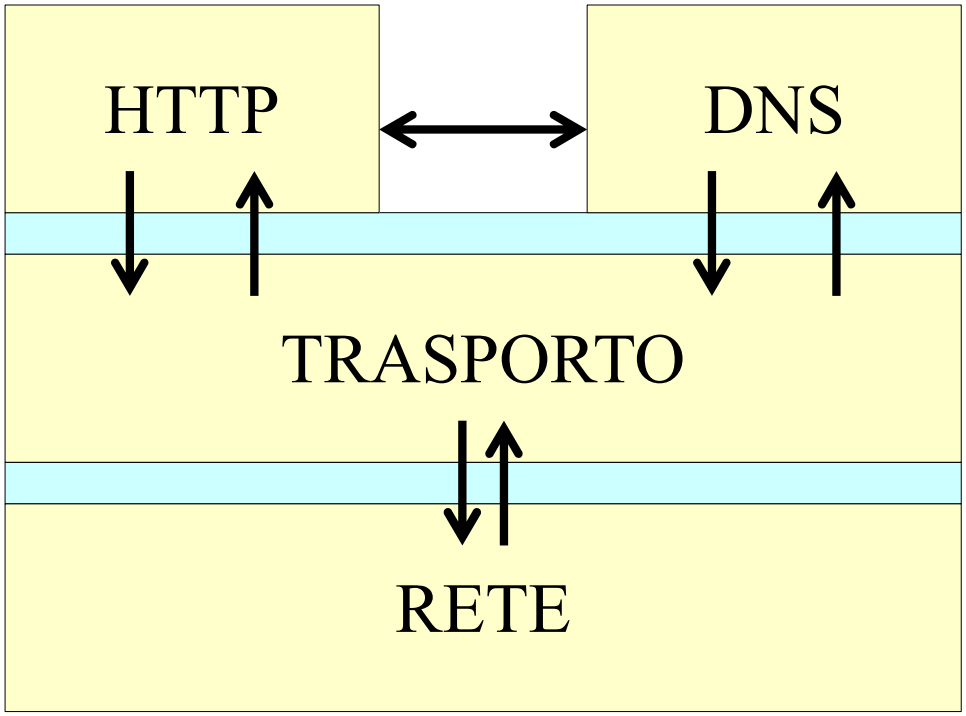
# Il servizio di risoluzione

Il DNS è posizionato nel livello applicativo

- gira su sistemi terminali, adotta il paradigma client-server
- si affida al sottostante protocollo di trasporto punto-punto per trasferire messaggi tra sistemi terminali

N.B.1 Non interagisce direttamente con gli utenti

N.B.2 Filosofia di Internet: complessità alle estremità della rete (confronto con reti di tlc)




una funzione fondamentale di Internet è implementata come protocollo dello strato applicazione

1. Clicca su
www.sss.com/contact.html

7. Mostra pagina p

DNS

2. www.sss.com ?

3. www.sss.com
=
128.123.6.22

Browser

4. c=TCPopen(128.123.6.22, 80)
5. TCPsend(c,"GET /contact.html")
6. p = TCPreceive(c)

TCP

# Domain Name System

- Il DNS è un meccanismo che deve:
  - specificare la sintassi dei nomi e le regole per gestirli
  - consentire la conversione da nomi a indirizzi e viceversa
- Esso è costituito essenzialmente da:
  1. uno **schema di assegnazione dei nomi** gerarchico e basato su domini
  2. un **database distribuito** contenente i nomi e le corrispondenze con gli indirizzi IP implementato con una gerarchia di name server
  3. un **protocollo** per la distribuzione delle informazioni sui nomi tra name server
     - host, name server comunicano per risolvere nomi (traduzione nome/indirizzo)
     - utilizzando UDP (porta 53) [oppure TCP]

# Servizi DNS

- Risoluzione di nomi di alto livello (hostname) in indirizzi IP
- Host aliasing
  - Un host può avere più nomi (nome canonico + sinonimi o alias)
  - Traduzione dei nomi in nome canonico/indirizzo IP
  - p.e. `relay1.west-coast.enterprise.com` (canonico) può avere due alias `enterprise.com` e `www.enterprise.com`
- Mail server aliasing
  - Sinonimi per mail server
    - Nome canonico: `relay1.west-coast.hotmail.com`
    - Alias `hotmail.com`
  - Esempio: nomi identici per mail server e web server
- Distribuzione carico
  - Distribuzione carico tra server replicati
  - Ad un hostname canonico corrispondono più indirizzi IP
  - Il DNS restituisce la lista di IP variandone l'ordinamento a ogni risposta

# SPAZIO DEI NOMI

# Struttura dei nomi -I

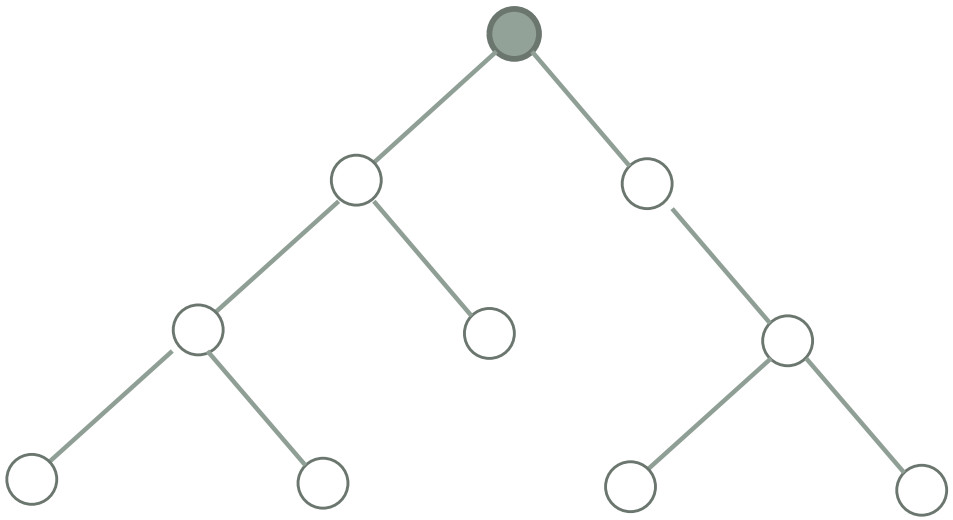
- Spazio dei nomi: deve permettere di identificare in modo univoco un host
- Struttura “flat” (piatta): sequenza di caratteri senza alcuna ulteriore struttura
  - adatta a reti con pochi elaboratori
  - Assegnazione dei nomi centralizzata

- Struttura gerarchica
  - Un nome è costituito da diverse parti
    - Nome organizzazione, dipartimento, ufficio, ecc..

