Livello applicazione: Protocollo DNS

Gaia Maselli

Identificazione degli host

Identificazione degli host

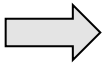
- Persone: molti identificatori:
  - nome, codice fiscale,
    numero della carta d'identità
- Host Internet hanno nomi (hostname)
  - www.google.com
  - w3.uniroma1.it
- I nomi sono facili da ricordare ma forniscono poca informazione sulla collocazione degli host all'interno di Internet
  - w3.uniroma1.it ci dice che l'host si trova probabilmente in Italia ma non dove
- Indirizzi IP per gli host:
  - indirizzo IP (32 bit) - usato per indirizzare i datagrammi
  - Più appropriato per le macchine

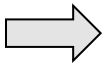
Indirizzo IP

- Consiste di 4 byte
  - E' costituito da una stringa in cui ogni punto separa uno dei byte espressi con un numero decimale compreso tra 0 e 255
- Presenta una struttura **gerarchica**
  - Leggendolo da destra a sinistra otteniamo informazioni sempre più specifiche sulla collocazione dell'host in Internet (rete di appartenenza)
- Esempio
  - 121.34.230.94

D: Come associare un indirizzo IP a un nome?

DNS: Domain Name System

www.uniroma1.it  

 130.186.99.43

Servizio DNS

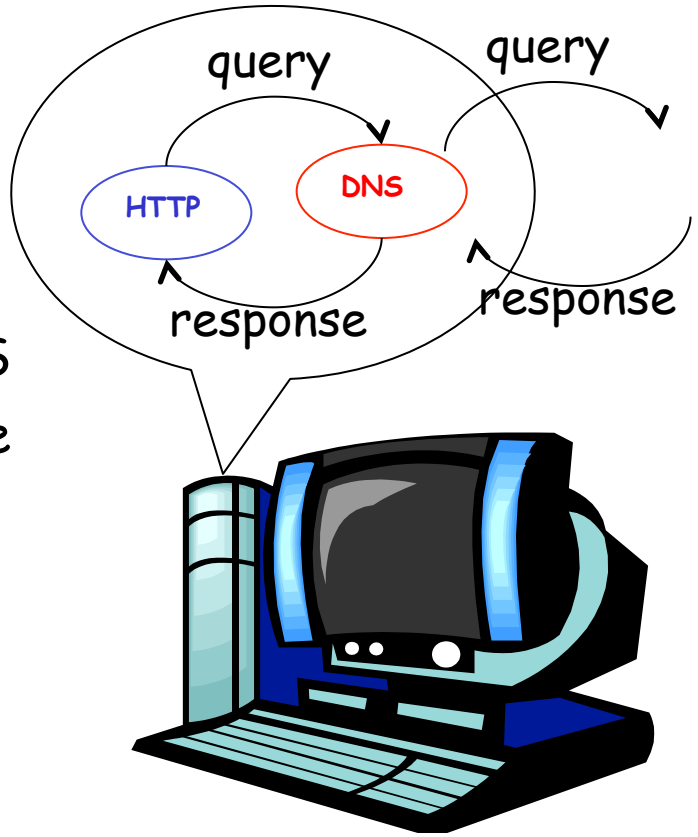
Domain Name System (RFC 1034, 1035):

- *Database distribuito* implementato in una gerarchia di *server DNS*
- *Protocollo a livello applicazione* che consente agli host di interrogare il database distribuito per *risolvere* i nomi (tradurre indirizzi/nomi)
- Il DNS viene utilizzato dagli altri protocolli di livello applicazione (HTTP, SMTP, FTP) per tradurre hostname in indirizzi IP
- Utilizza il trasporto UDP e indirizza la porta 53

Esempio di interazione con HTTP

Un browser (ossia client HTTP) di un host utente richiede la URL `www.someschool.edu`

1. L'host esegue il lato client dell'applicazione DNS
2. Il browser estrae il nome dell'host, `www.someschool.edu` dall'URL e lo passa al lato client dell'applicazione DNS
3. Il client DNS invia una query contenente l'hostname a un server DNS
4. Il client DNS riceve una risposta che include l'indirizzo IP corrispondente all'hostname
5. Ottenuto l'indirizzo IP dal DNS, il browser può dare inizio alla connessione TCP verso il server HTTP localizzato a quell'indirizzo IP



DNS: è un'applicazione?

- E` un protocollo del livello applicazione
  - Viene eseguito dagli end system secondo il paradigma client-server
  - Utilizza un protocollo di trasporto end-to-end per traferire messaggi tra gli end system (UDP)
- Non è un'applicazione con cui gli utenti interagiscono direttamente (eccetto amministratori di rete)
- Fornisce una funzionalità di base di internet per le applicazioni utente
- Rispecchia la filosofia di concentrare la complessità nelle parti periferiche della rete

Servizi DNS: aliasing

- Permette di associare un nome più semplice da ricordare a un nome complesso
- **Host aliasing**: un host può avere uno o più sinonimi (alias)
  - Esempio: `relay1.west-coast.enterprise.com` potrebbe avere due sinonimi, quali `enterprise.com` e `www.enterprise.com`
  - `relay1.west-coast.enterprise.com` è un hostname ***canonico***
  - `enterprise.com` e `www.enterprise.com` sono ***alias***
  - Gli alias sono più facili da ricordare
  - Il DNS può essere invocato da un'applicazione per l'hostname canonico di un sinonimo così come l'IP
- Mail server aliasing: spesso i mail server e il web server di una società hanno lo stesso alias, ma nomi canonici diversi
- Il DNS può essere invocato da un'applicazione per avere il nome canonico di un alias e il suo indirizzo IP

Servizi DNS: distribuzione del carico

- DNS viene utilizzato per distribuire il carico tra server replicati (es. web server)
- I siti con molto traffico (es. cnn.com) vengono replicati su più server, e ciascuno di questi gira su un sistema terminale diverso e presenta un indirizzo IP differente
- Hostname canonico associato a un insieme di indirizzi IP
- Il DNS contiene l'insieme di indirizzi IP
- Quando un client effettua un richiesta DNS per un nome mappato in un insieme di indirizzi, il server risponde con l'insieme di indirizzi ma variando l'ordinamento a ogni risposta
- La rotazione DNS distribuisce il traffico sui server replicati

DNS

- Traduce nomi in indirizzi IP
- Ai tempi di ARPANET era un file host.txt che veniva caricato durante la notte
- Adesso è un'applicazione che gira su ogni host
- Costituita da
  - un gran numero di server DNS distribuiti per il mondo
  - Un protocollo a livello applicazione che specifica la comunicazione tra server DNS e host richiedenti

Perché non centralizzare DNS?

