

*Reti di calcolatori e Internet:
Un approccio top-down*

7^a edizione
Jim Kurose, Keith Ross

Pearson Paravia Bruno Mondadori Spa

Capitolo 2: Livello di applicazione

- ❑ Principi delle applicazioni di rete
- ❑ Web e HTTP
- ❑ Posta elettronica
 - ❖ SMTP, POP3, IMAP
- ❑ DNS
- ❑ Applicazioni P2P
- ❑ Programmazione delle socket con TCP
- ❑ Programmazione delle socket con UDP

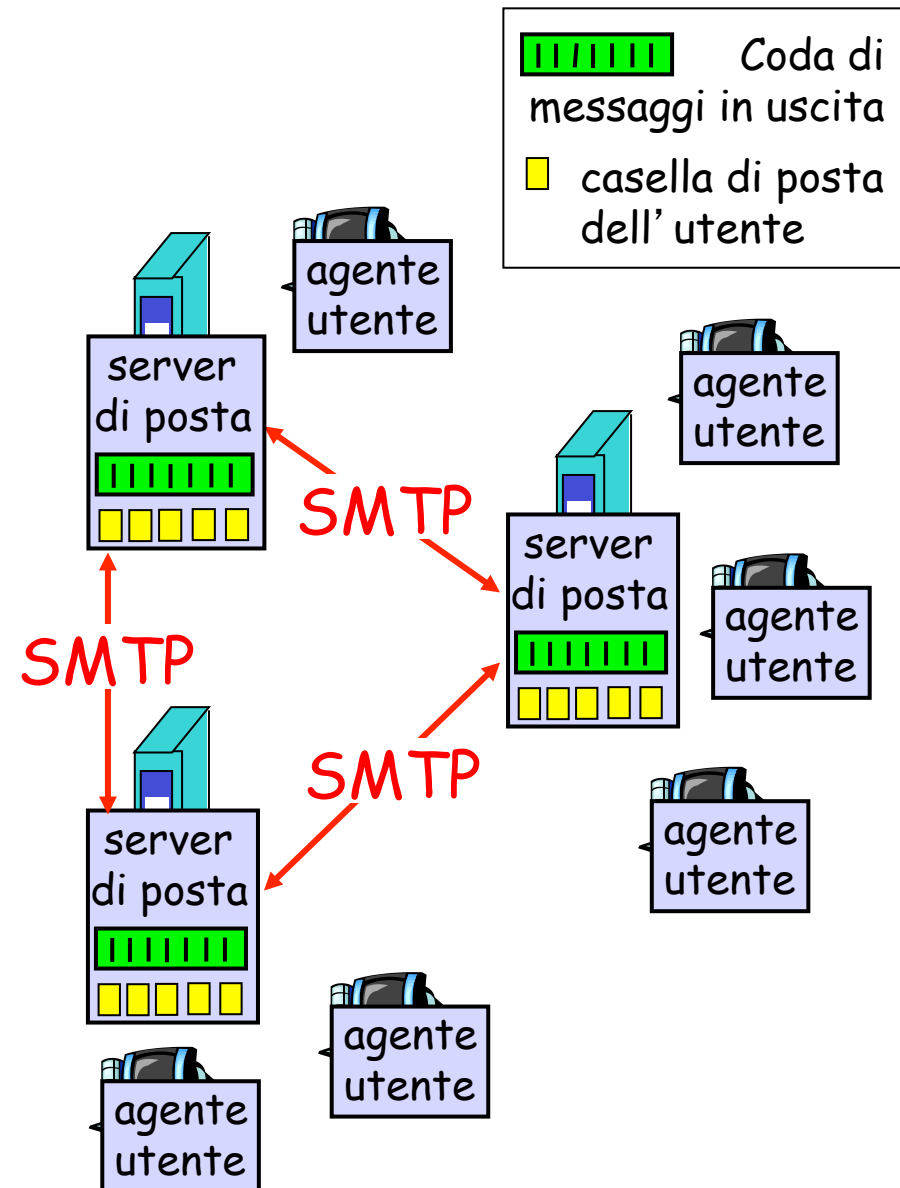
Posta elettronica

Componenti architetturali:

- ❑ agente utente (sorgente e destinatario)
- ❑ server di posta (sorgente e destinatario)