    `lab3.di.unipi.it`
  - Assegnazione dei nomi delegabile -> sistema (in buona parte) decentralizzato
  - Delega dell'autorità per l’assegnazione delle varie parti dello spazio dei nomi (il nome deve essere univoco)
  - Distribuzione responsabilità della conversione tra nomi e indirizzi

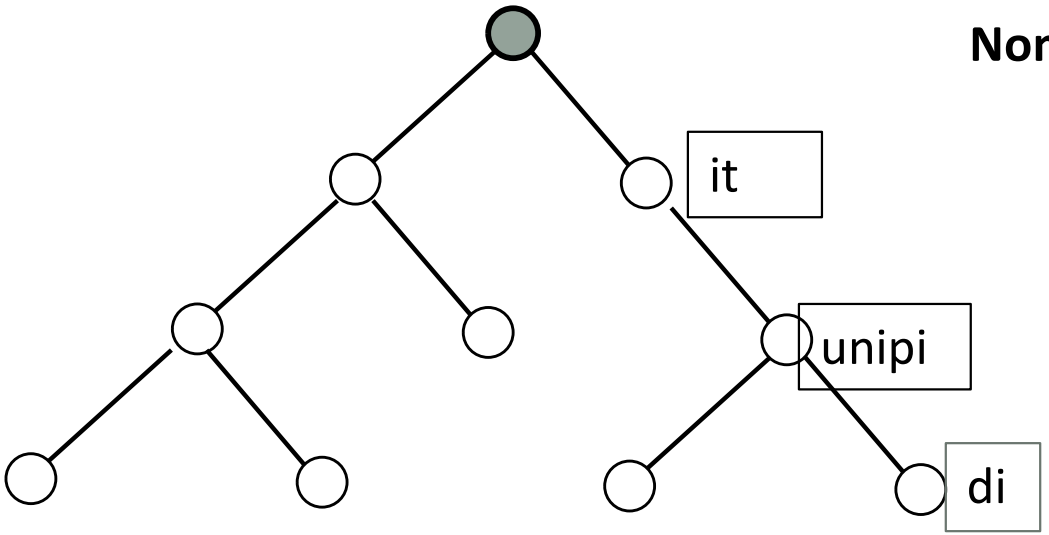
# Nomi di dominio

- Spazio dei nomi con struttura gerarchica
- I nomi hanno una struttura ad albero con un numero di livelli variabile
- Ogni nodo è individuato da un'etichetta (max 63 caratteri)
  - alla radice è associata un'etichetta vuota



# Nomi di dominio

- Ogni nodo dell'albero ha un nome di dominio
  - una sequenza di etichette separati da punti (.)
- DOMINIO: **sottoalbero nello spazio dei nomi di dominio che viene identificato dal nome di dominio del nodo radice del sottoalbero**.
- Un dominio può essere suddiviso in ulteriori domini, detti sottodomini