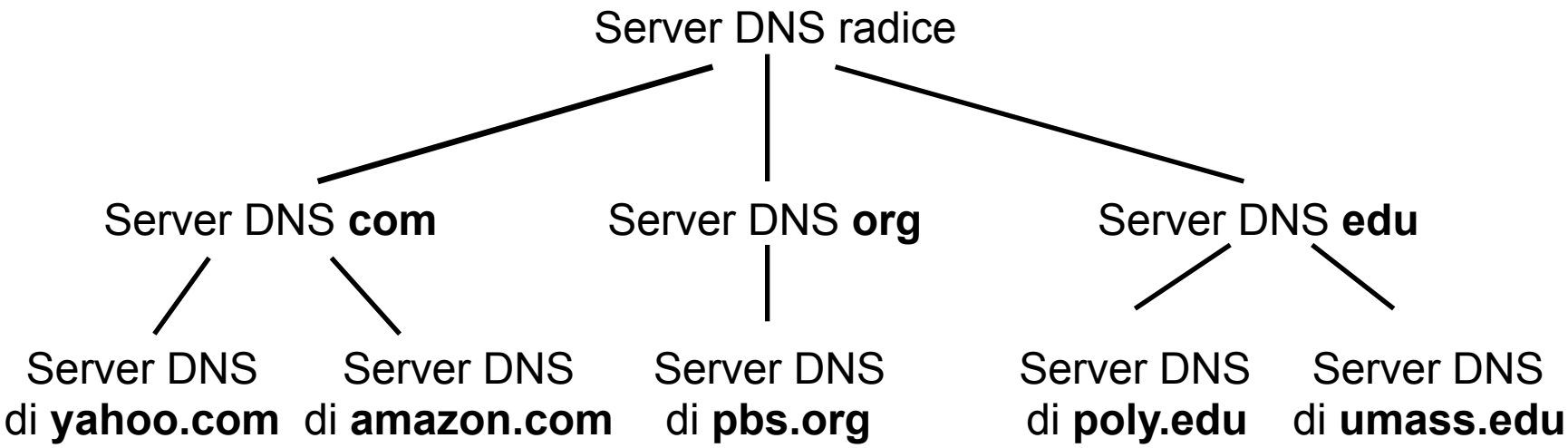
- singolo punto di guasto
- volume di traffico
- database centralizzato distante
- manutenzione

Un database centralizzato su un singolo server DNS non è *scalabile* !

Gerarchia DNS

- Nessun server DNS mantiene il mapping per tutti gli host in Internet
- Il mapping è distribuito su svariati server DNS
- Ci sono 3 classi di server DNS organizzati in una gerarchia:
  - **Root**
  - **Top-level domain (TLD)**
  - **Authoritative**
- Ci sono poi i server DNS **locali** con cui interagiscono direttamente le applicazioni

Database distribuiti e gerarchici

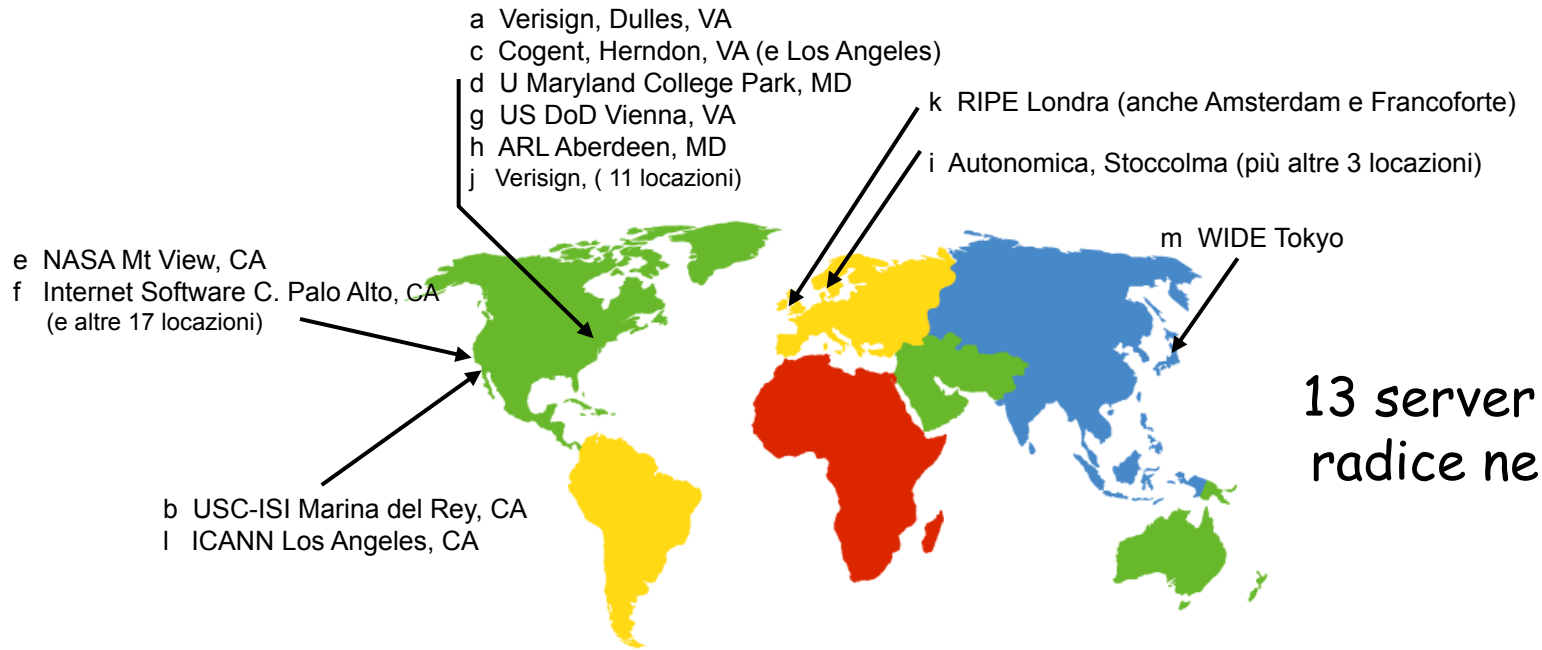


Esempio: Il client vuole l'IP di www.amazon.com

- Il client interroga il server radice (root) 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**

DNS: server DNS radice

- In Internet ci sono 13 server DNS radice
- Ognuno di questi server è replicato per motivi di sicurezza e affidabilità (in totale diventano 247 root server)
- I root server vengono contattati dai server DNS locali
- Server DNS radice:
  - contatta un server DNS TLD se non conosce la mappatura
  - ottiene la mappatura
  - restituisce la mappatura al server DNS locale