Agente utente

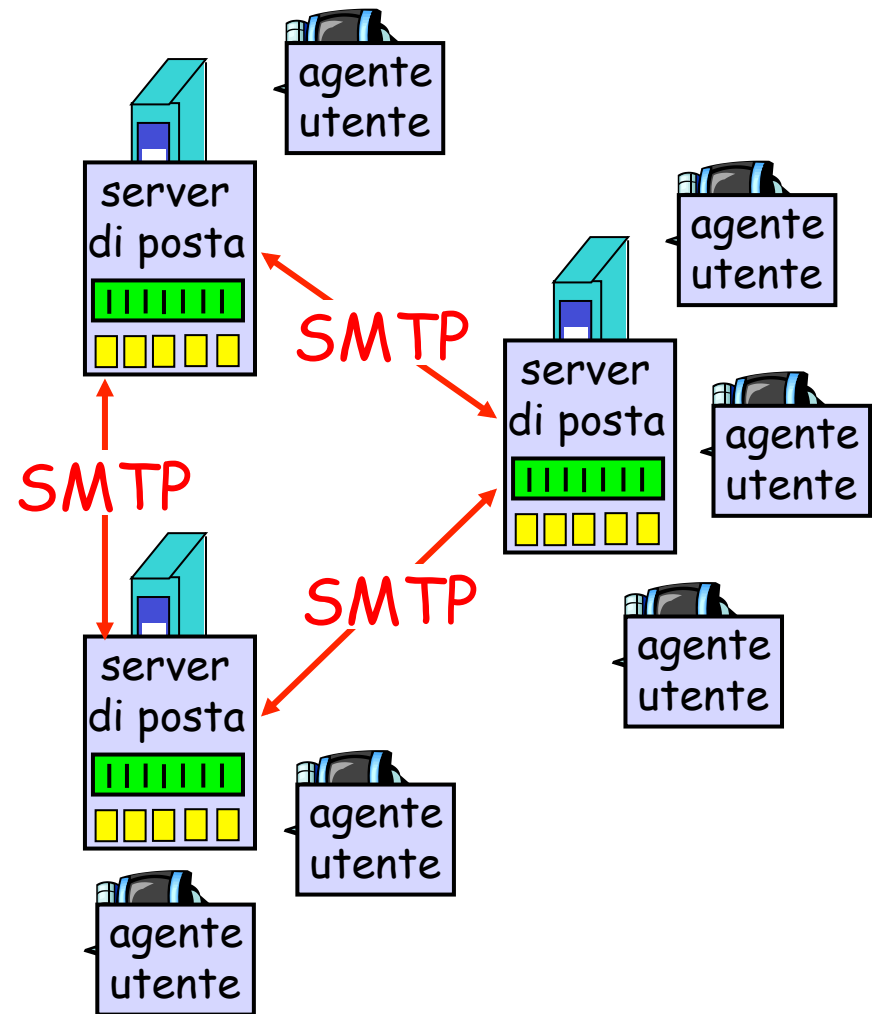
- ❑ composizione, editing, lettura dei messaggi di posta elettronica
- ❑ esempi: Outlook, Thunderbird, Mail
- ❑ i messaggi in uscita o in arrivo sono memorizzati sul server



Posta elettronica: server di posta

Server di posta

- ❑ **Casella di posta** (*mailbox*) contiene i messaggi in arrivo per l'utente
- ❑ **Coda di messaggi** da trasmettere
- ❑ **Protocollo SMTP** tra server di posta per inviare messaggi di posta elettronica
 - ❖ client: server di posta trasmittente
 - ❖ "server": server di posta ricevente

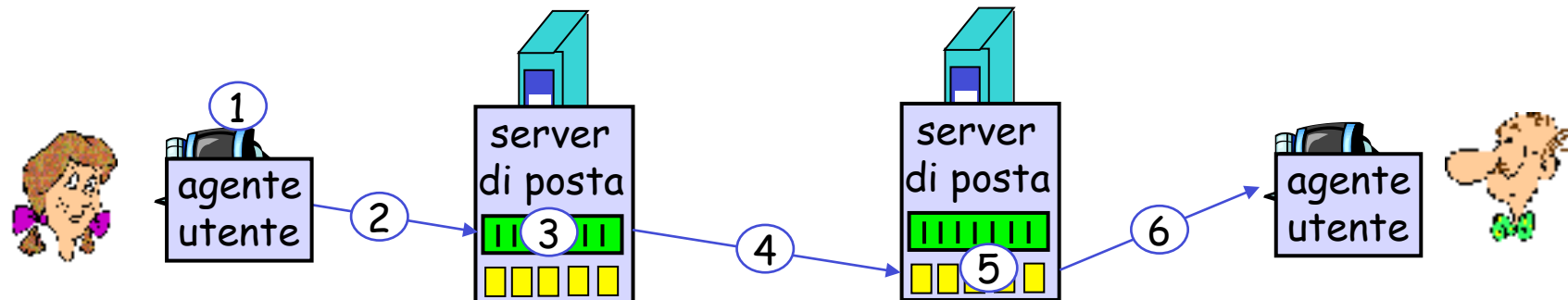


Posta elettronica: SMTP [RFC 2821]

- ❑ usa TCP per trasferire in modo affidabile i messaggi di posta elettronica dal client al server, porta 25
- ❑ trasferimento diretto: il server trasmittente al server ricevente
- ❑ tre fasi per il trasferimento
 - ❖ handshaking (saluto) di livello applicativo
 - ❖ trasferimento di messaggi
 - ❖ chiusura
- ❑ interazione comando/risposta
 - ❖ **comandi**: testo ASCII
 - ❖ **risposta**: codice di stato ed espressione
- ❑ i messaggi devono essere nel formato ASCII a 7 bit

Scenario: Alice invia un messaggio a Bob

- 1) Alice usa il suo agente utente per comporre il messaggio da inviare "a" bob@school.edu
- 2) L'agente utente di Alice invia un messaggio al server di posta di Alice; il messaggio è posto nella coda di messaggi
- 3) Il lato client di SMTP apre una connessione TCP con il server di posta di Bob
- 4) Il client SMTP invia il messaggio di Alice sulla connessione TCP
- 5) Il server di posta di Bob pone il messaggio nella casella di posta di Bob
- 6) Bob invoca il suo agente utente per leggere il messaggio



Esempio di interazione SMTP

```
S: 220 hamburger.edu
C: HELO crepes.fr
S: 250 Hello crepes.fr, pleased to meet you
C: MAIL FROM: <alice@crepes.fr>
S: 250 alice@crepes.fr... Sender ok
C: RCPT TO: <rob@hamburger.edu>
S: 250 rob@hamburger.edu ... Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: Do you like ketchup?
C: How about pickles?
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 hamburger.edu closing connection
```