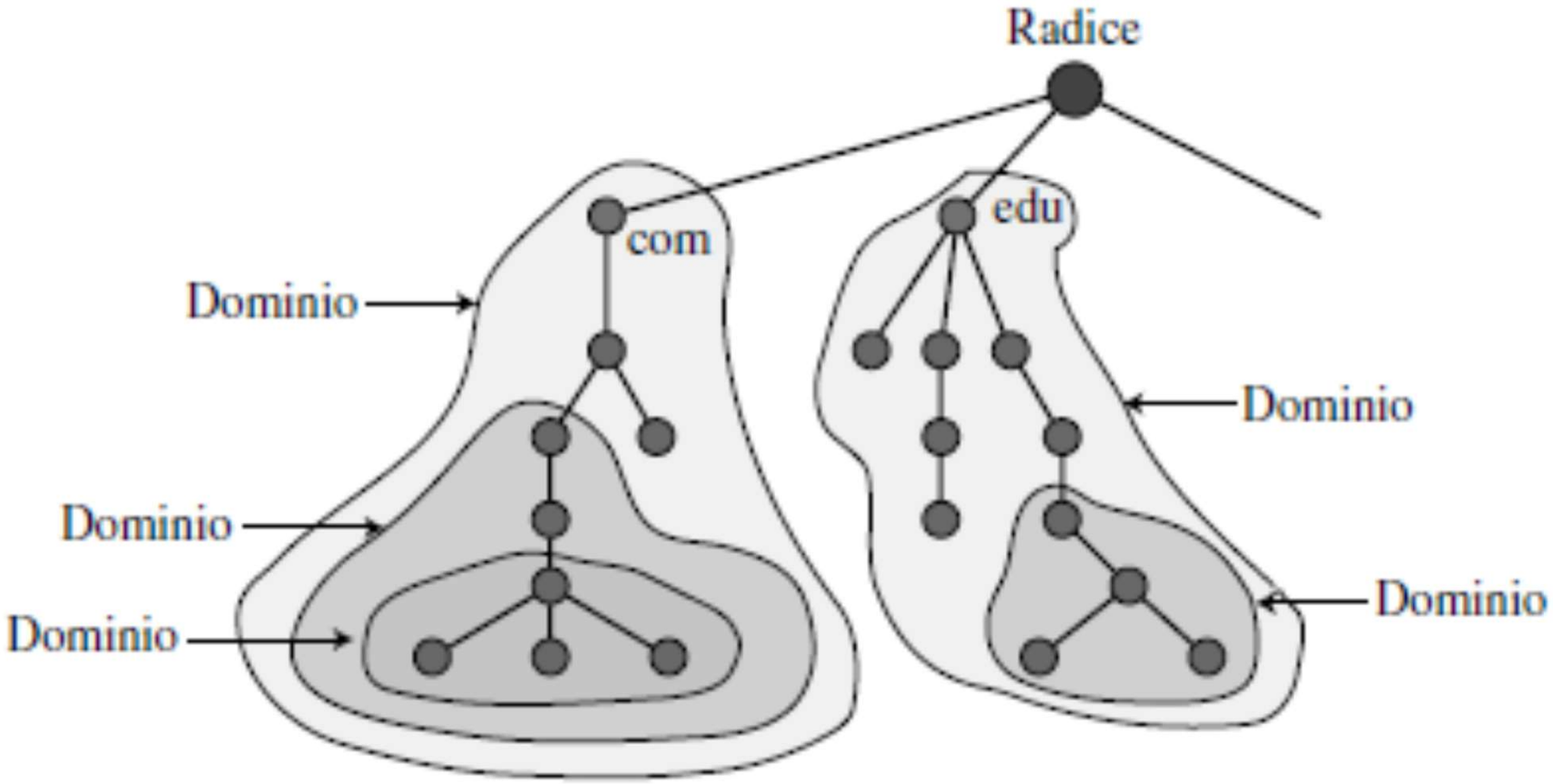



**Nomi di dominio**

| | |
|---|---|
| it | Dominio top-level |
| unipi.it | Dominio di secondo livello |
| di.unipi.it | Dominio di terzo livello |

# Domini

- Internet divisa in diverse centinaia di domini - ogni dominio partizionato in sotto-domini e così via



# Nomi di dominio

- In Internet i nomi gerarchici delle macchine sono assegnati in base alla **struttura delle organizzazioni** che ottengono **l'autorità** per porzioni dello spazio dei nomi
- La struttura gerarchica permette autonomia nella scelta dei nomi all'interno di un dominio (l'univocità è comunque garantita)
- Es. **server1**.di.unipi.it e **server1.**cs.cornell.edu sono due nomi diversi!

# Esempi di top-level domain

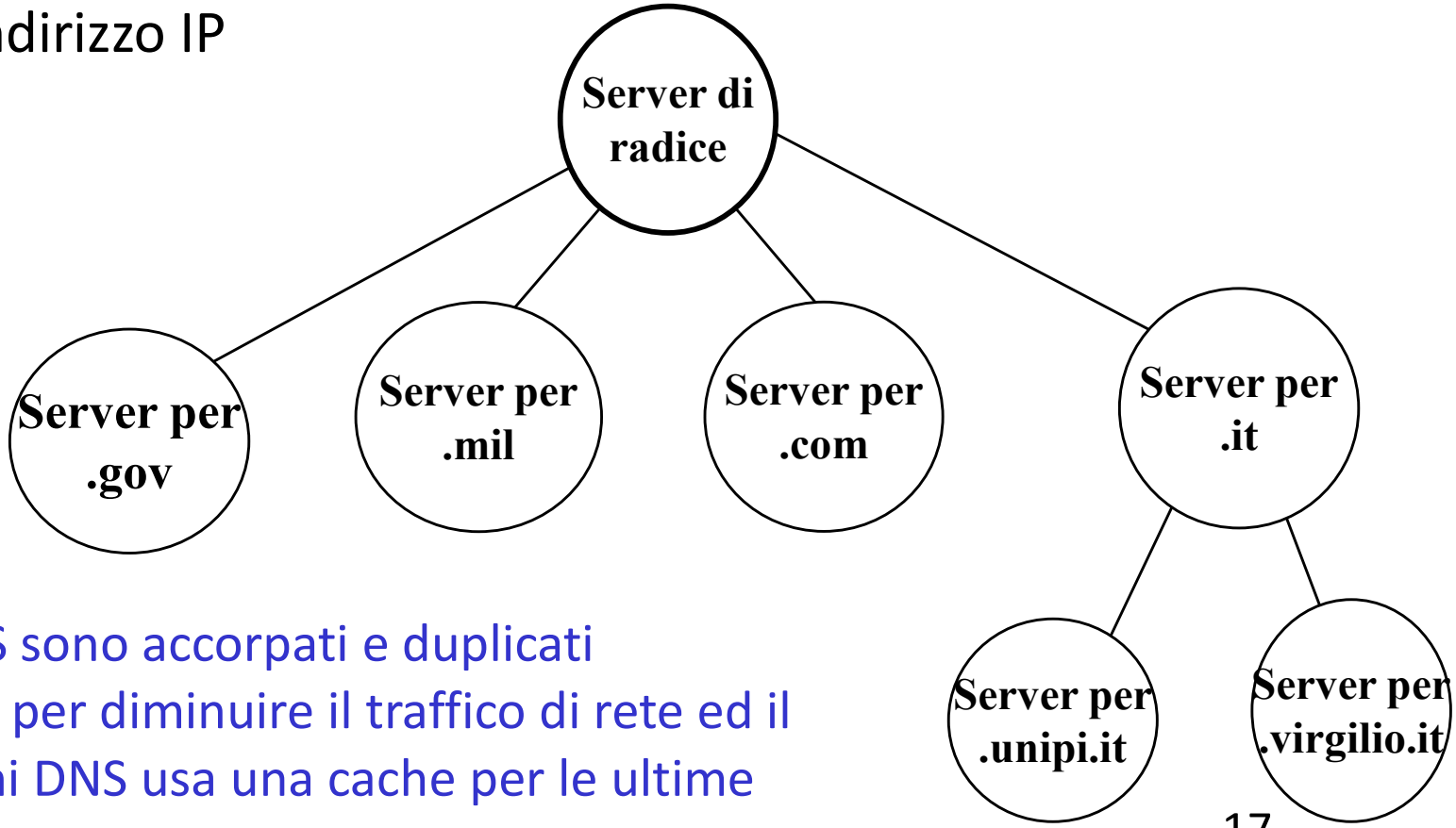
| | |
|---|---|
| com | Organizzazioni commerciali |
| edu | Istituti di istruzione (università, scuole) |
| mil | Gruppi militari |
| gov | Istituzioni governative (USA) |
| net | Principali centri di supporto alla rete |
| org | Organizzazioni diverse dalle precedenti |
| Codice geogr.<br>(ir, uk, us, fr, etc.) | Schema geografico per nazioni |

- Mantenuti da IANA (**Internet Assigned Numbers Authority (IANA)**
- https://data.iana.org/TLD/tlds-alpha-by-domain.txt