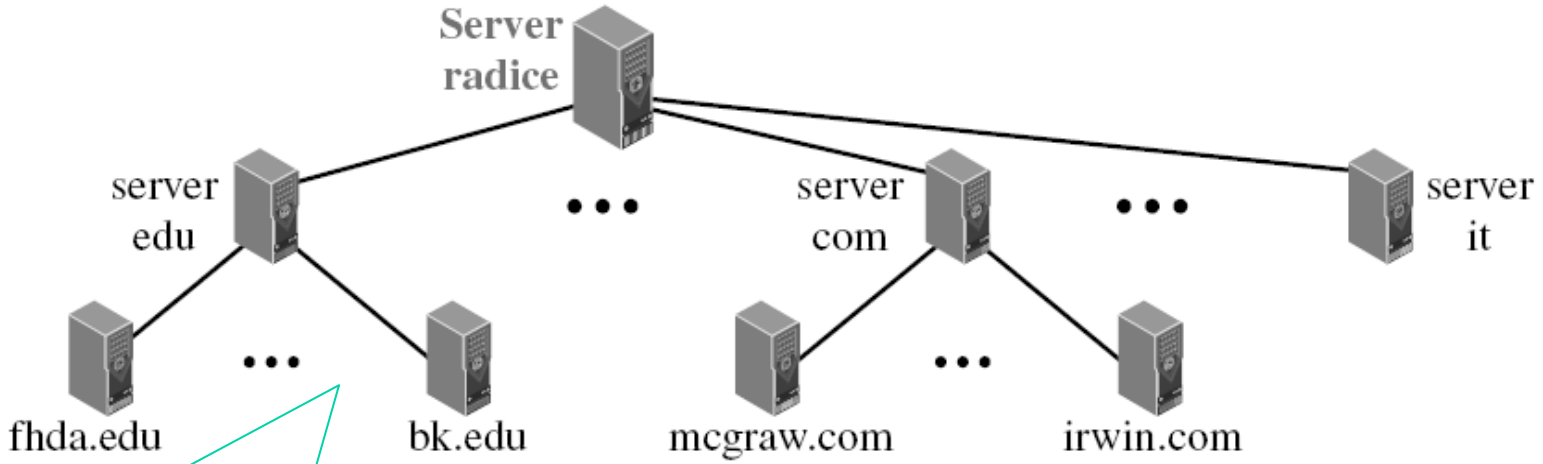
13 server DNS radice nel mondo

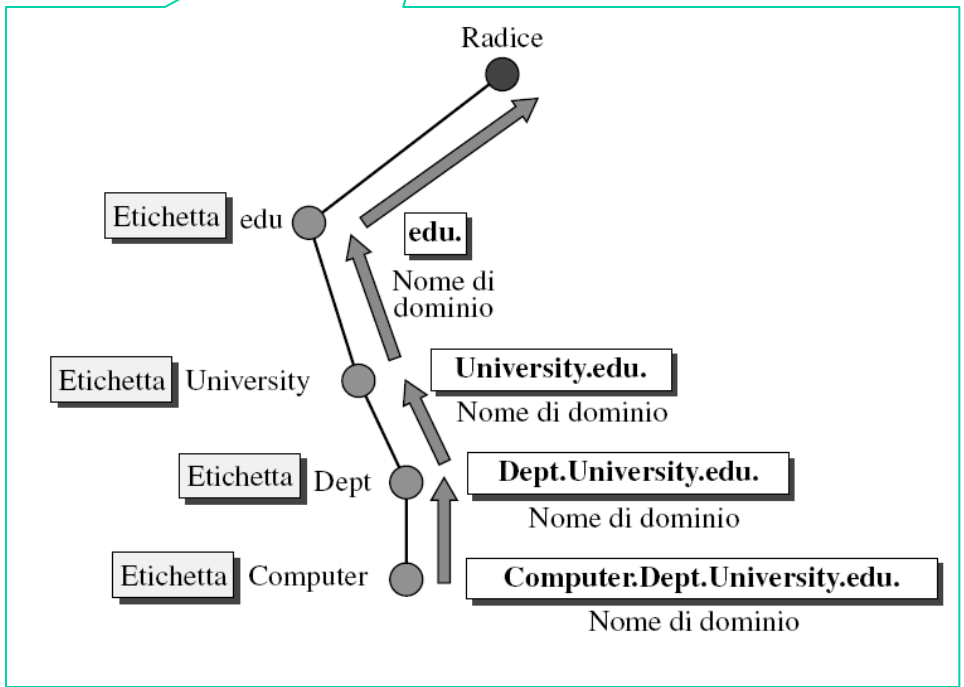
Server TLD e server di competenza

- Server TLD (top-level domain): si occupano dei domini **com**, **org**, **net**, **edu**, ecc. e di tutti i domini locali di alto livello, quali **it**, **uk**, **fr**, **ca** e **jp**.
  - La compagnia Verisign Global Registry Services gestisce i server TLD per il dominio com
  - La compagnia Educause gestisce quelli per il dominio edu

- Server di competenza (*authoritative server*): ogni organizzazione dotata di host Internet pubblicamente accessibili (quali i server web e i server di posta) deve fornire i record DNS di pubblico dominio che mappano i nomi di tali host in indirizzi IP.
  - possono essere mantenuti dall'organizzazione (università) o da un service provider
  - In genere sono due server (primario e secondario)

esempio


Server
radice
server
edu
server
com
server
it
fhda.edu
bk.edu
mcgraw.com
irwin.com


Radice
Etichetta
edu
edu.
Nome di
dominio
Etichetta
University
University.edu.
Nome di dominio
Etichetta
Dept
Dept.University.edu.
Nome di dominio
Etichetta
Computer
Computer.Dept.University.edu.
Nome di dominio

Etichette dei domini generici

Tabella 2.10 Etichette dei domini generici.

Etichetta	*Descrizione*	*Etichetta*	*Descrizione*
aero	Compagnie aeree e aziende aerospaziali	int	Organizzazioni internazionali
biz	Aziende (simile a com)	mil	Organizzazioni militari
com	Organizzazioni commerciali	museum	Musei
coop	Associazioni di cooperazione	name	Nomi di persone
edu	Istituzioni educative	net	Organizzazioni che si occupano di reti
gov	Istituzioni governative	org	Organizzazioni senza scopo di lucro
info	Fornitori di servizi informativi	pro	Organizzazioni professionali

