Lecture 4.

Naming System in the Internet

G.Bianchi, G.Neglia, V.Mancuso

Università degli Studi di Palermo

Three levels of addressing

➔Host names

- ⇨symbolic name, arbitrary length, arbitrary # of fields
- ⇨Why names at all?
  - →Numeric addr. tough for humans to remember
  - →Numeric addr. impossible to guess
- ⇨Problems with names
  - →Variable length (machines prefer fixed length)
  - →Potentially long

cerbero.elet.polimi.it

131.175.21.1

A3:B2:32:31:8:4A

➔IP address

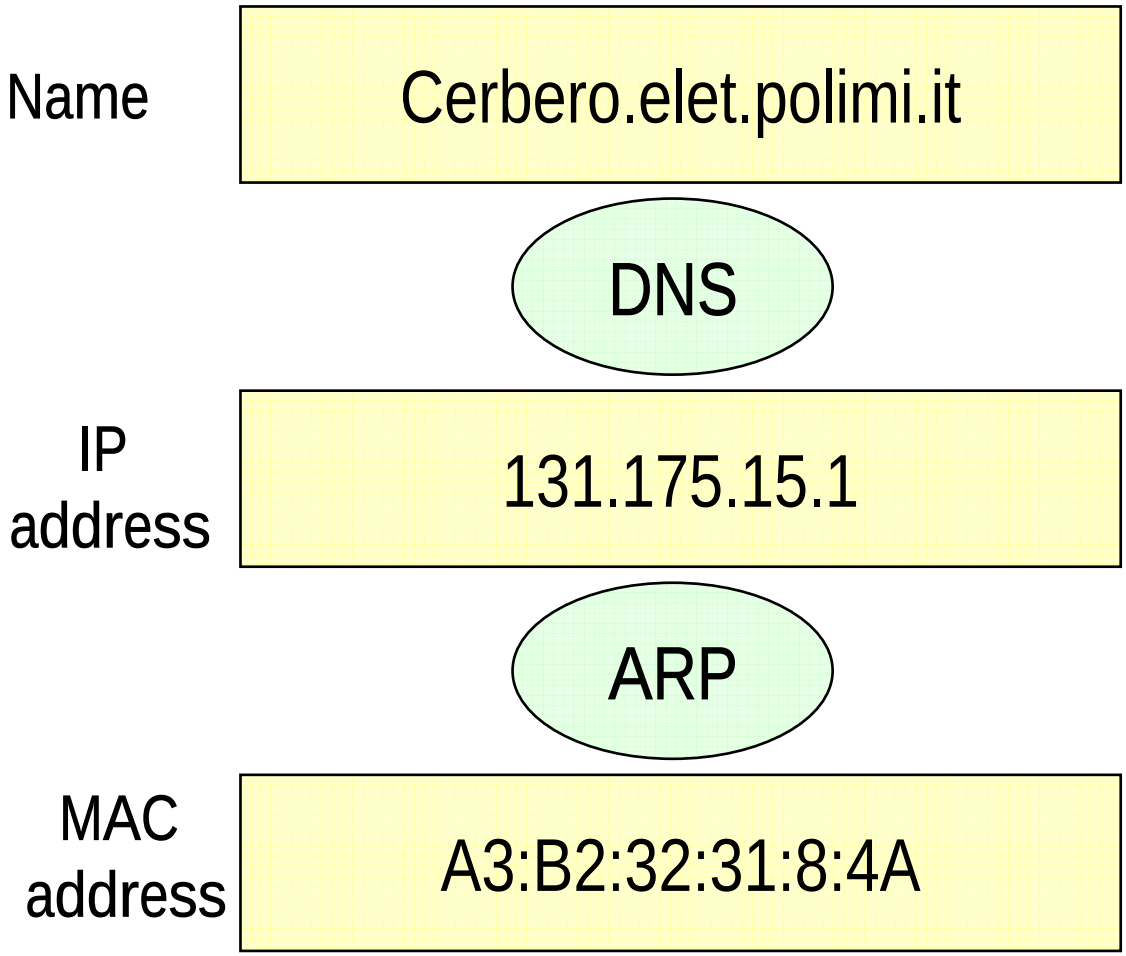
- ⇨32 bits (4 bytes)
- ⇨Structured for routing!

➔MAC address

- ⇨hw address assigned to interface card
- ⇨Ethernet (802): 48 bits (6 bytes)

Name versus address separation:
provides a level of indirection: network administrators can move computers around networks (changing address and not name)