SMTP: note finali

- ❑ SMTP usa connessioni persistenti
- ❑ SMTP richiede che il messaggio (intestazione e corpo) sia nel formato ASCII a 7 bit
- ❑ Il server SMTP usa CRLF.CRLF per determinare la fine del messaggio

Confronto con HTTP:

- ❑ HTTP: pull
- ❑ SMTP: push
- ❑ Entrambi hanno un'interazione comando/risposta in ASCII, codici di stato
- ❑ HTTP: ciascun oggetto è incapsulato nel suo messaggio di risposta
- ❑ SMTP: più oggetti vengono trasmessi in un unico messaggio

Formato dei messaggi di posta elettronica

SMTP: protocollo per scambiare messaggi di posta elettronica

RFC 5322: standard per il formato dei messaggi di testo:

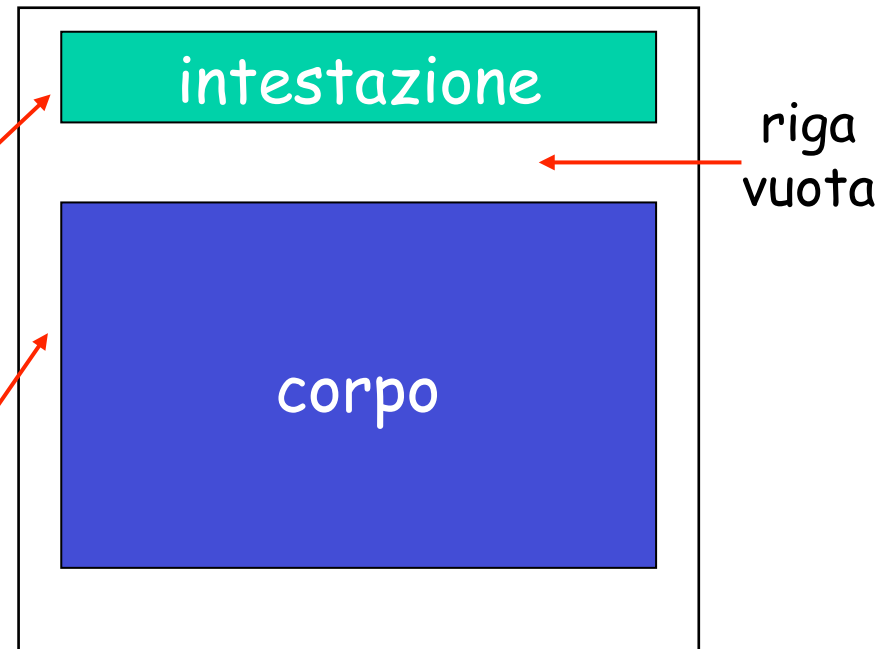
- ❑ Righe di intestazione, per esempio

- ❖ To/A:
- ❖ From/Da:
- ❖ Subject/Oggetto:

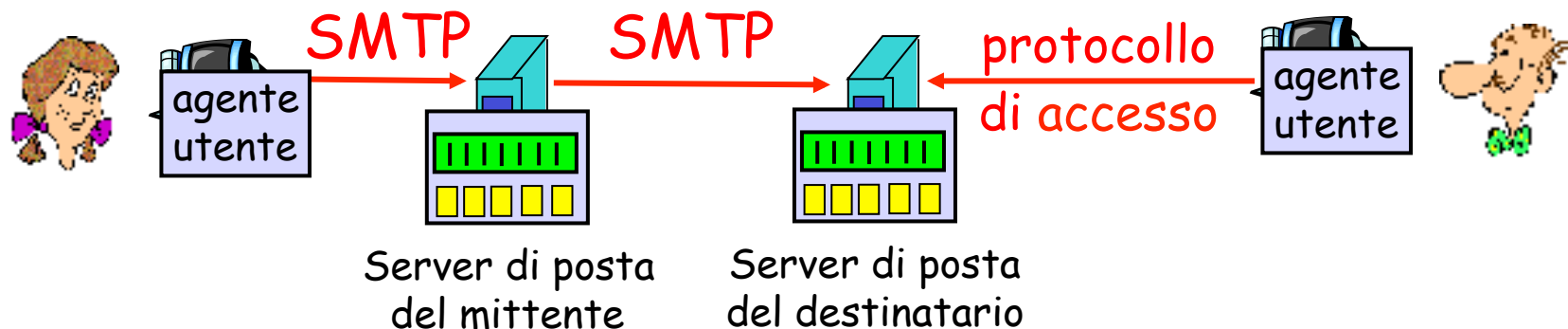
differenti dai comandi SMTP!

- ❑ corpo

- ❖ il "messaggio", soltanto caratteri ASCII



Protocolli di accesso alla posta



- ❑ SMTP: consegna/memorizzazione sul server del destinatario
- ❑ Protocollo di accesso alla posta: ottenere i messaggi dal server
 - ❖ POP: Post Office Protocol [RFC 1939]
 - autorizzazione (agente <--> server) e download
 - ❖ IMAP: Internet Mail Access Protocol [RFC 1730]
 - più funzioni (più complesse)
 - manipolazione di messaggi memorizzati sul server
 - ❖ HTTP: gmail, Hotmail, Libero, ecc.