Server DNS locale

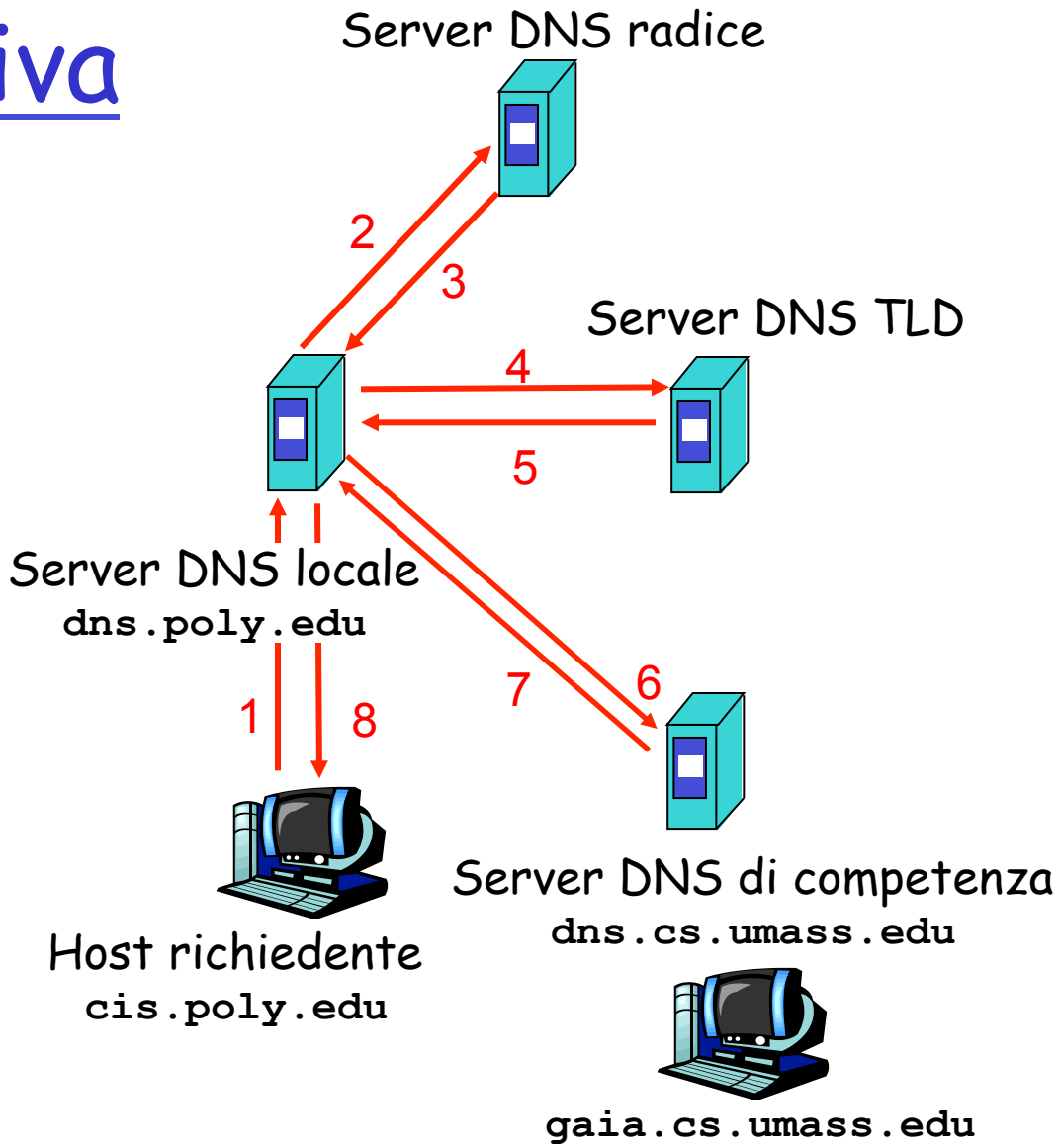
- Non appartiene strettamente alla gerarchia dei server
- Ciascun ISP (università, società, ISP residenziale) ha un server DNS locale.
  - detto anche "default name server"
- Quando un host effettua una richiesta DNS, la query viene inviata al suo server DNS locale
  - il server DNS locale opera da proxy e inoltra la query in una gerarchia di server DNS

Query iterativa

- L'host cis.poly.edu vuole l'indirizzo IP di gaia.cs.umass.edu

Query iterativa: (2-7)

- Il server contattato risponde con il nome del server da contattare
- "Io non conosco questo nome, ma puoi chiederlo a questo server".

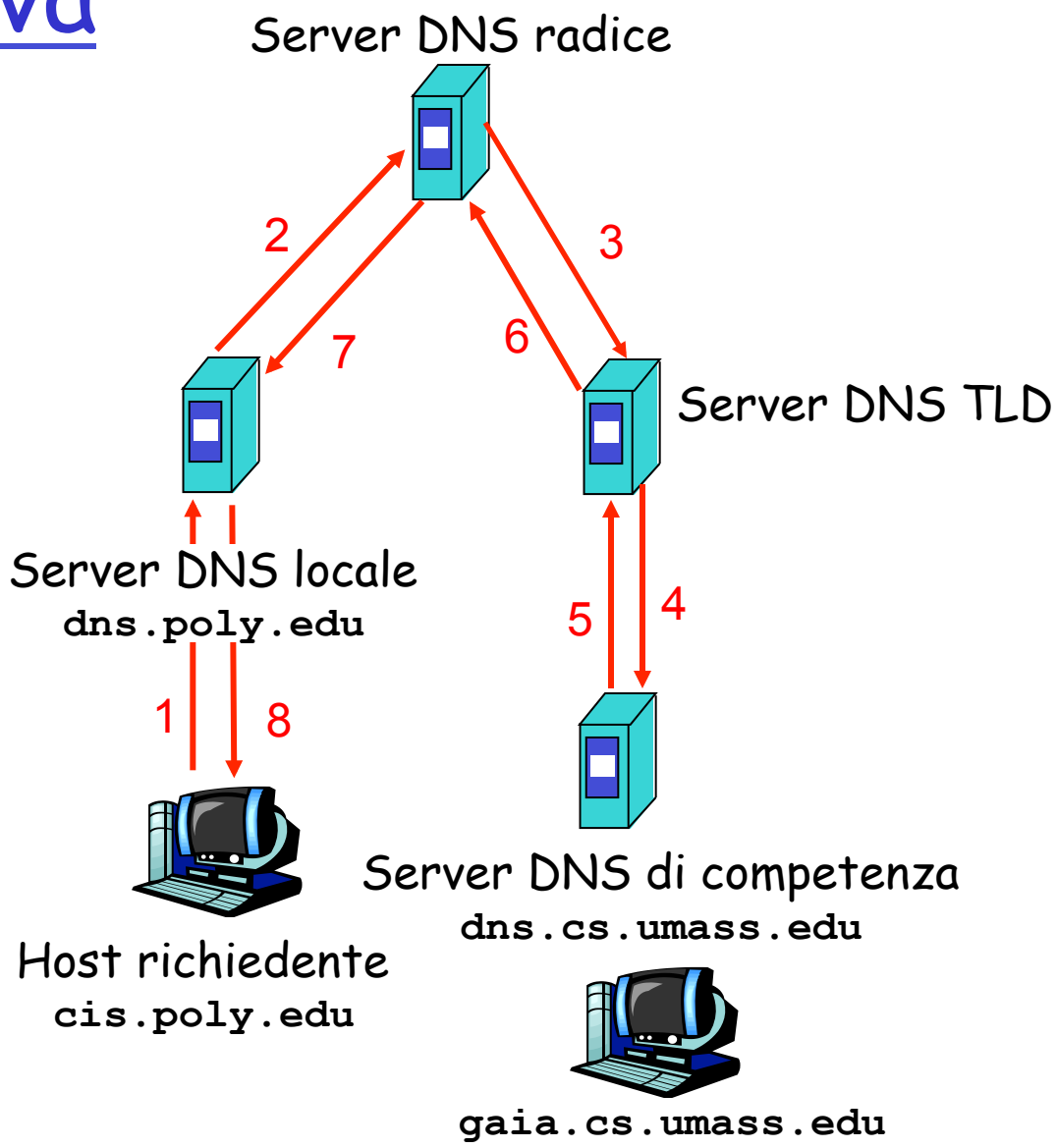


N.B. Per ottenere la mappatura di un hostname sono stai inviati 8 messaggi !!!