# GERARCHIA DEI NAME SERVER

# Name Servers

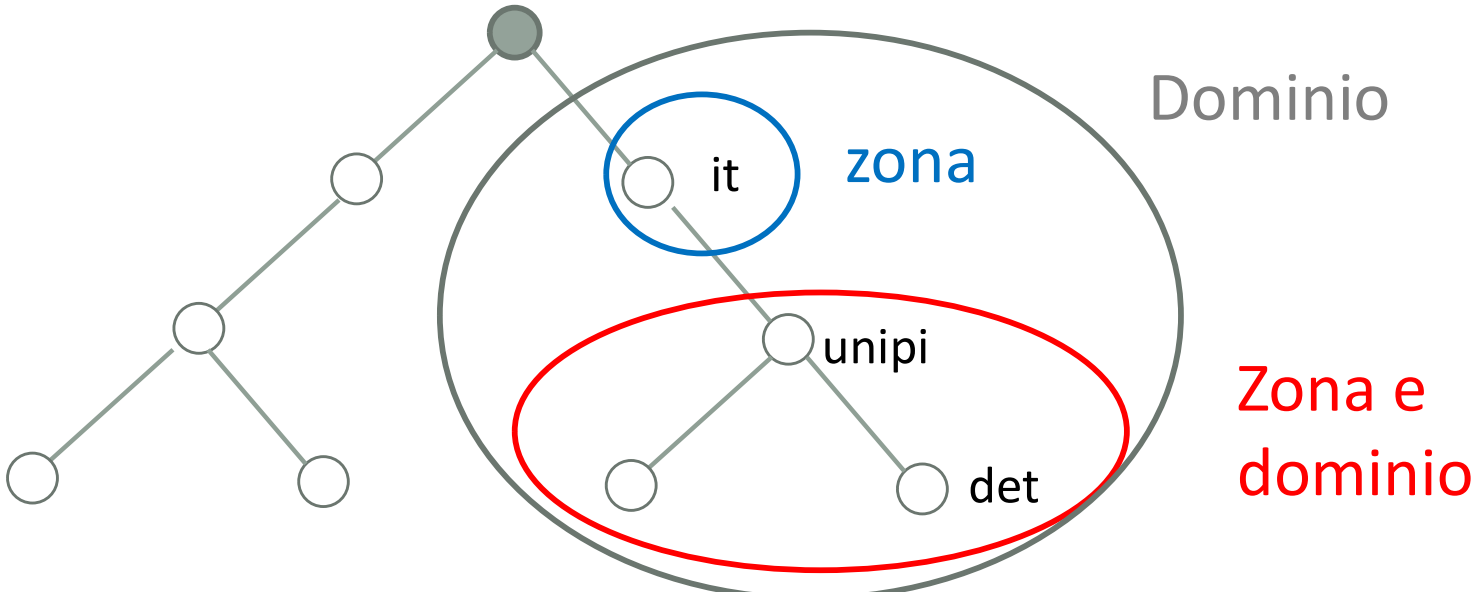
- DNS: Database distribuito implementato in una gerarchia di più name servers
- Name server: programma che gestisce la conversione da nome di dominio a indirizzo IP




Nota: spesso i DNS sono accorpati e duplicati (sicurezza); inoltre per diminuire il traffico di rete ed il carico dei DNS ogni DNS usa una cache per le ultime richieste espletate.

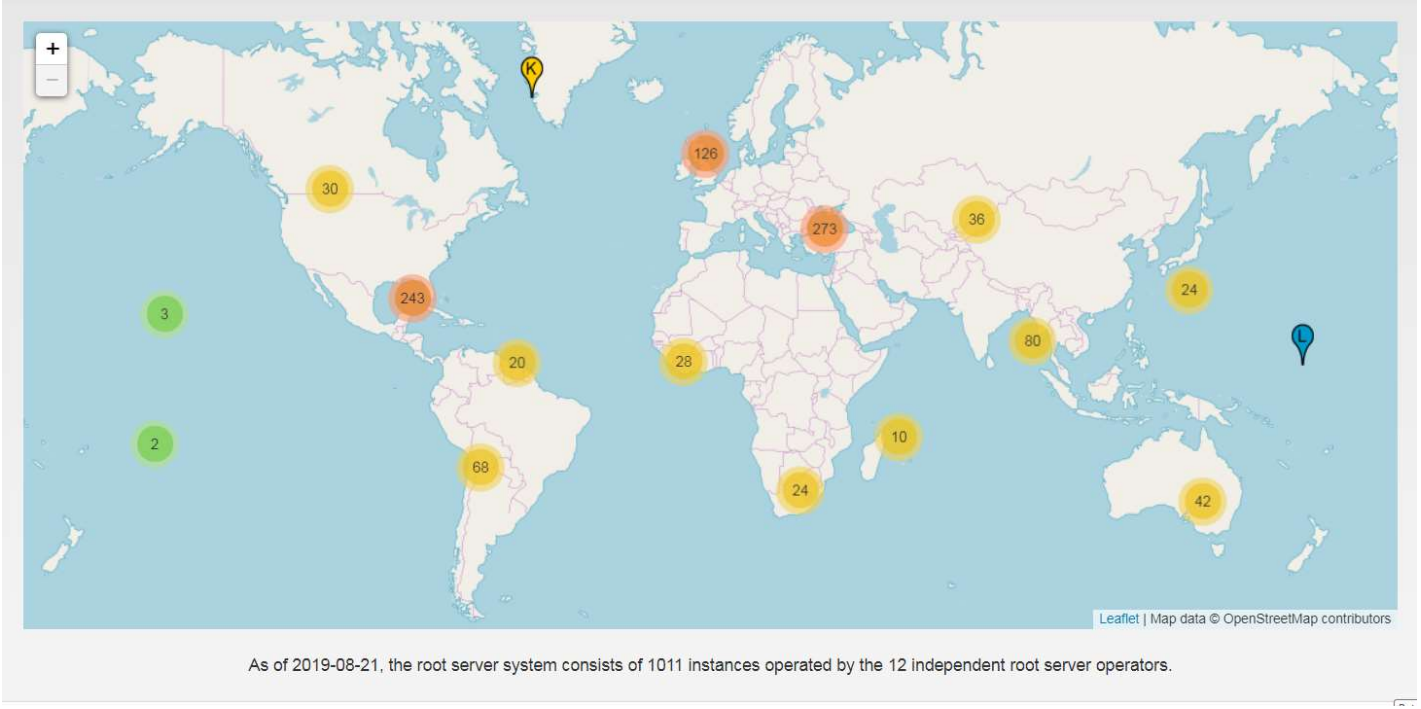
# Name servers

- Informazioni sui domini ripartite su più name server
- Zona: porzione dello spazio dei nomi di dominio che è gestita da una specifica amministrazione
  - Zona e dominio non coincidono necessariamente
- Il server immagazzina **le informazioni relative alla propria zona inclusi i riferimenti ai name server dei domini di livello inferiore**