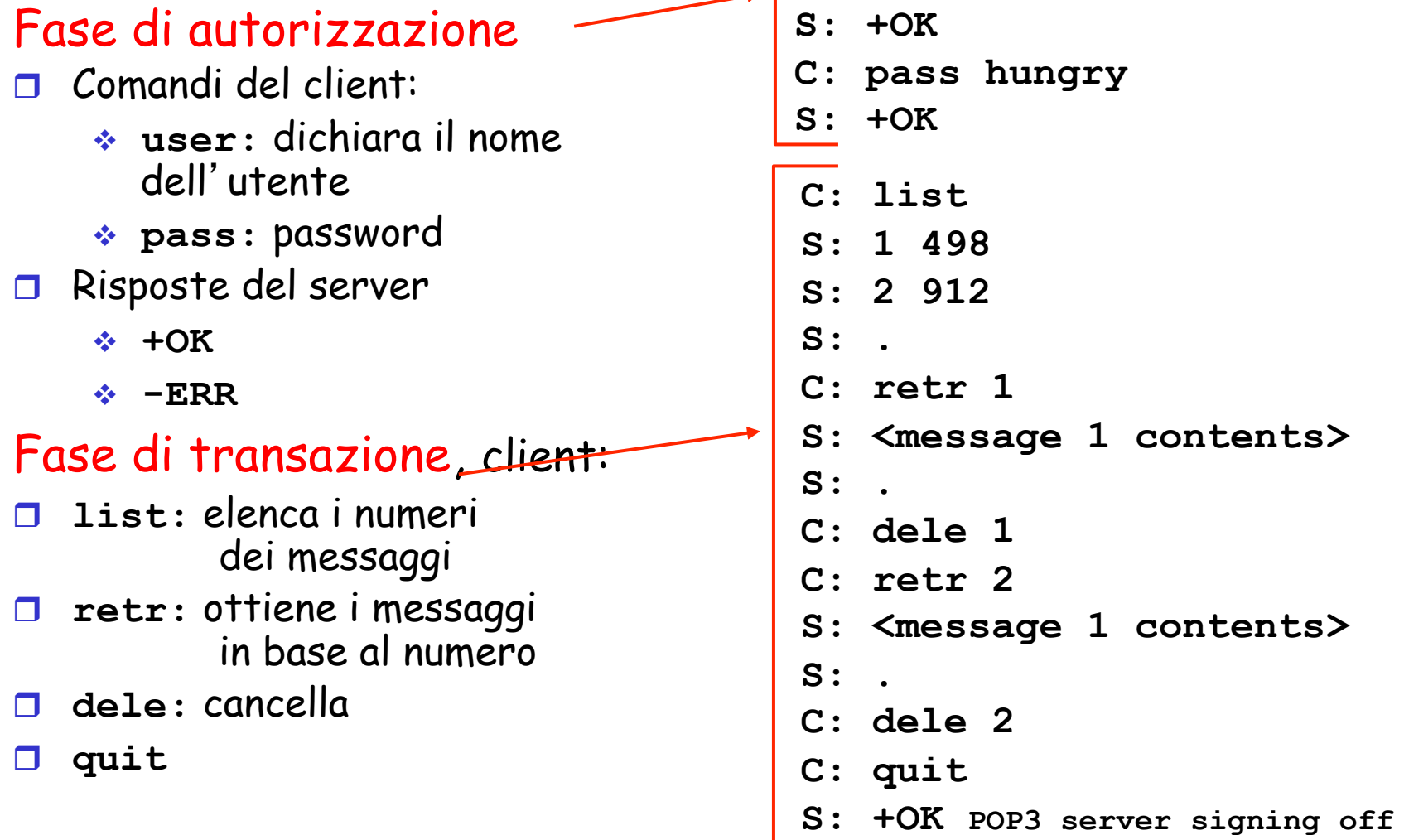
Protocollo POP3

Fase di autorizzazione

- ❑ Comandi del client:
 - ❖ `user`: dichiara il nome dell'utente
 - ❖ `pass`: password
- ❑ Risposte del server
 - ❖ `+OK`
 - ❖ `-ERR`

Fase di transazione, client:

- ❑ `list`: elenca i numeri dei messaggi
- ❑ `retr`: ottiene i messaggi in base al numero
- ❑ `dele`: cancella
- ❑ `quit`



```
S: +OK POP3 server ready
C: user rob
S: +OK
C: pass hungry
S: +OK
```

```
C: list
S: 1 498
S: 2 912
S: .
C: retr 1
S: <message 1 contents>
S: .
C: dele 1
C: retr 2
S: <message 1 contents>
S: .
C: dele 2
C: quit
S: +OK POP3 server signing off
```

POP3 (altro) e IMAP

Ancora su POP3

- ❑ Il precedente esempio usa la modalità "scarica e cancella"
- ❑ Roberto non può rileggere le e-mail se cambia client
- ❑ Modalità "scarica e mantieni": copia i messaggi su più client
- ❑ POP3 è un protocollo senza stato tra le varie sessioni

IMAP

- ❑ Mantiene tutti i messaggi in un unico posto: il server
- ❑ Consente all'utente di organizzare i messaggi in cartelle
- ❑ IMAP conserva lo stato dell'utente tra le varie sessioni:
 - ❖ I nomi delle cartelle e l'associazione tra identificatori dei messaggi e nomi delle cartelle

Capitolo 2: Livello di applicazione

- ❑ Principi delle applicazioni di rete
- ❑ Web e HTTP
- ❑ Posta elettronica
 - ❖ SMTP, POP3, IMAP
- ❑ DNS
- ❑ Applicazioni P2P
- ❑ Programmazione delle socket con TCP
- ❑ Programmazione delle socket con UDP

DNS: Domain Name System

Persone: molti identificatori:

- ❖ nome, codice fiscale, numero della carta d'identità

Host e router di Internet:

- ❖ indirizzo IP (32 bit) - usato per indirizzare i datagrammi
- ❖ "nome", ad esempio, www.yahoo.com - usato dagli esseri umani

D: Come associare un indirizzo IP a un nome?