Query ricorsiva

Query ricorsiva:

- Affida il compito di tradurre il nome al server DNS contattato



DNS: caching

- DNS sfrutta il caching per migliorare le prestazioni di ritardo e per ridurre il numero di messaggi DNS che "rimbalzano" in Internet
- Una volta che un server DNS impara la mappatura, la mette nella *cache*
  - le informazioni nella cache vengono invalidate (spariscono) dopo un certo periodo di tempo (es. 2 giorni)
  - tipicamente un server DNS locale memorizza nella cache gli indirizzi IP dei server TLD (ma anche quelli di competenza)
    - quindi i server DNS radice non vengono visitati spesso
- Esempio: più utenti in dipartimento che si connettono sul sito dell'università di Berkley
- I meccanismi di aggiornamento/notifica sono progettati da IETF
  - RFC 2136
  - http://www.ietf.org/html.charters/dnsind-charter.html

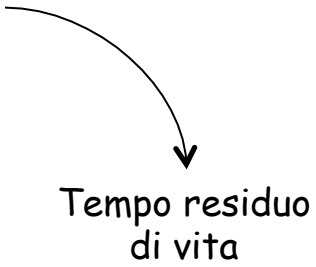
DNS record e messaggi

- Il mapping è mantenuto nei database sotto forma di **resource record (RR)**
- Ogni RR mantiene un mapping (es. tra hostname e indirizzo IP, alias e nome canonico, etc.)
- I record vengono spediti tra server e all'host richiedente all'interno di **messaggi DNS**
- Un messaggio può contenere più RR

Record DNS

- Database distribuito che memorizza i record di risorsa o resource record (RR).
- Ogni messaggio di risposta DNS trasporta uno o più RR

Formato RR: `(Name, Value, Type, TTL)`



Record DNS

Formato RR: **(Name, Value, Type, TTL)**

- Type=A

Hostname ➔ IP address

  - **name** è il nome dell'host
  - **value** è l'indirizzo IP

Es. (`relay1.bar.foo.com`, 45.37.93.126, A)

Record DNS

Formato RR: **(Name, Value, Type, TTL)**

- Type=CNAME

Alias ➔ Canonical Name

  - **name** è il nome alias di qualche nome "canonico" (nome vero)
  - **value** è il nome canonico

Es. (`foo.com`, `relay1.bar.foo.com`, CNAME)

Record DNS

Formato RR: **(Name, Value, Type, TTL)**

- Type=NS

Domain name ➔ Name Server

  - **name** è il dominio (ad esempio foo.com)
  - **value** è il nome dell'host del server di competenza di questo dominio

Es. (`foo.com`, `dns.foo.com`, NS)

Record DNS

Formato RR: **(Name, Value, Type, TTL)**

- Type=MX

Alias ➔ mail server canonical name

  - **value** è il nome canonico del server di posta associato a **name**

Es. (`foo.com, mail.bar.foo.com,` MX)

Tipi di record

Tabella 2.11 Tipi di record.

Tipo	*Interpretazione del campo valore*
A	Indirizzo IPv4 a 32 bit (si veda il Capitolo 4)
NS	Identifica i server autoritativi di una zona
CNAME	Indica che un nome di dominio è un alias (un nome alternativo) per il nome di dominio ufficiale (detto anche canonico)
SOA	Specifica una serie di informazioni autoritative riguardo una zona
MX	Indica il server di posta del dominio
AAAA	Indirizzo IPv6 (si veda il Capitolo 4)

Esempio

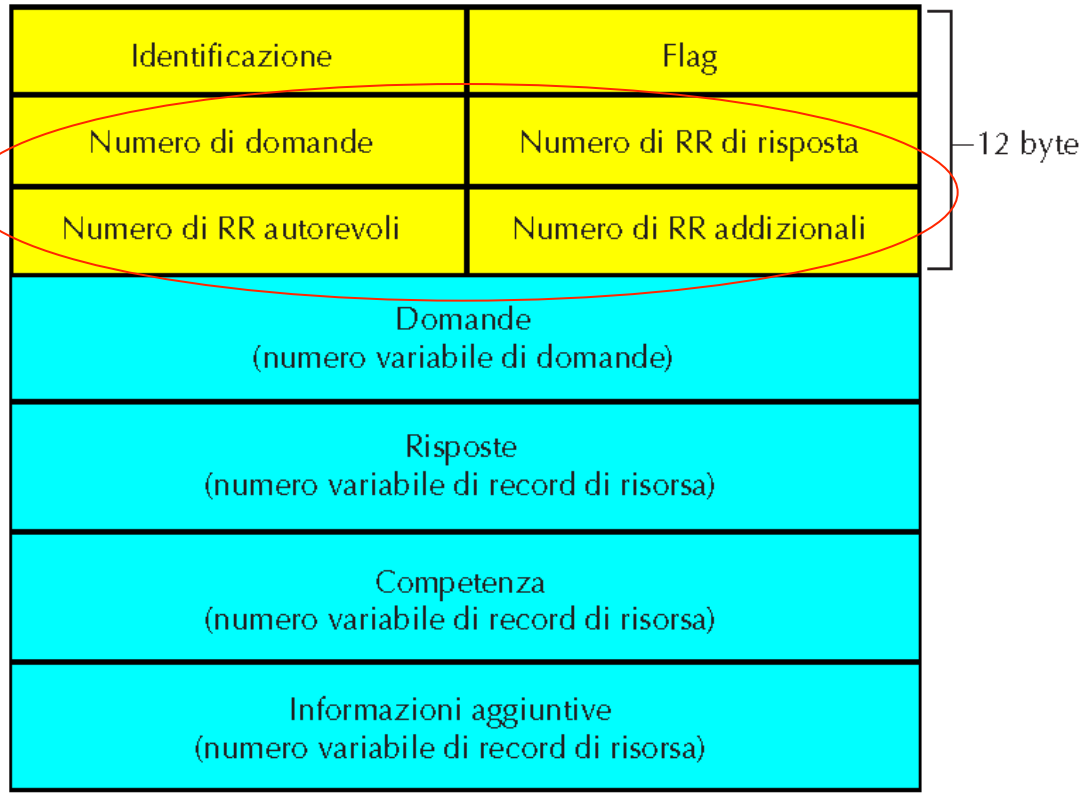
- Server di competenza per un hostname
  - Contiene un record di tipo A per l'hostname
  - Es. (`corsi.di.uniroma1.it, 131.111.45.68, A`)
- Server non di competenza per un dato hostname
  - Contiene un record di tipo NS per il dominio che include l'hostname
  - Contiene un record di tipo A che fornisce l'indirizzo IP del server DNS nel campo `value` del record NS
- Es.:
  - Un server TLD it non è competente per l'host `corsi.di.uniroma1.it`
  - Contiene (`uniroma1.it, dns.uniroma1.it, NS`)
    (`dns.uniroma1.it, 128.119.40.111, A`)

Messaggi DNS

Protocollo DNS: domande (query) e messaggi di *risposta*, entrambi con lo stesso *formato*

Intestazione del messaggio

- Identificazione: numero di 16 bit per la domanda; la risposta alla domanda usa lo stesso numero
- Flag:
  - domanda o risposta
  - richiesta di ricorsione
  - ricorsione disponibile
  - risposta di competenza (il server è competente per il nome richiesto)
- Numero di: numero di occorrenze delle quattro sezioni di tipo dati che seguono