# Gerarchia dei server

- Server radice (Root Name server)
  - Responsabile dei record della zona radice
    - server che riconosce tutti i domini di massimo livello e conosce il server che risolve ciascun dominio.
  - restituisce le informazioni sui name server di TLD
  - ~ centinaia di root name servers in tutto il mondo




http://www.root-servers.org/

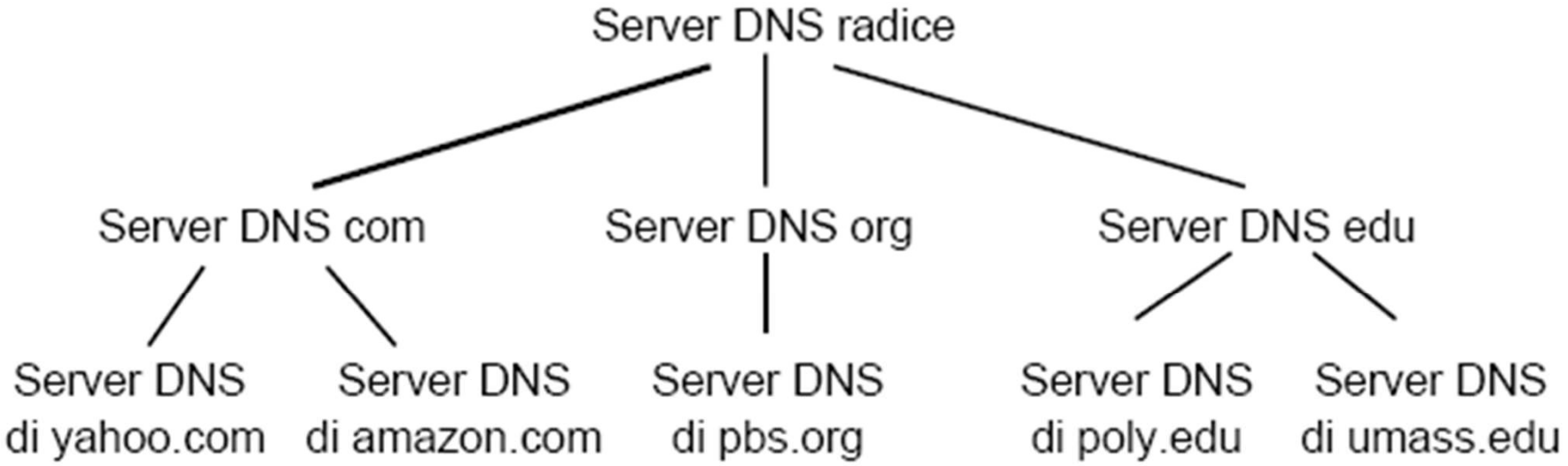
# Gerarchia dei server

- Server top-level domain:
  - mantiene le informazioni dei nomi di dominio che appartengono a un certo TLD
  - restituisce le informazioni sui name server di competenza dei sottodomini
- Server di competenza (authoritative name server):
  - Autorità per una certa zona
  - memorizza nome e indirizzo IP di un insieme di host
  - può effettuare traduzioni  nome/indirizzo per quegli host
  - Per una certa zona ci possono essere server di competenza primari e secondari
    - Server primari mantengono il file di zona
    - Server secondari ricevono il file di zona e offrono il servizio di risoluzione

# Local name server

- Quando un programma (es. browser) deve trasformare un nome in un indirizzo IP chiama un programma in locale detto resolver, passando il nome come parametro di ingresso.
- Se il resolver non ha l'associazione richiesta, interroga un name server di cui conosce l'IP (local name server)
- Il local name server cerca il nome nelle sue tabelle, se trova l'associazione restituisce l'indirizzo al resolver, altrimenti inoltra la query alla gerarchia DNS
- local name server
  - non appartiene strettamente alla gerarchia dei server
  - ogni ISP (università, società , ISP) ha il suo *(default) name server locale*
  - Le query DNS vengono prima rivolte al name server locale
    - Il server DNS locale opera da proxy e inoltra la query in una gerarchia di server DNS

# Risoluzione dei nomi




Il client vuole l'IP di www.amazon.com; 1ª approssimazione:

- Il client interroga il server radice per trovare il server DNS com
- Il client interroga il server DNS com per ottenere il server DNS amazon.com
- Il client interroga il server DNS amazon.com per ottenere l'indirizzo IP di www.amazon.com