Domain Name System:

- ❑ *Database distribuito* implementato in una gerarchia di *server DNS*
- ❑ *Protocollo a livello di applicazione* che consente agli host, ai router e ai server DNS di comunicare per *risolvere* i nomi (tradurre indirizzi/nomi)
 - ❖ Si noti: funzioni critiche di Internet implementate come protocollo a livello di applicazione
 - ❖ complessità nelle parti periferiche della rete

DNS

Servizi DNS

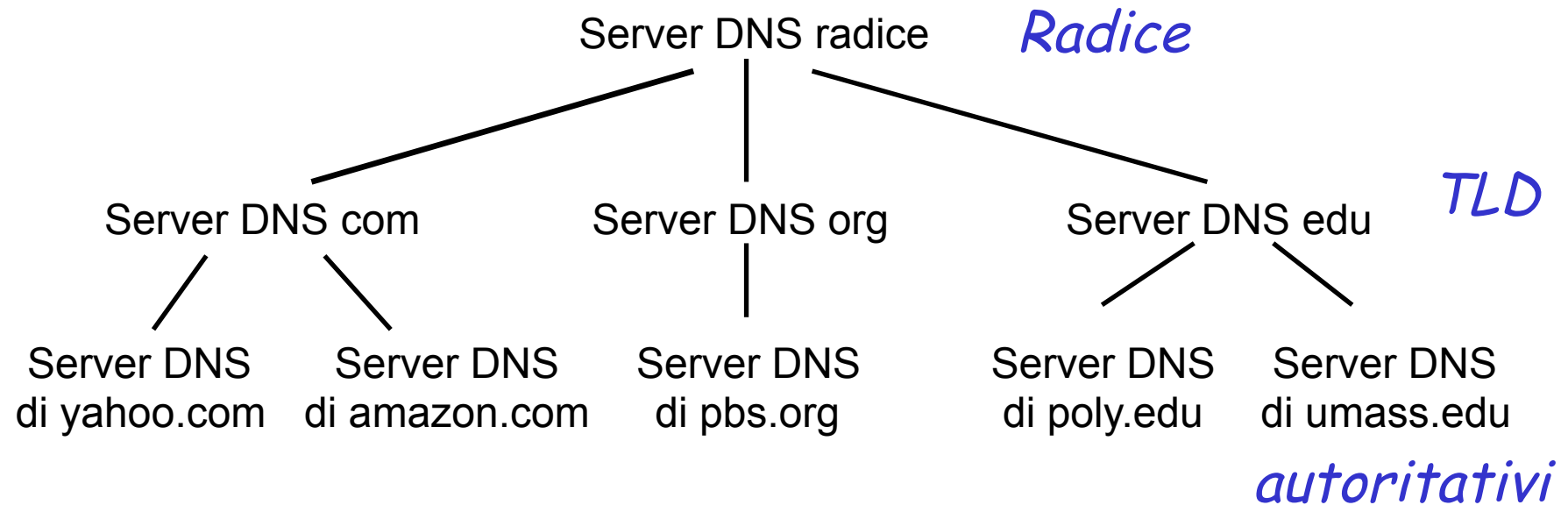
- ❑ Traduzione degli hostname in indirizzi IP
- ❑ Host aliasing
 - ❖ un host può avere più nomi
- ❑ Mail server aliasing
- ❑ Distribuzione locale
 - ❖ server web replicati: insieme di indirizzi IP per un nome canonico

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*!