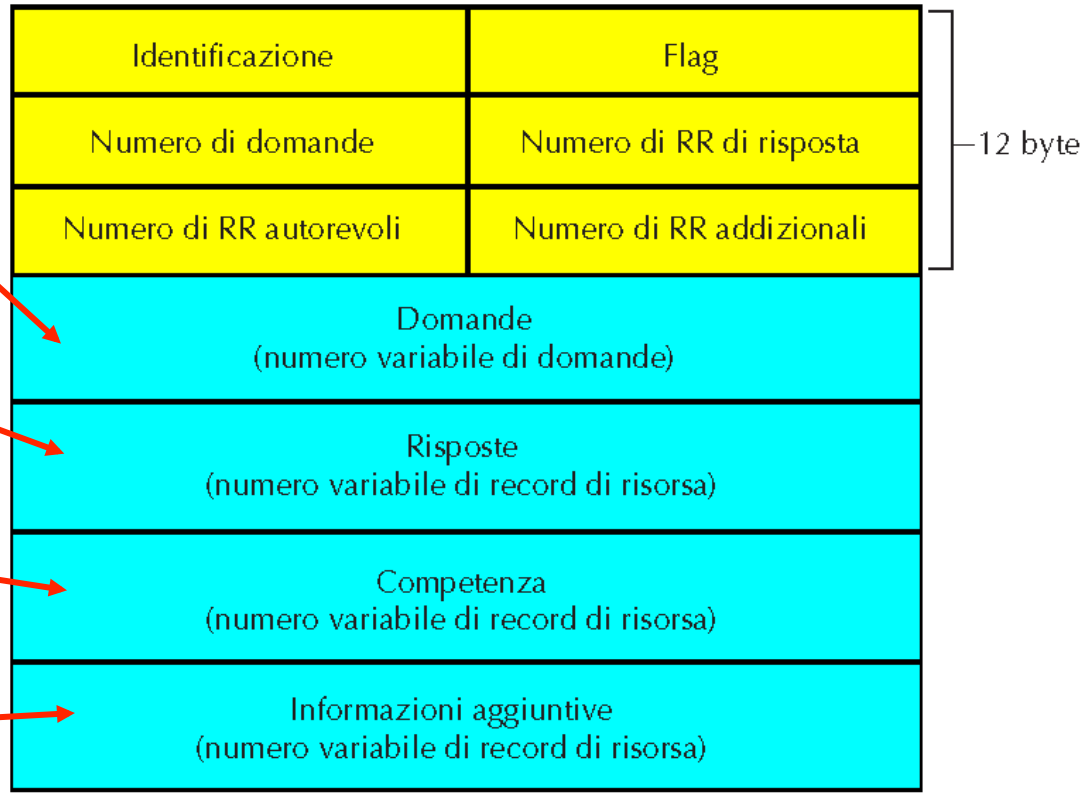
Messaggi DNS

Campi per il nome richiesto e il tipo di domanda (A, MX)

RR nella risposta alla domanda
Più RR nel caso di server replicati

Record per i server di competenza

Informazioni extra che possono essere usate



Nel caso di una risposta MX, il campo di risposta contiene il record MX con il nome canonico del server di posta, mentre la sezione aggiuntiva contiene un record di tipo A con l'indirizzo IP relativo all'hostname canonico del server di posta

Inserire record nel database DNS

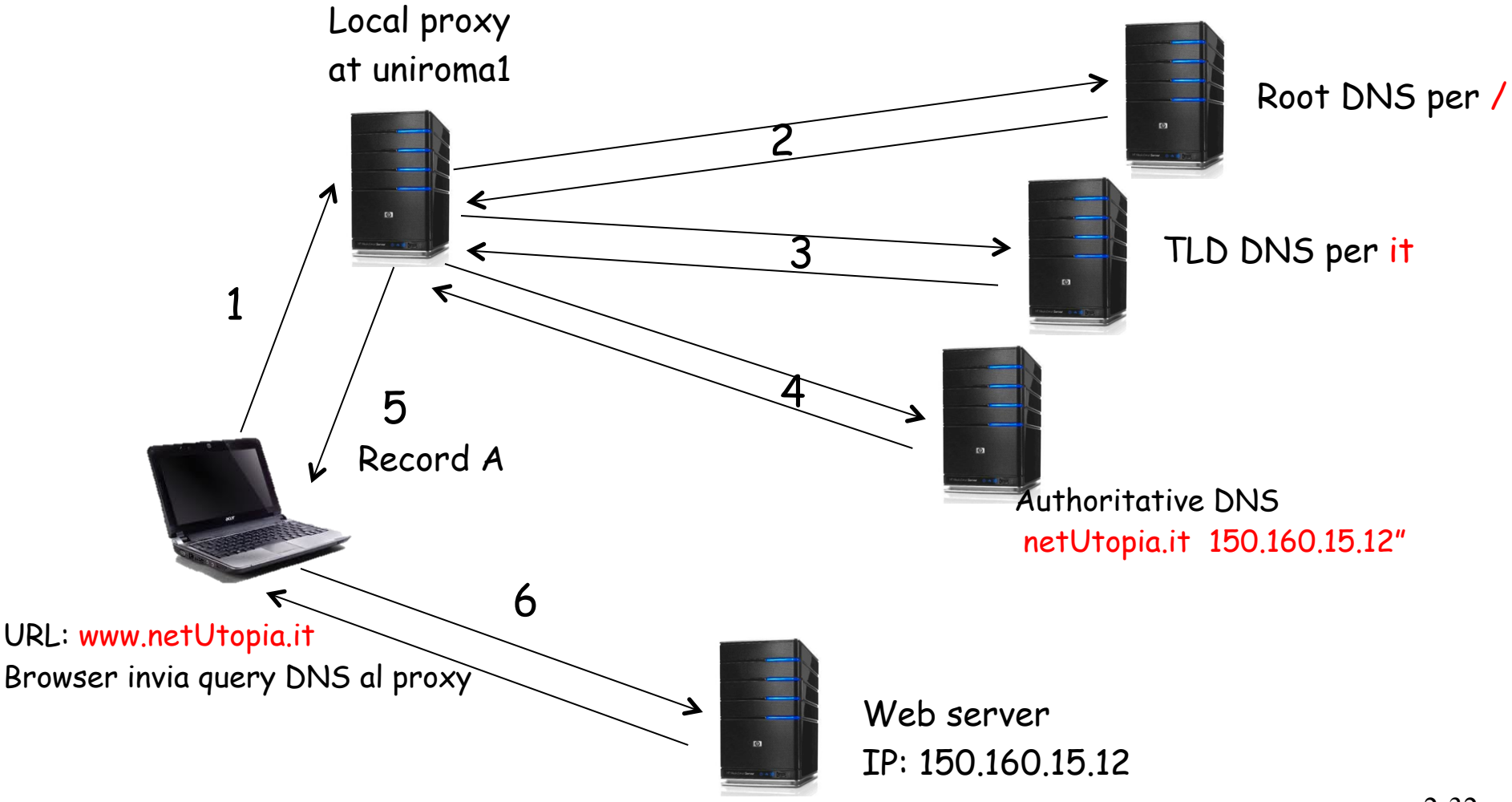
- Esempio: abbiamo appena avviato la nuova società "Network Utopia"