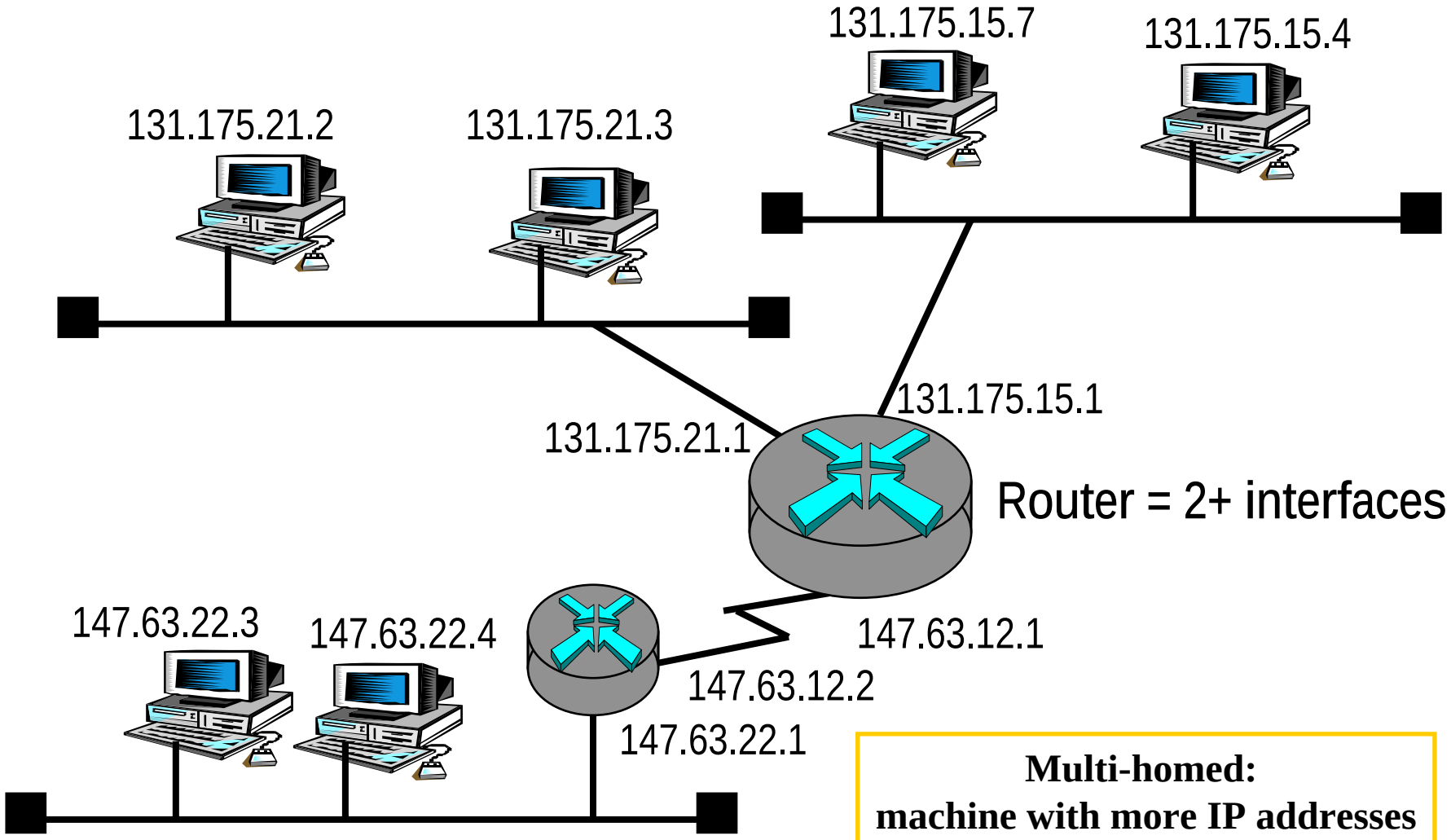
Name conversion protocols



- **Q: Does each name, IP address & MAC address uniquely specifies a machine?**
- **A: Yes and No**
  - eg: try to resolve www.yahoo.com
  - load balancing techniques: many machines associated to a very loaded name

Address Association

IP Address associated to an interface (not to a host)



Multi-homed:
machine with more IP addresses
(not necessarily routers)

DNS - Domain Name System

basics RFCs:
1034, 1035

updates & technical RFCs
1591, 2136, 2181, 2182, 2535

most recent:
RFC 2929 (BCP42), 2930, sep 2000

Naming in the ARPANET: not a problem

(few tens of nodes)

- **Unique file HOSTS.TXT maintained by NIC**
- **unix stations copied at installation HOSTS.TXT into /etc/hosts**
- **every night: hosts fetch file HOSTS.TXT and updated local copy**

HOSTS.TXT

rolf	234
mark	321
john	123
milind	23
……	……

Naming in the multimillion hosts Internet: two problems

- **efficient name assignment**
  - hierarchical naming space
  - decentralizes name assignment while avoiding name contention
    - abc.kkk.it differs from abc.kkk.de
  - practically unlimited naming space
- **efficient name resolution**
  - distributed database approach, to avoid enormous HOSTS.TXT file!
  - fundamental to avoid performance impairments and distribute load
  - problems: consistency & efficiency

What the Internet's DNS is

- **A systematic namespace (domain name space)**
  - independently managed by different people/organizations
- **a distributed database system**
  - implemented in a hierarcy of DNS servers
- **an application-layer protocol**
  - protocols that allow retrieval of information
  - protocol that allow sinchronization between servers
  - conventions for using the information

What DNS does?

- **Map hostname to IP address**
- **Map IP address to hostname**
- **provide email routing information**
  - ⇨bianchi@elet.polimi.it --> morgana.elet.polimi.it
- **handle aliases**
  - ⇨ftp.elet.polimi.it --> fusberta.elet.polimi.it
- **(load distribution)**

DNS
as an application-level protocol

- **based on client-server paradigm**
- **default port number for DNS service: 53**
- **basically runs over UDP (but uses also TCP):**
  - ⇨ TCP for transfers of entire database to secondary servers (replication).
  - ⇨ UDP for lookups
  - ⇨ If more than 512 bytes in response - requestor resubmits request using TCP.

What are domain names used for?

- **To identify computers (hosts) on the Internet**
  - →morgana.elet.polimi.it
- **To identify organisations**
  - →bianchi.it *(bycicles….)*
- **To map other information to a form that is usable with the DNS infrastructure**
  - →IP addresses, Telephone numbers, ...

DNS hierarchy

➔Dotted notation

⇨names from right to left SHOULD indicate a naming hierarchy

Other rules:

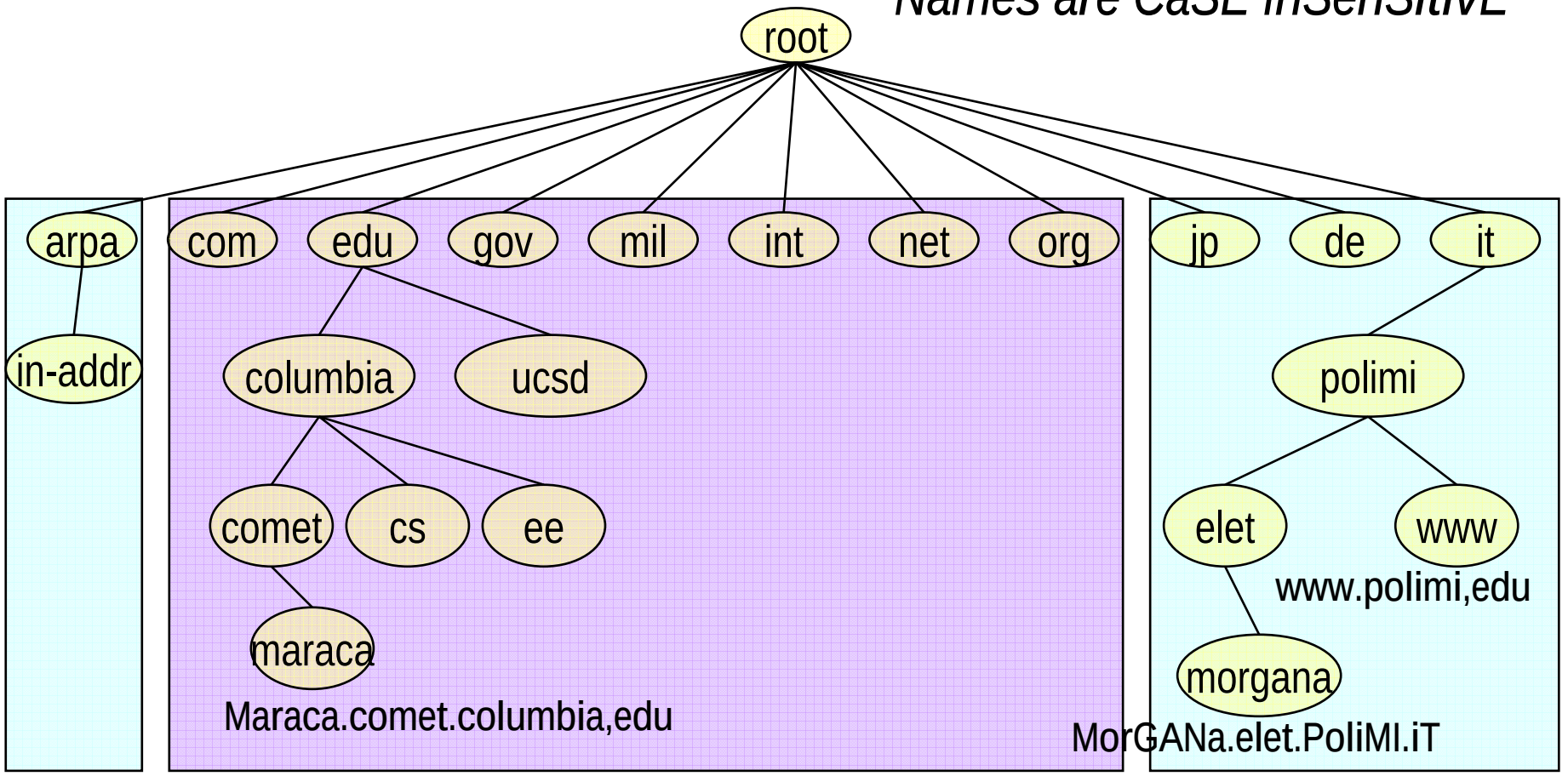
- component names = 63 characters long

- full path names must not exceed 255 characters

Domain Name Space

a portion

Names are CaSE InSenSitivE



Arpa domains

Generic domains

Country domains

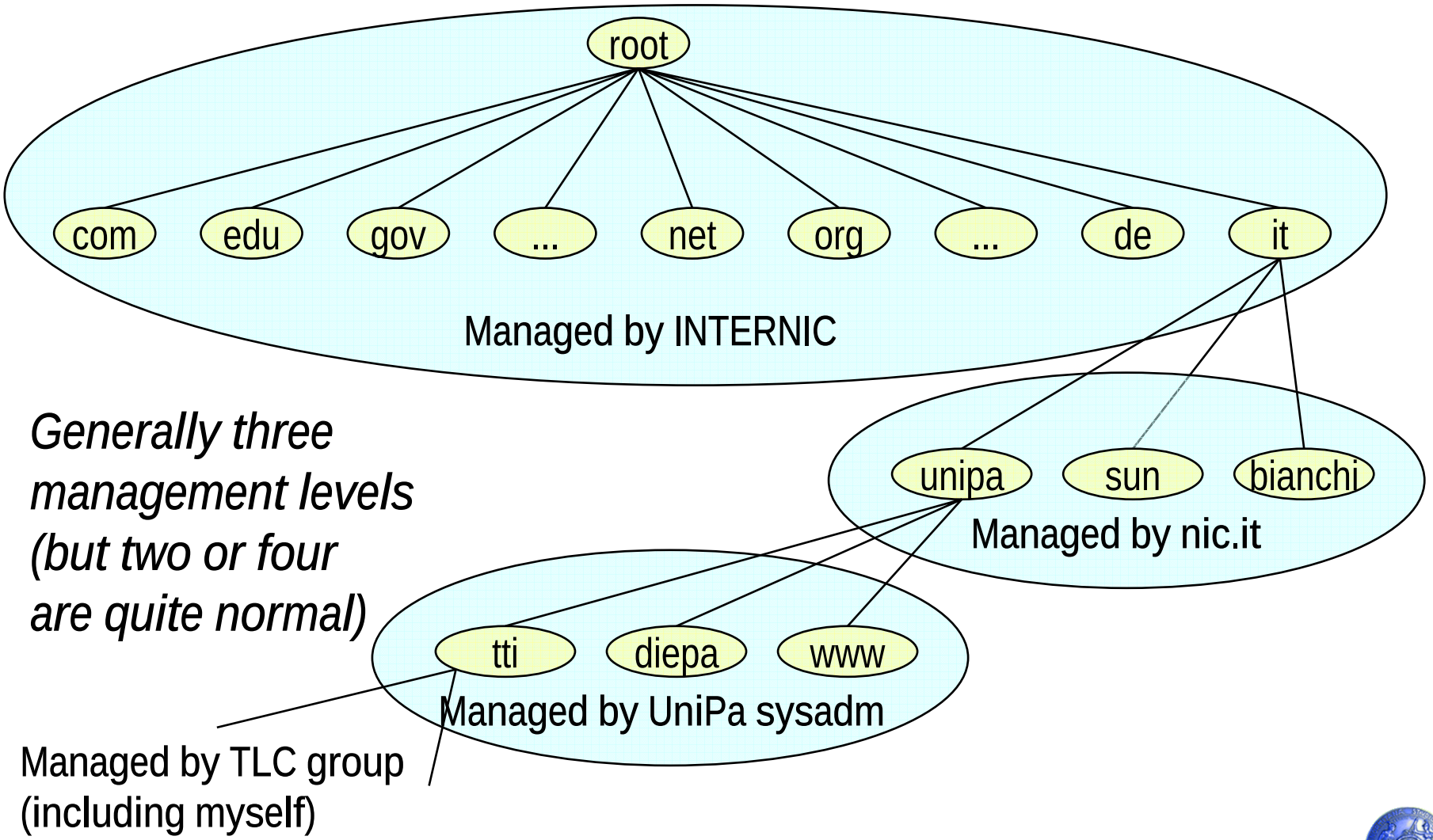
Top level domains (re)organization

classificazione per tipologia	
Top level domain name	**Tipo di organizzazione**
COM	Commerciali
EDU	Accademiche e didattiche
GOV	Statali
MIL	Militari
NET	Centri di Gestione di Internet
ARPA	ARPANET (obsoleto)
INT	Organizzazioni internazionali
ORG	Altre organizzazioni
AERO	industria trasporti aerei
ASIA	
BIZ	business
CAT	comunità catalana
COOP	Associazioni cooperative
...	
...	

new

http://www.iana.org/gtld/gtld.htm

DNS Management



Generally three management levels (but two or four are quite normal)

.it situation

- **www.nic.it**
  - registration authority italiana
  - naming authority italiana
- **413.081 registered domains**
  - (as of dec 12, 2000)
- **nameservers:**
  - dns.nic.it (primary)