Database distribuiti e gerarchici



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

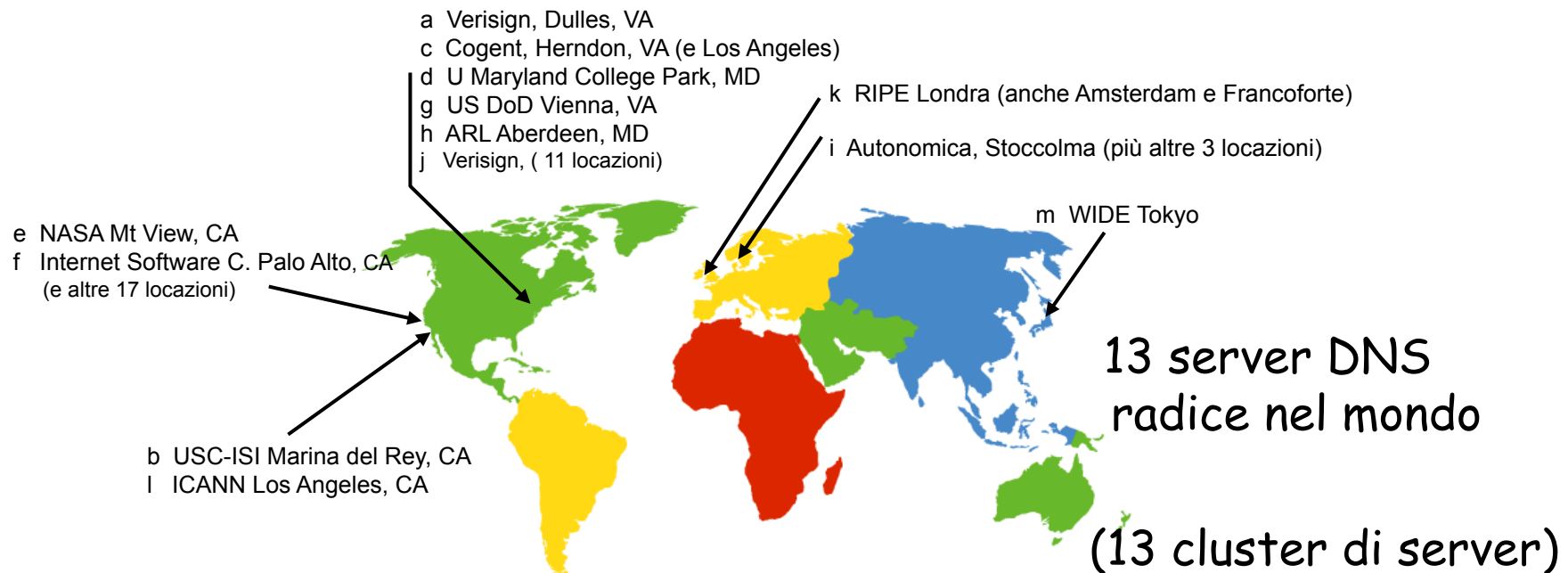
- ❑ Il client interroga il server radice per trovare il server DNS com
- ❑ Il client interroga il server DNS com per ottenere il server DNS amazon.com
- ❑ Il client interroga il server DNS amazon.com per ottenere l'indirizzo IP di www.amazon.com

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

DNS: server DNS radice

- ❑ contattato da un server DNS locale che non può tradurre il nome
- ❑ server DNS radice:
 - ❖ contatta un server DNS autorizzato se non conosce la corrispondenza
 - ❖ ottiene la corrispondenza
 - ❖ restituisce la corrispondenza al server DNS locale



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 uk, fr, ca e jp.
 - ❖ Network Solutions gestisce i server TLD per il dominio com
 - ❖ Educause gestisce quelli per il dominio edu

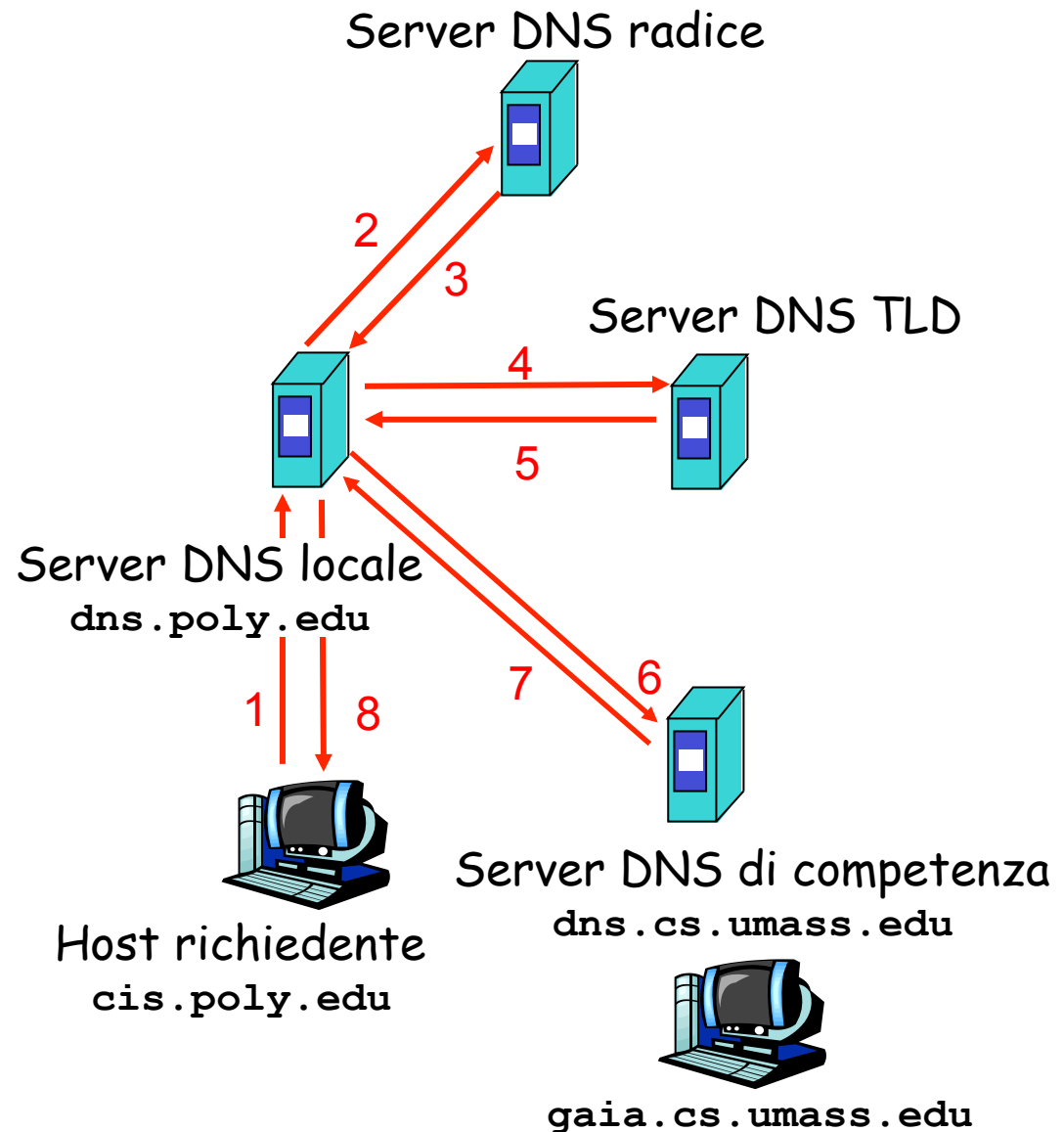
- ❑ **Server di competenza (authoritative server):** ogni organizzazione dotata di host Internet pubblicamente accessibili (quali i server web e i server di posta) deve fornire i record DNS di pubblico dominio che mappano i nomi di tali host in indirizzi IP.
 - ❖ possono essere mantenuti dall'organizzazione o dal service provider