N.B. E' possibile aggiungere nuovi domini al DNS contattando un registrar (aziende commerciali accreditate dall' ICANN).
Il registrar in cambio di un compenso verifica l'unicità del dominio richiesto e lo inserisce nel database

- Registriamo il nome **networkuptopia.it** presso registrar (www.registro.it)
  - Inseriamo nel server di competenza un record tipo A per www.networkuptopia.it e un record tipo MX per networkutopia.it
  - Forniamo al registrar i nomi e gli indirizzi IP dei server DNS di competenza (primario e secondario)
  - Registrar inserisce due RR nel server TLD `it`:

  - `(networkutopia.it, dns1.networkutopia.it, NS)`
  - `(dns1.networkutopia.it, 212.212.212.1, A)`
- In che modo gli utenti otterranno l'indirizzo IP del nostro sito web?

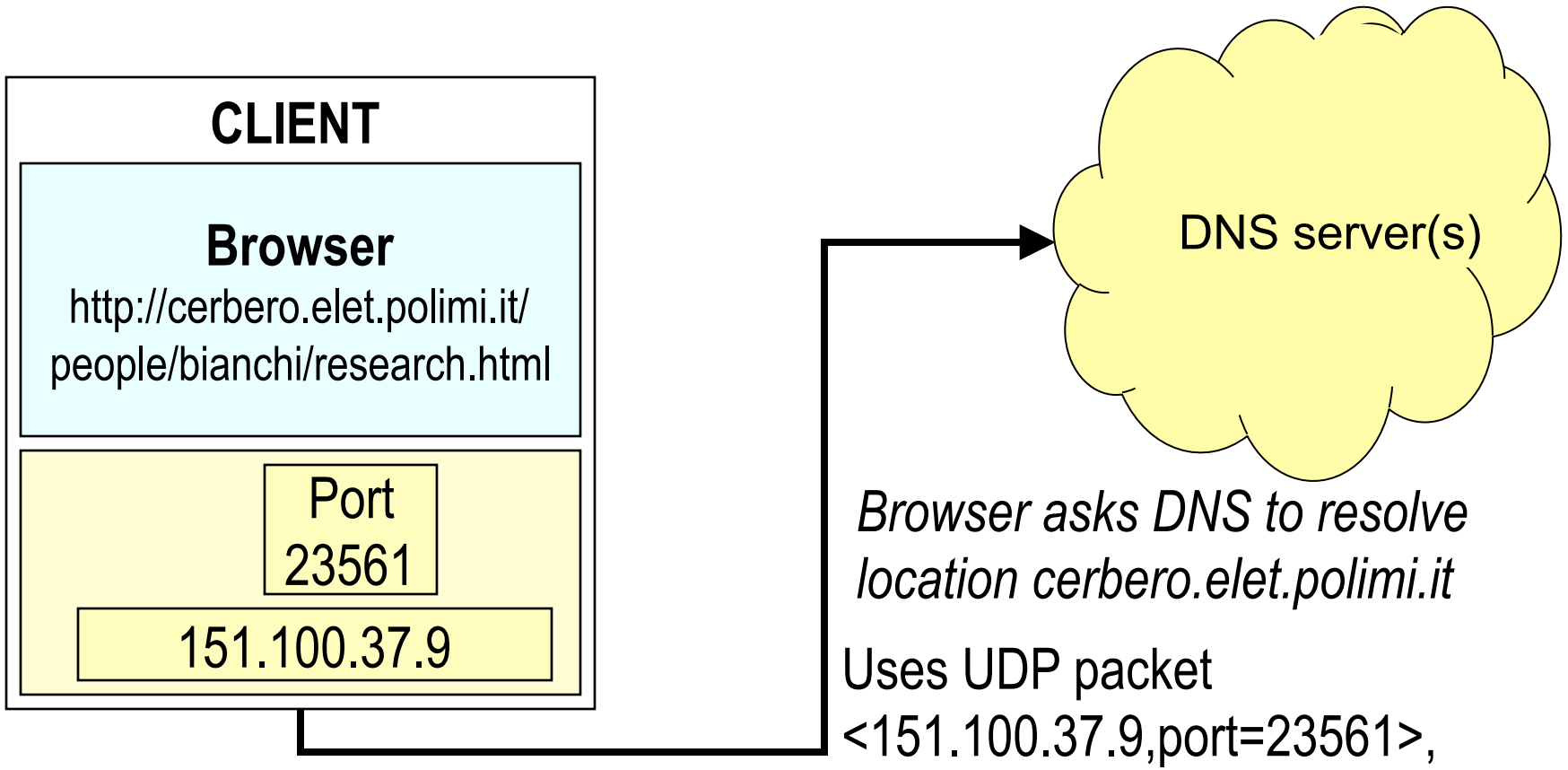
esempio


Local proxy
at uniroma1
Root DNS per /
TLD DNS per it
Authoritative DNS
netUtopia.it 150.160.15.12"
Web server
IP: 150.160.15.12
1
2
3
4
5
Record A
6
URL: www.netUtopia.it
Browser invia query DNS al proxy