  - nameserver.cnr.it — server2.infn.it
  - nsripe.net (holland) — dns2.it.net
  - dns2.iunet.it — ns2.psi.net (USA)
  - ns.eu.net (holland)
- **root server (from morgana.elet.polimi.it)**
  - a.root-servers.net — 198.41.0.4

Domain Name Concepts

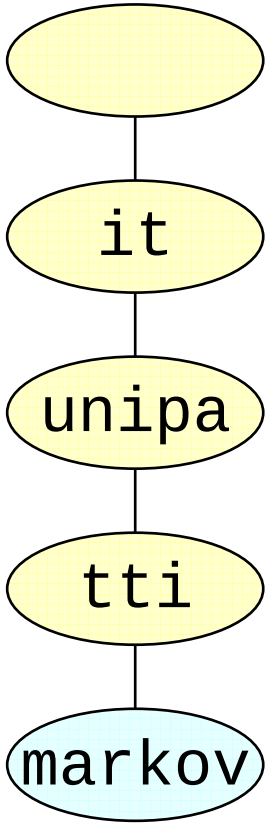
➔**domain name:** the sequence of labels that lead from the host to the top of the worldwide naming tree.

➔**Domain:** subtree of the worldwide naming tree.

⇨It unipa.it tti.unipa.it

➔**Node:**

⇨markov.tti.unipa.it



Different uses of the term "domain"

- **Sometimes, the term "domain" is used to refer to a single name**
  - ⇨such as polimi.it
- **Sometimes, the term "domain" is used to refer to all the names (subdomains) that are hierarchically below a particular name**
  - ⇨in this usage, the polimi.it domain includes elet.polimi.it, math.polimi.it, etc.

Concept of Zone

➔**NON OVERLAPPING sub-tree for which naming authority has been delegated**

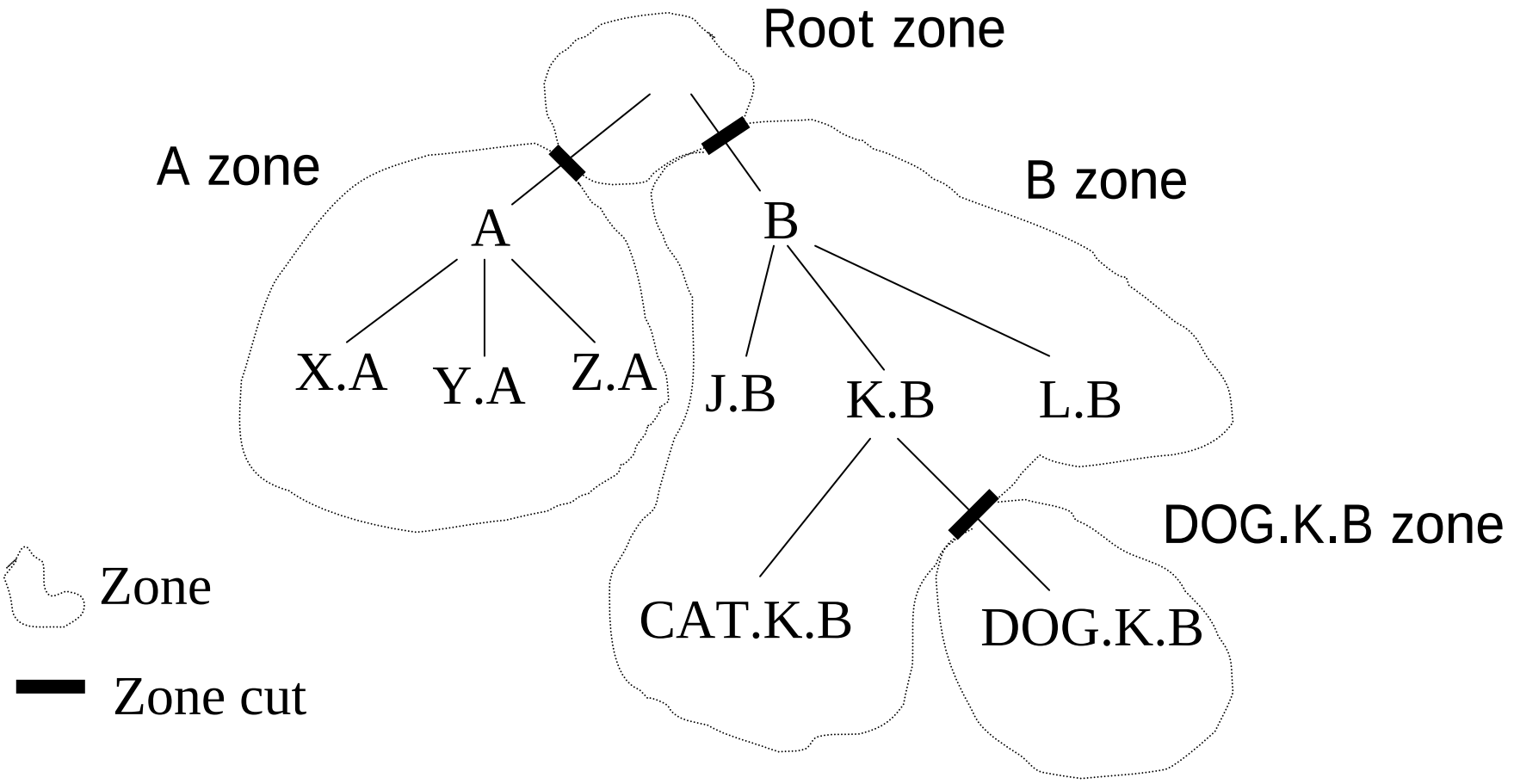
➔**Think of the namespace as a tree or graph of nodes joined by arcs**