# Esempio

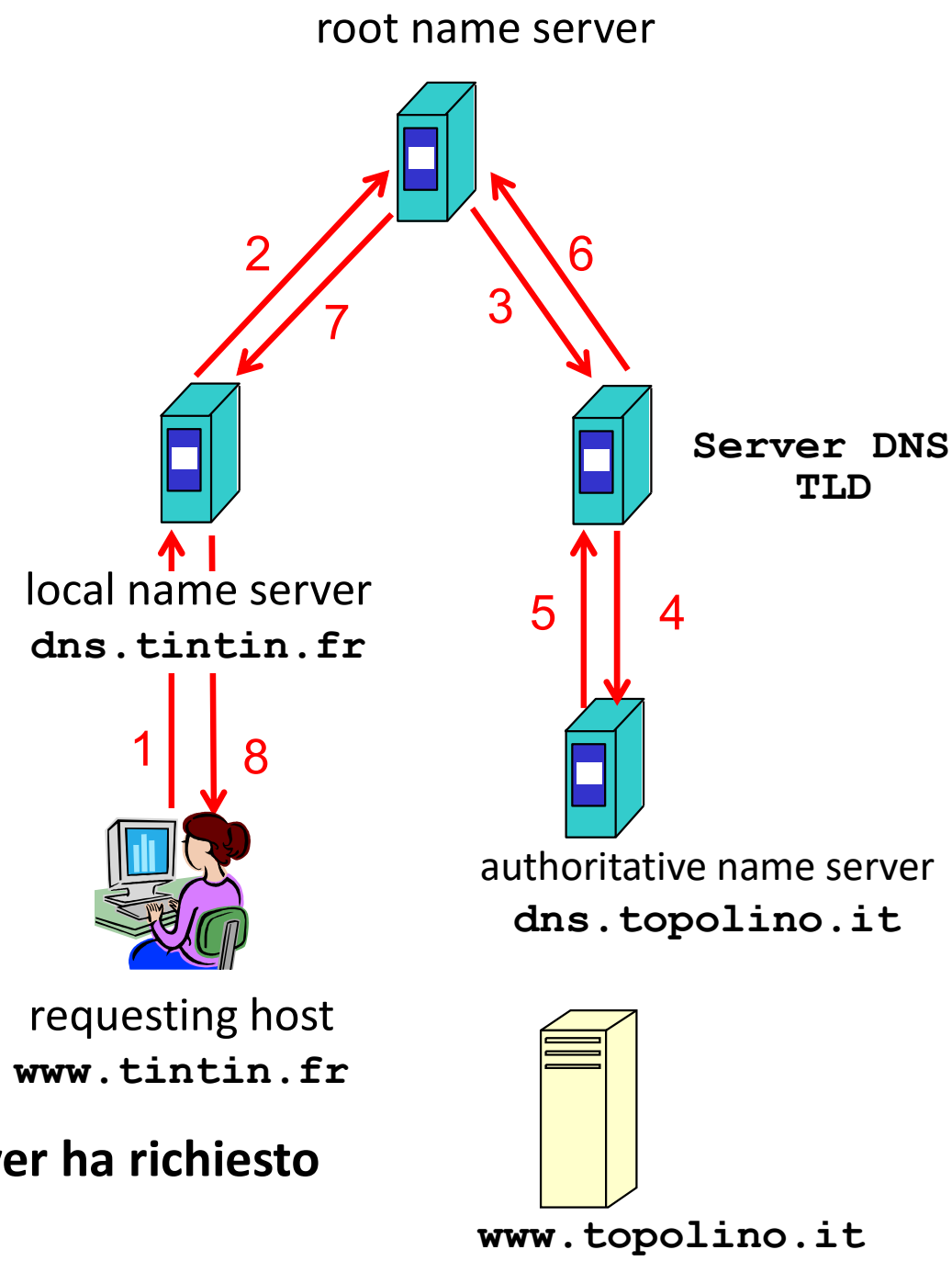
host www.tintin.fr cerca l'indirizzo IP di www.topolino.it

1. Contatta il suo DNS locale

   dns.tintin.fr

2. dns.tintin.fr contatta il root name server, se è necessario

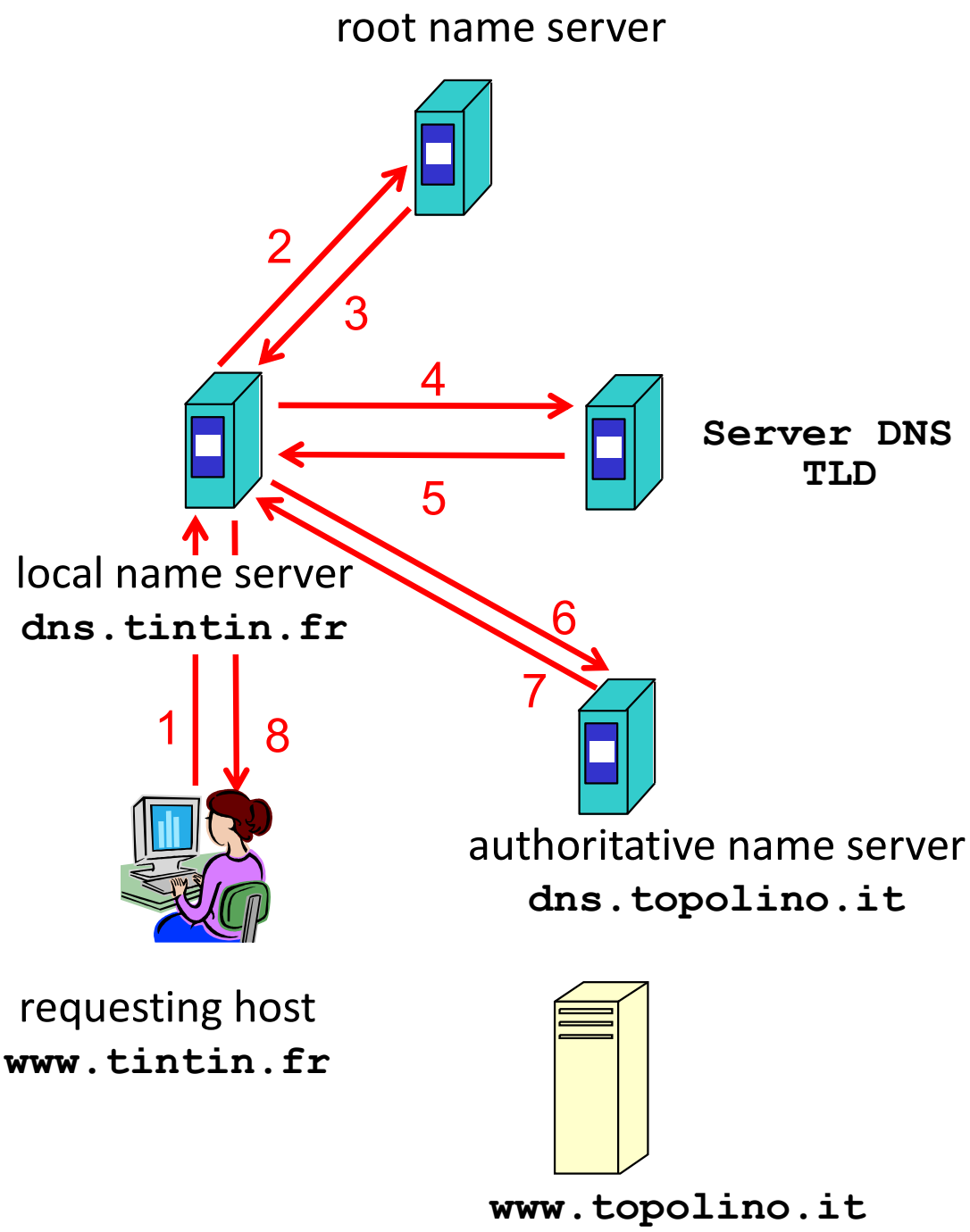
3. root name server contatta l'authoritative name server, dns.topolino.it, se è necessario