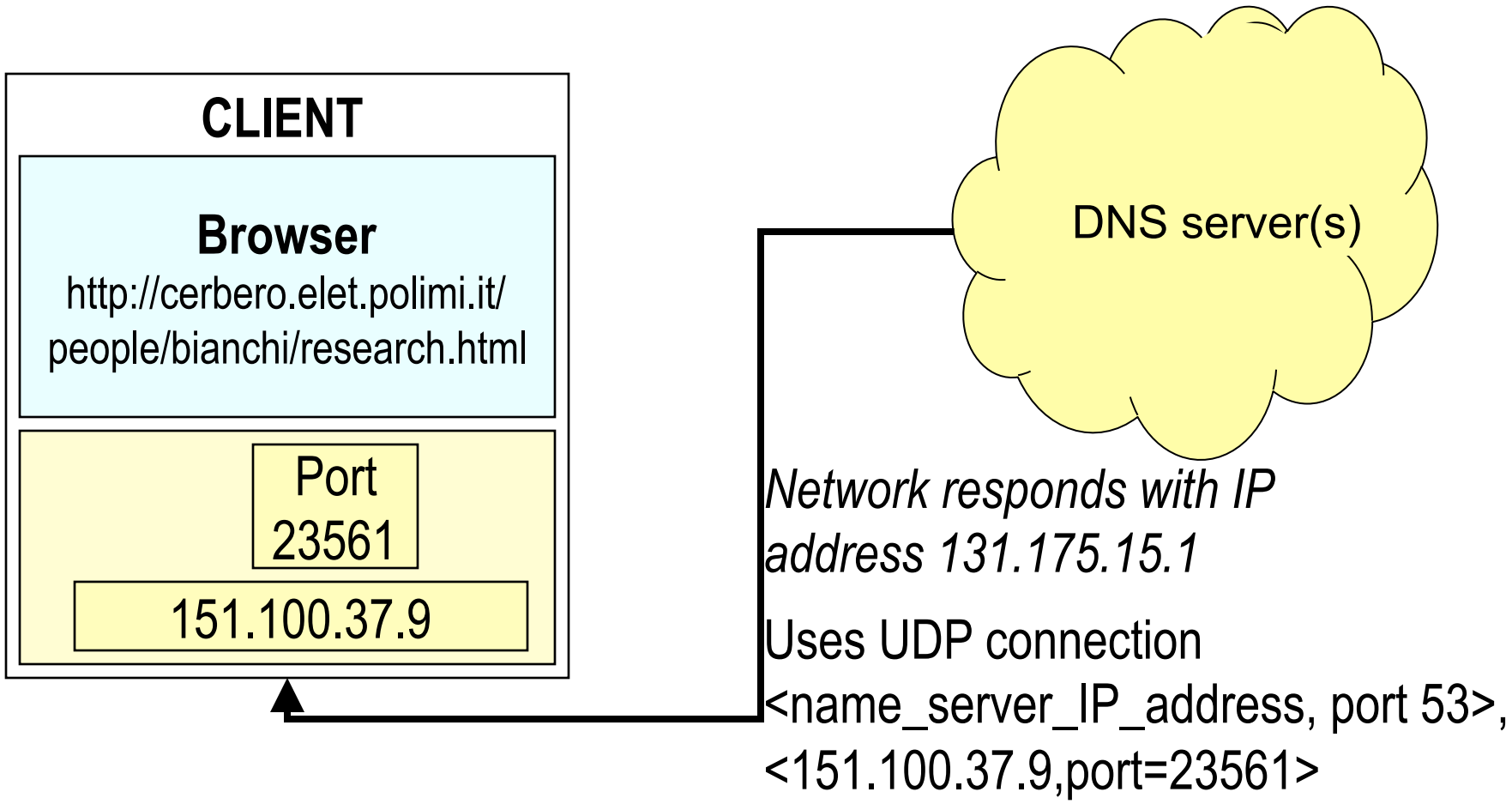
Perche' UDP?

- Less overhead
  - Messaggi corti
  - Tempo per set-up connessione di TCP lungo
  - Un unico messaggio deve essere scambiato tra una coppia di server (nella risoluzione contattati diversi server—se si usasse TCP ogni volta dovremmo mettere su la connessione!!)
- Se un messaggio non ha risposta entro un timeout?
  - Semplicemente viene ri-inviato dal resolver (problema risolto dallo strato applicativo)

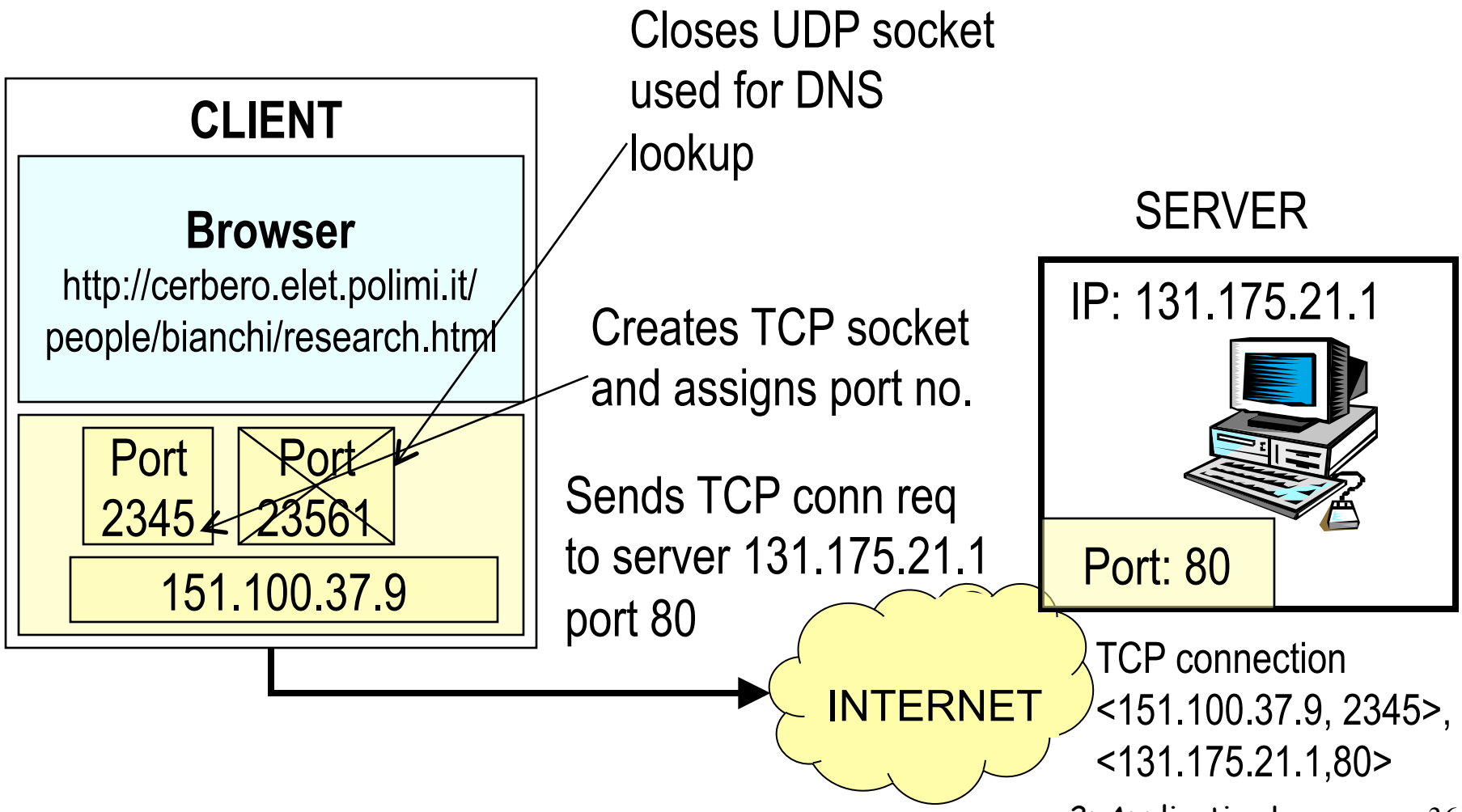
Un esempio: uso di DNS da parte di un client web



opening transport session: client side, step b



opening transport session: client side, step c



opening transport session: server side

- httpd (http daemon) process listens for arrival of connection requests from port 80.
- Upon connection request arrival, server decides whether to accept it, and send back a TCP connection accept
- This opens a TCP connection, uniquely identified by client address+port and server address+port 80 (coppia di indirizzi socket)

Prova pratica

- `Nslookup`: command-line tool to query Internet DNS interactively
- `nslookup` dal prompt dei comandi
  - `nslookup www.di.uniroma1.it`
  - `nslookup –type=NS uniroma1.it`
  - `nslookup uniroma1.it 151.100.4.13`
  - `nslookup –type=NS .`