⇨Each node represents a domain name

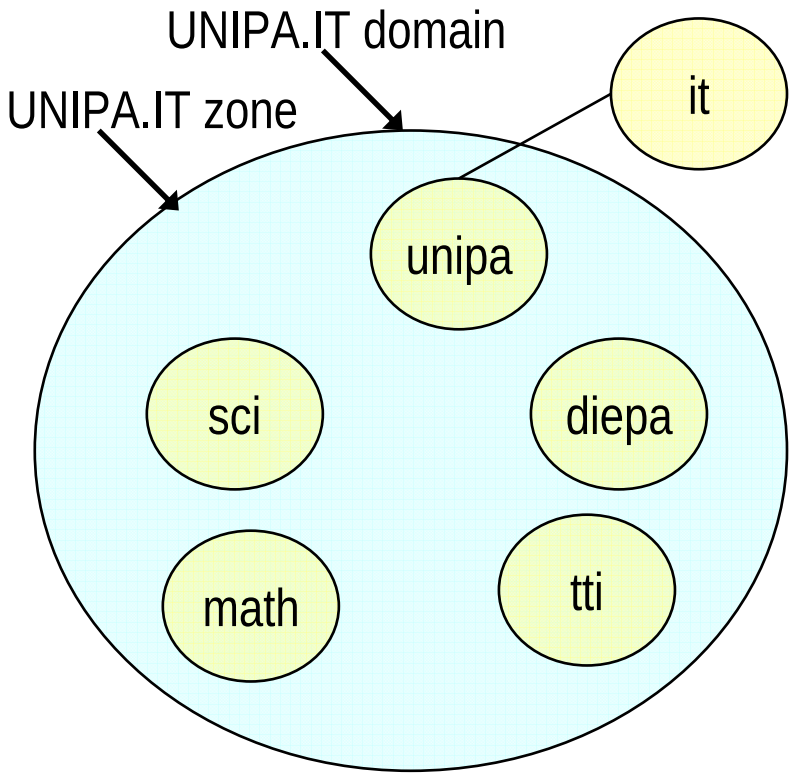
➔**Now cut some of the arcs**

⇨Each cut represents a delegation of administrative control

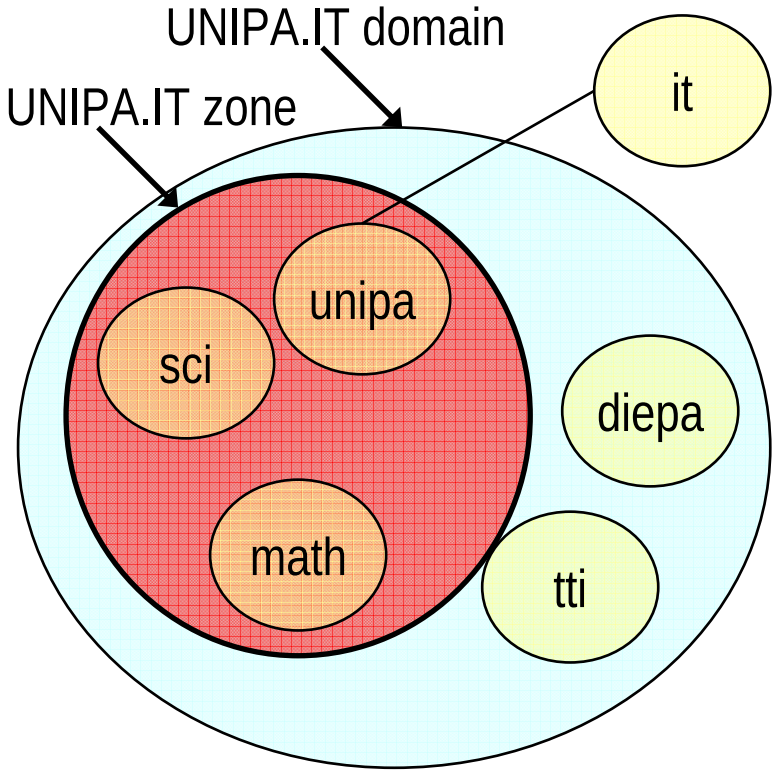
Concept of zone



Zones vs Domains



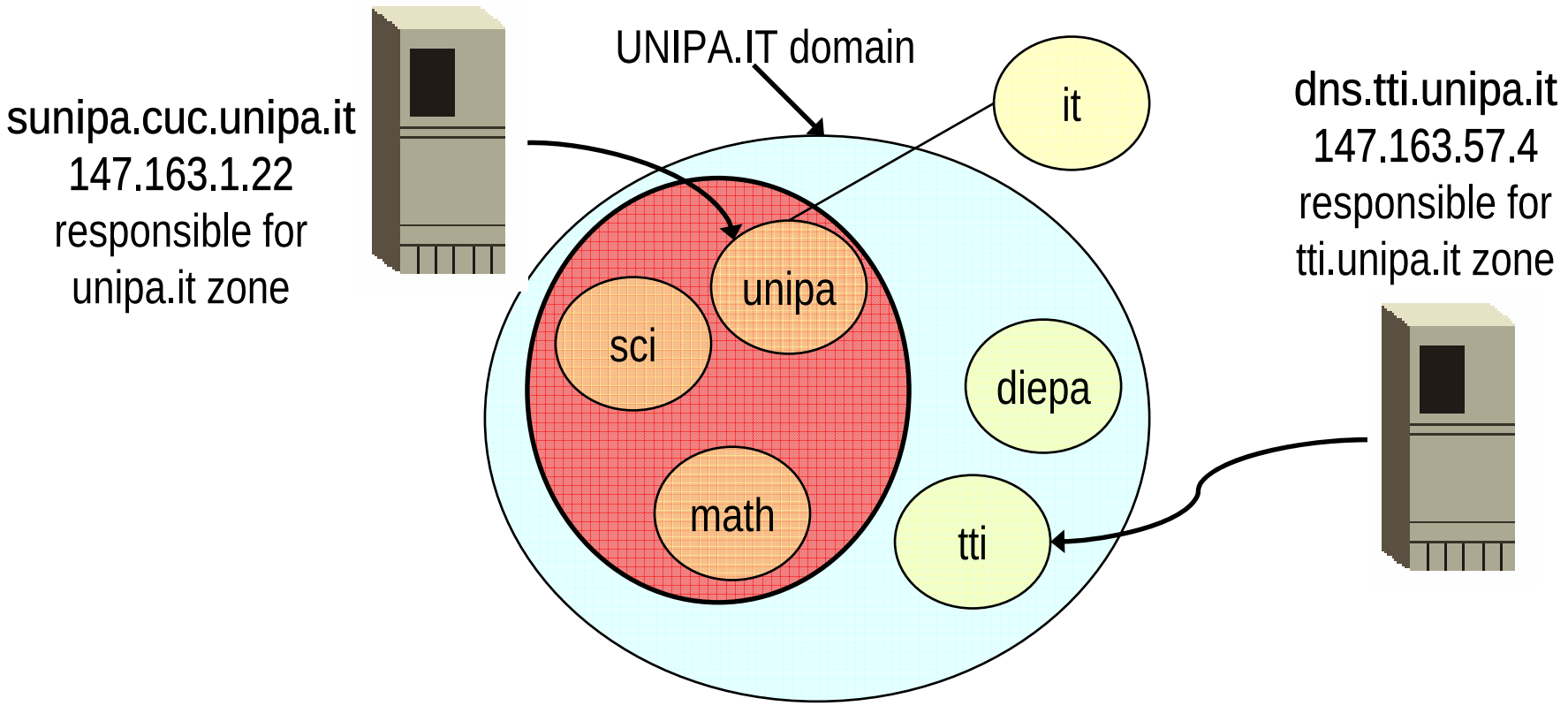
Case 1: single DNS administration
(not the real case)



Case 2: diepa and tti have
authority for their zones

Name Server

server that store information about the zone



Authoritative server: provides "original" information for a zone
ALWAYS capable to resolve name-Ipaddr association

DNS servers

- **Root DNS servers**
  - 13 servers (www.root-servers.org)
- **Top-Level Domain (TLD) servers**
- **Authoritative servers**
  - for a specific zone
- **Local DNS server**