**Query RICORSIVA: il local name server ha richiesto una conversione completa**

# Esempio

- Query ITERATIVA:
  - Le risposte sono restituite direttamente al client
  - Viene restituito il riferimento al server da contattare successivamente



# DNS: caching ed aggiornamento dei record

- Una volta che un name server ha appreso una associazione, la mette nella *cache*
  - I record nella cache vengono cancellati dopo un certo tempo (timeout-TTL)
- I meccanismi di update/notifica sono descritti nella RFC 2136
- Migliora il ritardo e riduce il numero di messaggi DNS

# Record DNS

DNS: database distribuito di *resource records* (RR)

RR format: **(name, value, type,ttl)**

- TTL: quando la risorsa dovrà essere rimossa dalla cache
- I significati di Name e Value dipendono da Type
- Type=A
  - Name: hostname
  - Value: indirizzo IP
- Type= CNAME
  - Name: hostname (sinonimo)
  - Value: nome canonico dell'host
- Type=NS
  - Name: nome di dominio (e.g. unipi.it)
  - Value: hostname dell'authoritative name server per quel dominio
- Type=MX
  - Name: nome di dominio
  - Value: nome canonico del server di posta associato a name

*ci sono altri type...*

# Esempi di record DNS

- (HostName, indirizzoIPdiHostName, A, …)

  (www.cnn.com, 157.166.224.25, A, …)

  (www.cnn.com, 157.166.224.26, A, …)

- (Dominio, NomediAuthoritativeServePerDominio, NS, …)

  (cli.di.unipi.it, nameserver.cli.di.unipi.it*, NS,...)

[* il messaggio di risposta potrà contenere anche (nameserver.cli.di.unipi.it, 131.114.120.2,A,...)]

- (Alias, HostNameMailServerConTaleAlias, MX, …)

  (cli.di.unipi.it, mailserver.cli.di.unipi.it*, MX, …)

[* il messaggio di risposta potrà contenere anche (mailserver.cli.di.unipi.it, 131.114.11.39, A, ...)]

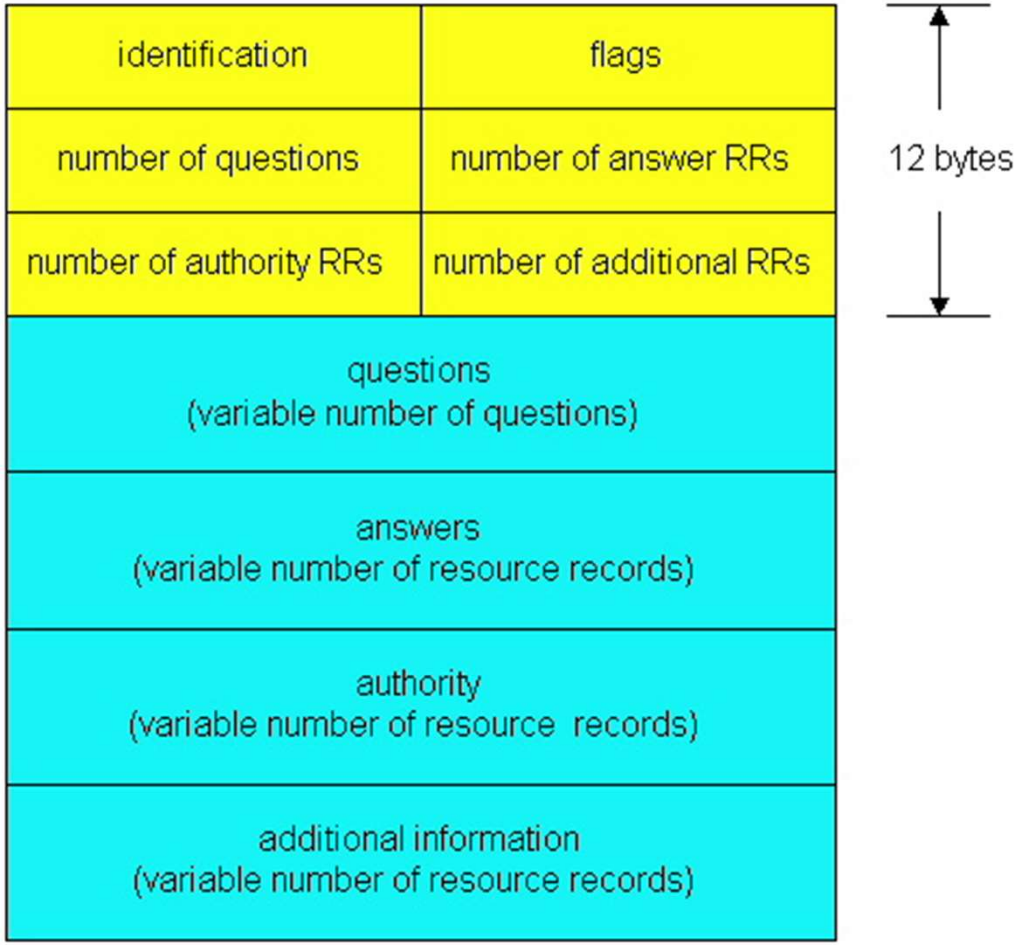
# Messaggi DNS

protocollo DNS

- UDP sulla porta 53
- permette uso di TCP (per messaggi di grandi dimensioni, tipicamente trasferimenti dati (file di zona) tra server DNS]
- messaggi di *query* e *reply*, entrambi con lo stesso *formato di messaggio*