Esempio

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

Query iterativa:

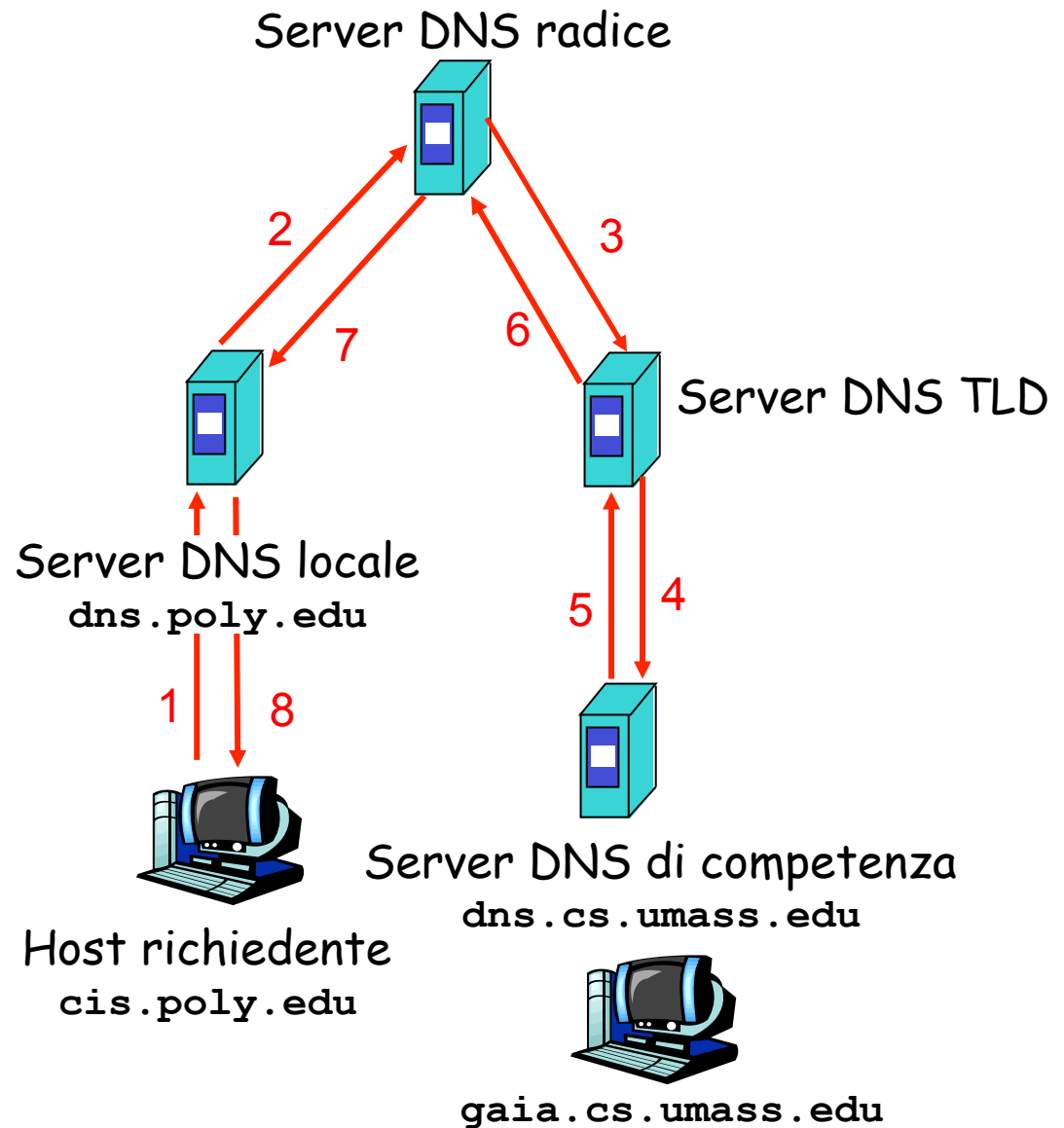
- Il server contattato risponde con il nome del server da contattare
- “Io non conosco questo nome, ma puoi chiederlo a questo server”.



Esempio

Query ricorsiva:

- ❑ Affida il compito di tradurre il nome al server DNS contattato
- ❑ Compito difficile?