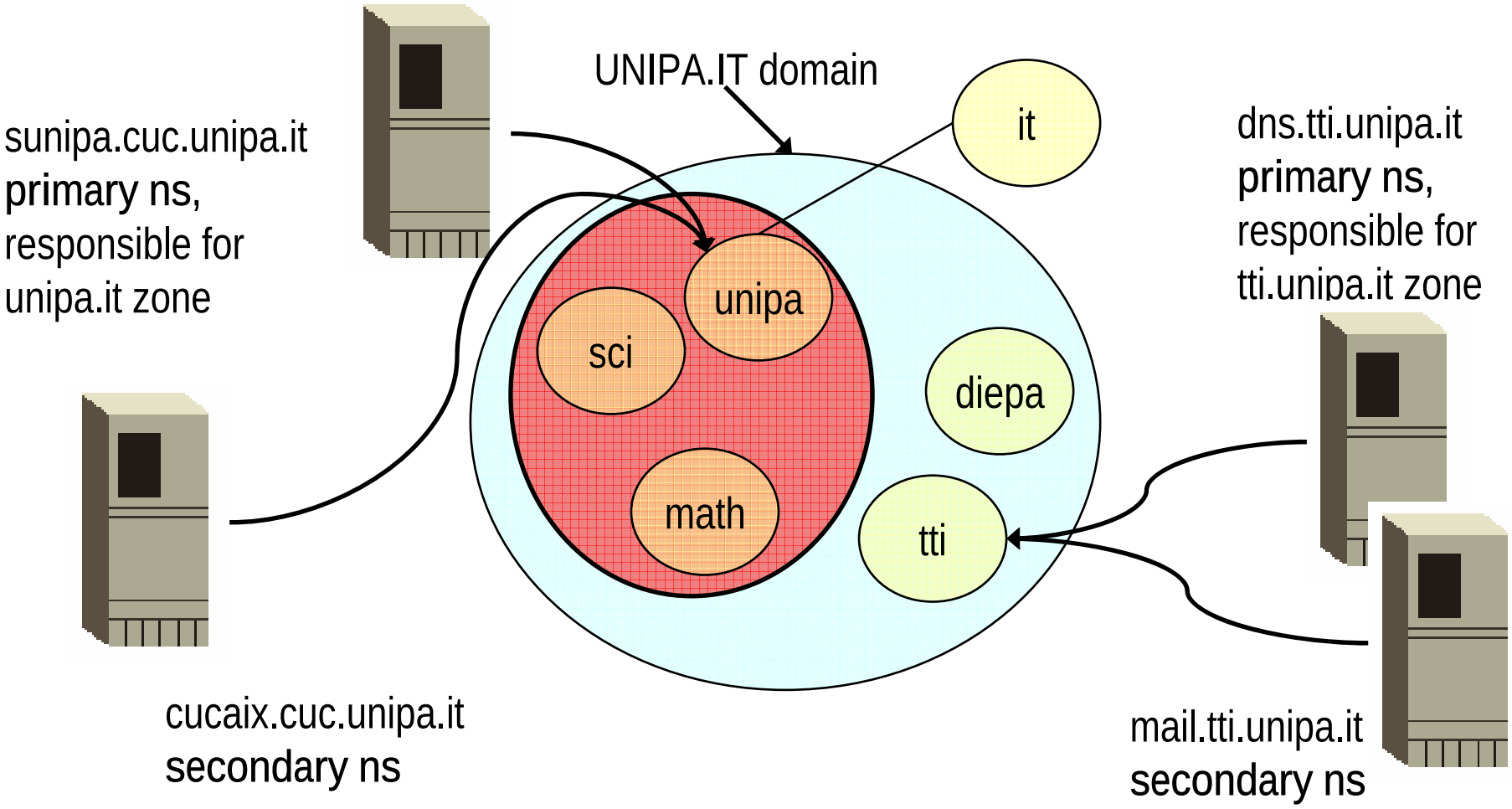
Name server requirements

- ⇨A query should be resolved as fast as possible;
- ⇨It should be available 24 hours a day;
- ⇨It should be reachable via fast communication lines;
- ⇨It should be located in the centre of the network topology;
- ⇨It should be robust, without errors and interrupts.

➔**For reliability, at least two DNS servers per *zone* are *mandatory***

- ⇨One primary or master
- ⇨One or more secondaries or slaves