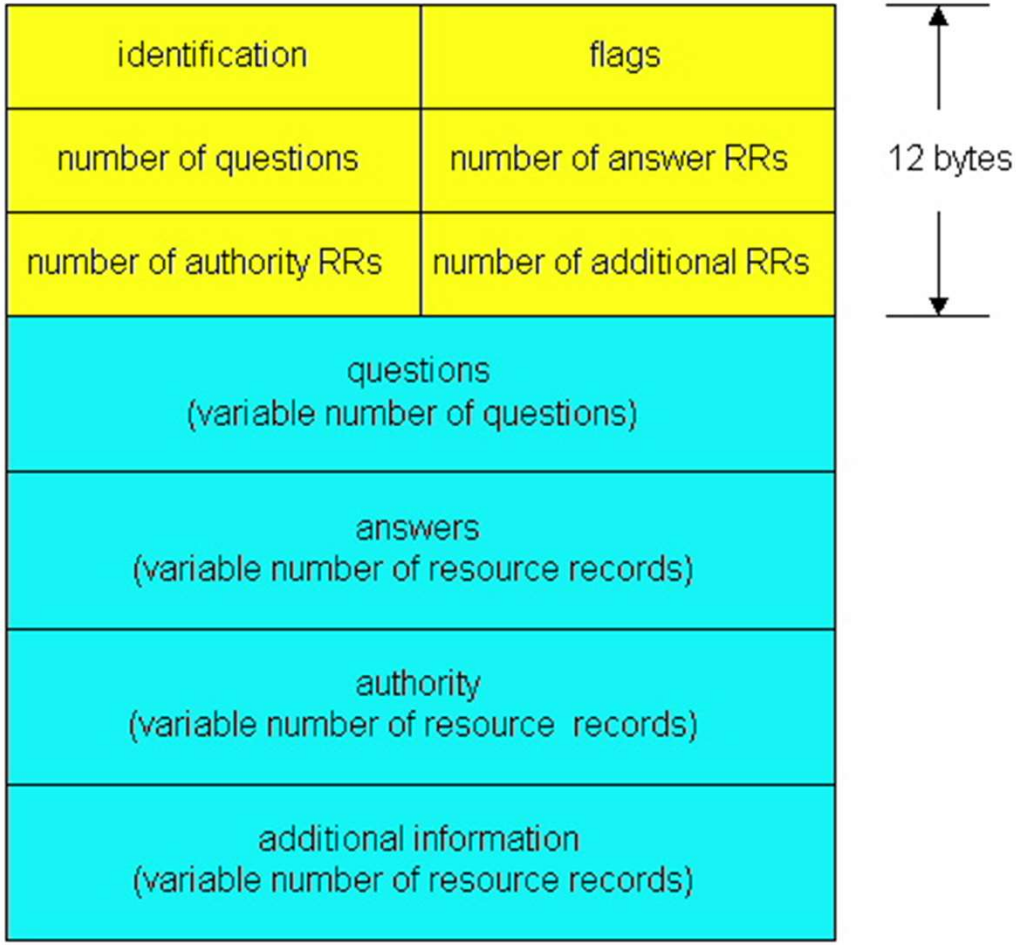
msg header

- identification: 16 bit- identification per query, la risposta (reply) ad una query usa lo stesso identification id
- flags:
  - Query (0) or reply (1)
  - reply is authoritative



# DNS message

- **Domande**: campi per il nome richiesto e il tipo di query
- **Risposte**: RR nella risposta alla domanda
- **Competenza**: record relativi ai server di competenza
- Informazioni aggiuntive



# Messaggi DNS: query e messaggi di risposta

- risposte contengono
  - zero o più RR nella sezione “Answer”
  - zero o più RR nella sezione “Additional”
- Esempio:

```
dns.poly.edu -----> TLD server
Question: QName=gaia.cs.umass.edu QType=NS

dns.poly.edu <---- TLD server
Answer: (umass.edu. dns.umass.edu, NS)     ------
Additional: (dns.umass.edu, 128.115.40.41, A)
```

# dig

*dig (domain information groper) is a flexible tool for interrogating DNS name servers. It performs DNS lookups and displays the answers that are returned from the name server(s) that were queried. Most DNS administrators use dig to troubleshoot DNS problems because of its flexibility, ease of use and clarity of output*

```
dig @server name type
```

Server: is the name or IP address of the name server to query.

Name: is the name of the resource record that is to be looked up.

Type_ indicates what type of query is required. If no type argument is supplied, dig will perform a lookup for an A record.

```
dig www.unipi.it @8.8.8.8
```

```
dig unipi.it MX @8.8.8.8
```

```
dig unipi.it ANY @8.8.8.8
```

# dig - example

```
 dig MX unipi.it

; <<>> DiG 9.16.21 <<>> MX unipi.it
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 46654
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 3, ADDITIONAL: 4

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;unipi.it.                      IN      MX

;; ANSWER SECTION:
unipi.it.               60      IN      MX      50 emailsecurity.unipi.it.

;; AUTHORITY SECTION:
unipi.it.               86400   IN      NS      ns1.garr.net.
unipi.it.               86400   IN      NS      ns2.unipi.it.
unipi.it.               86400   IN      NS      ns1.unipi.it.

;; ADDITIONAL SECTION:
emailsecurity.unipi.it. 60      IN      A       131.114.142.148
ns1.unipi.it.           35202   IN      A       131.114.21.10
ns2.unipi.it.           35202   IN      A       131.114.21.5

;; Query time: 36 msec
;; SERVER: 151.5.216.150#53(151.5.216.150)
;; WHEN: Mon Oct 16 16:00:45 ora legale Europa occidentale 2023
;; MSG SIZE  rcvd: 177
```

# Note

- Ogni organizzazione dotata di host Internet pubblicamente accessibili (es. server web e server di posta) deve fornire i record DNS di pubblico dominio che mappano i nomi di tali host in indirizzi IP (server mantenuti dall'organizzazione o ISP).
- Tali record possono essere mantenuti in un proprio DNS o tramite un gestore di servizi
- **Riferimenti**
  - IETF RFC 1034 https://tools.ietf.org/html/rfc1035
  - IETF RFC 1035 https://tools.ietf.org/html/rfc1035
  - e successive…

## Approfondimento

Determinare, analizzando la RFC 1034, in che modo viene stabilito se una query DNS viene gestita in modo ricorsivo o meno.

# DNS Hijacking

- il DNS è altamente decentralizzato. Nessun singolo server DNS contiene tutti gli indirizzi IP e i rispettivi domini. Una query viaggia lungo una catena di server DNS prima di ottenere il risultato.

- Il DNS Hijacking («dirottamento») è la pratica di restituire risposte non corrette alle query DNS reindirizzando il client verso siti malevoli

  - Local Hijacking
  - Router Hijacking
  - Rogue Hijacking
  - Man-in-the-Middle Attack