DNS: caching e aggiornamento dei record

- Una volta che un server DNS impara la corrispondenza, la mette nella *cache*
 - ❖ le informazioni nella cache vengono invalidate (spariscono) dopo un certo periodo di tempo
 - ❖ tipicamente un server DNS locale memorizza nella cache gli indirizzi IP dei server TLD
 - quindi i server DNS radice non vengono visitati spesso
- I meccanismi di aggiornamento/notifica sono progettati da IETF
 - ❖ RFC 2136
 - ❖ <http://www.ietf.org/html.charters/dnsind-charter.html>

Record DNS

DNS: database distribuito che memorizza i record di risorsa (RR)

Formato RR: (name, value, type, ttl)

□ Type=A

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

□ Type=NS

- ❖ name è il dominio (ad esempio foo.com)
- ❖ value è il nome del server di competenza di questo dominio

□ Type=CNAME

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

www.ibm.com è in realtà

servereast.backup2.ibm.com

- ❖ value è il nome canonico

□ Type=MX

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

Messaggi DNS

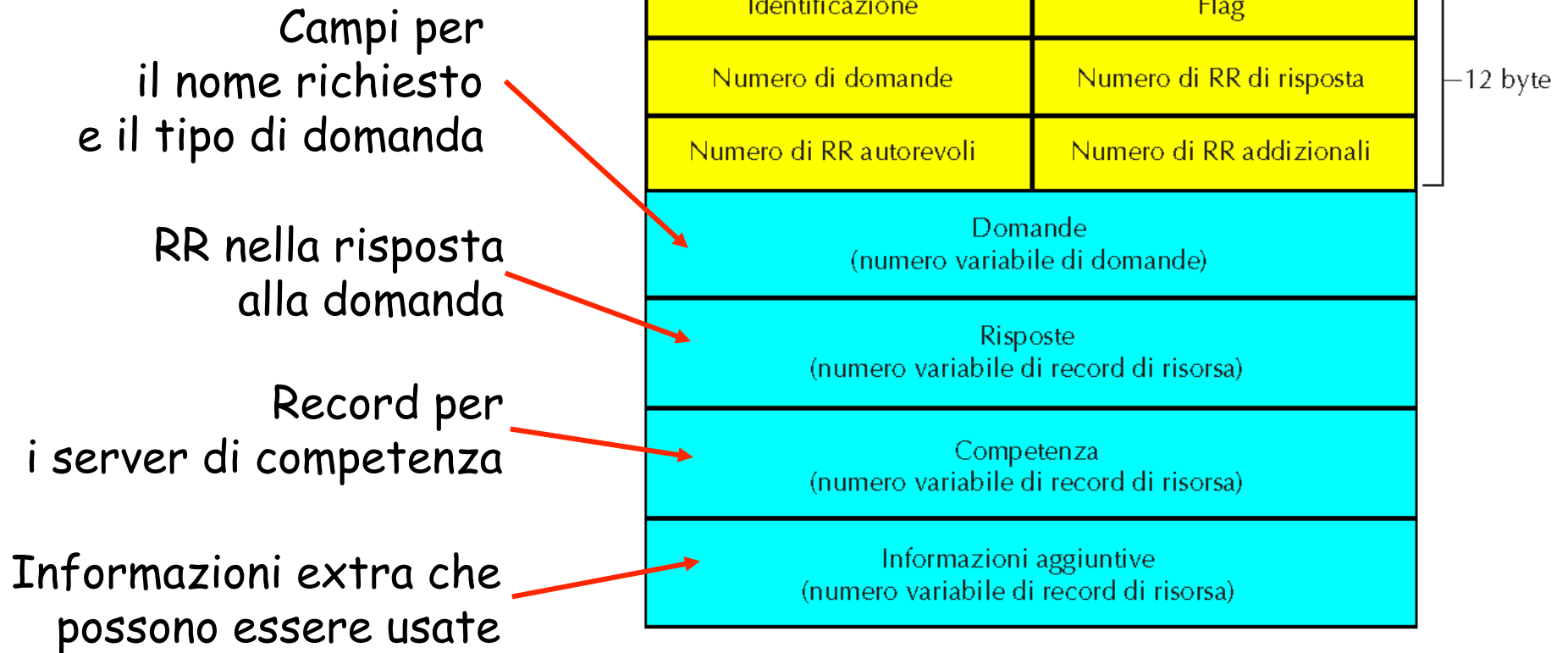
Protocollo DNS: *domande* (query) e messaggi di *risposta*, entrambi con lo stesso *formato*

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

Identificazione	Flag	12 byte
Numero di domande	Numero di RR di risposta	
Numero di RR autorevoli	Numero di RR aggiuntivi	
Domande (numero variabile di domande)		
Risposte (numero variabile di record di risorsa)		
Competenza (numero variabile di record di risorsa)		
Informazioni aggiuntive (numero variabile di record di risorsa)		

Messaggi DNS



Inserire record nel database DNS

- ❑ Esempio: abbiamo appena avviato la nuova società "Network Utopia"
- ❑ Registriamo il nome `networkutopia.com` presso **registrar** (ad esempio, Network Solutions)
 - ❖ Forniamo a registrar i nomi e gli indirizzi IP dei server DNS di competenza (primario e secondario)
 - ❖ Registrar inserisce due RR nel server TLD com:
 - ❖ `(networkutopia.com, dns1.networkutopia.com, NS)`
 - ❖ `(dns1.networkutopia.com, 212.212.212.1, A)`
- ❑ Inseriamo nel server di competenza un record tipo A per `www.networkutopia.com` e un record tipo MX per `networkutopia.com`
- ❑ **In che modo gli utenti otterranno l'indirizzo IP del nostro sito web?**