- ⇨Slaves periodically update from master

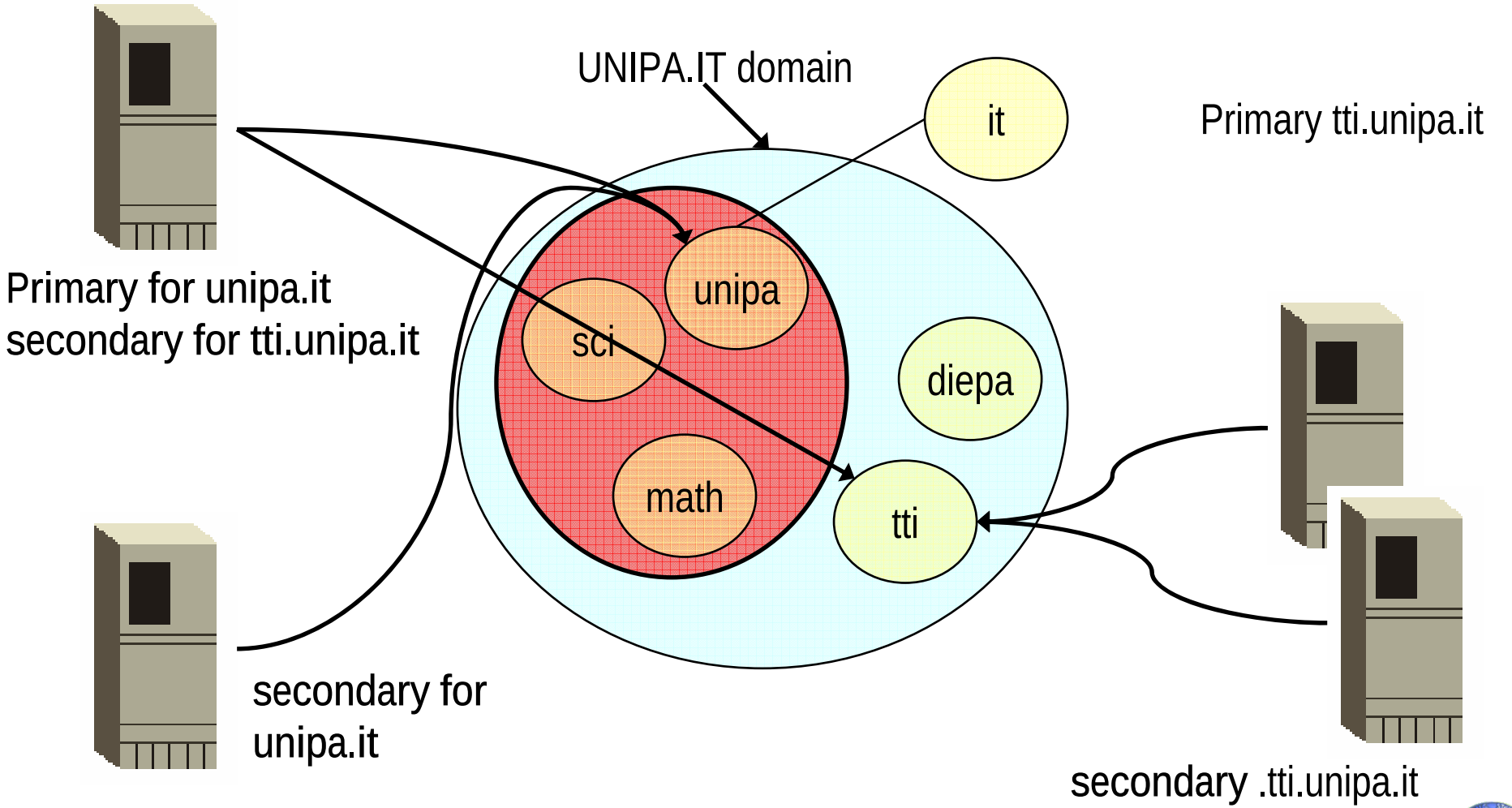
Primary & secondary Name Servers



Reliability

- ➔If one server does not reply, clients will ask another server
- ➔That's why there are several servers for each zone
- ➔may be (and generally are) on same network: not recommended! See RFC 2182 (SELECTION AND OPERATION OF SECONDARY DNS SERVERS)
- ➔At least avoid a single point of failure

More secondary Name Servers (meshing allowed)



DNS resolution

A distributed database

DNS Clients

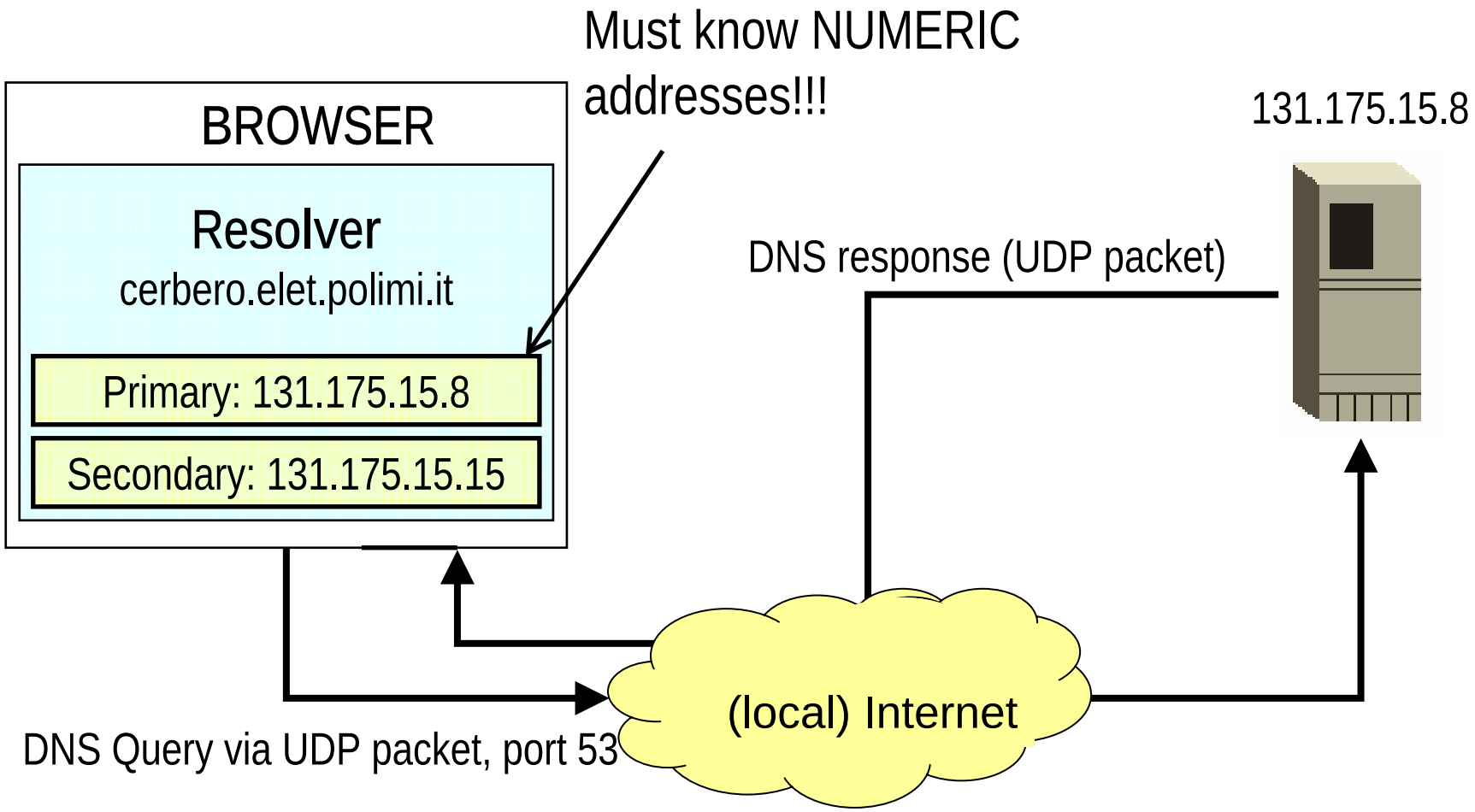
- **A DNS client is called a *resolver*.**
- **A call to `gethostbyname()` is handled by a resolver (typically part of the client).**
- **Most Unix workstations have the file /etc/resolv.conf that contains the local domain and the addresses of DNS servers for that domain.**

Example:
/etc/resolv.conf
cerbero.elet.polimi.it
131.175.15.1

Domain	**elet.polimi.it**
nameserver	**131.175.21.8**
nameserver	**131.175.21.1**
nameserver	**131.175.12.1**

DNS resolution

(when our name server can make it alone...)



Why DNS is a distributed DB

- ➔Thousands of servers around the world
- ➔Each server has authoritative information about some subset of the namespace
- ➔There is no central server that has information about the whole namespace
- ➔**If a question gets sent to a server that does not know the answer, that is not a problem**

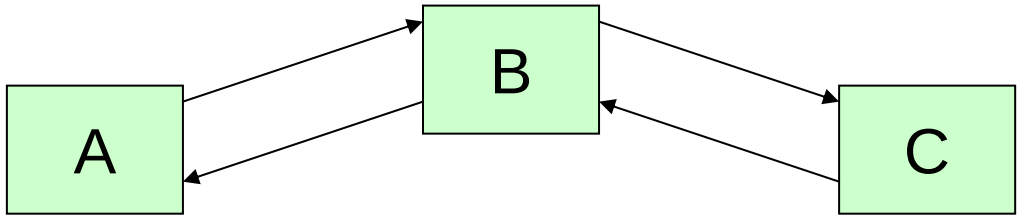
When the local nameserver is not able to resolve

- ➔If a server has no clue about where to find the address for a hostname, ask the root server.
- ➔The root server will tell you what nameserver to contact.
- ➔A request may get forwarded a few times.

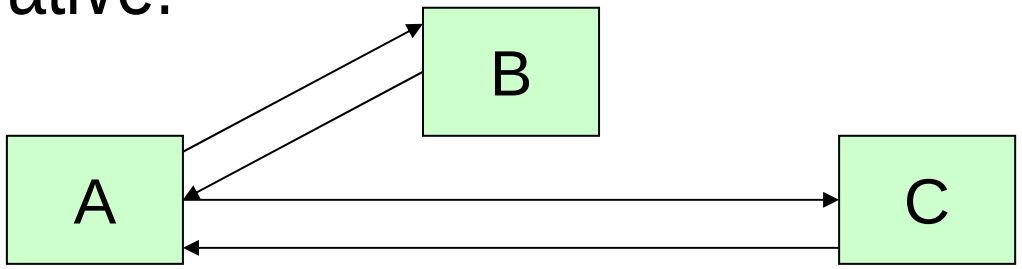
Interaction

➔ ***name servers*** **interaction may be:**

⇨recursive:



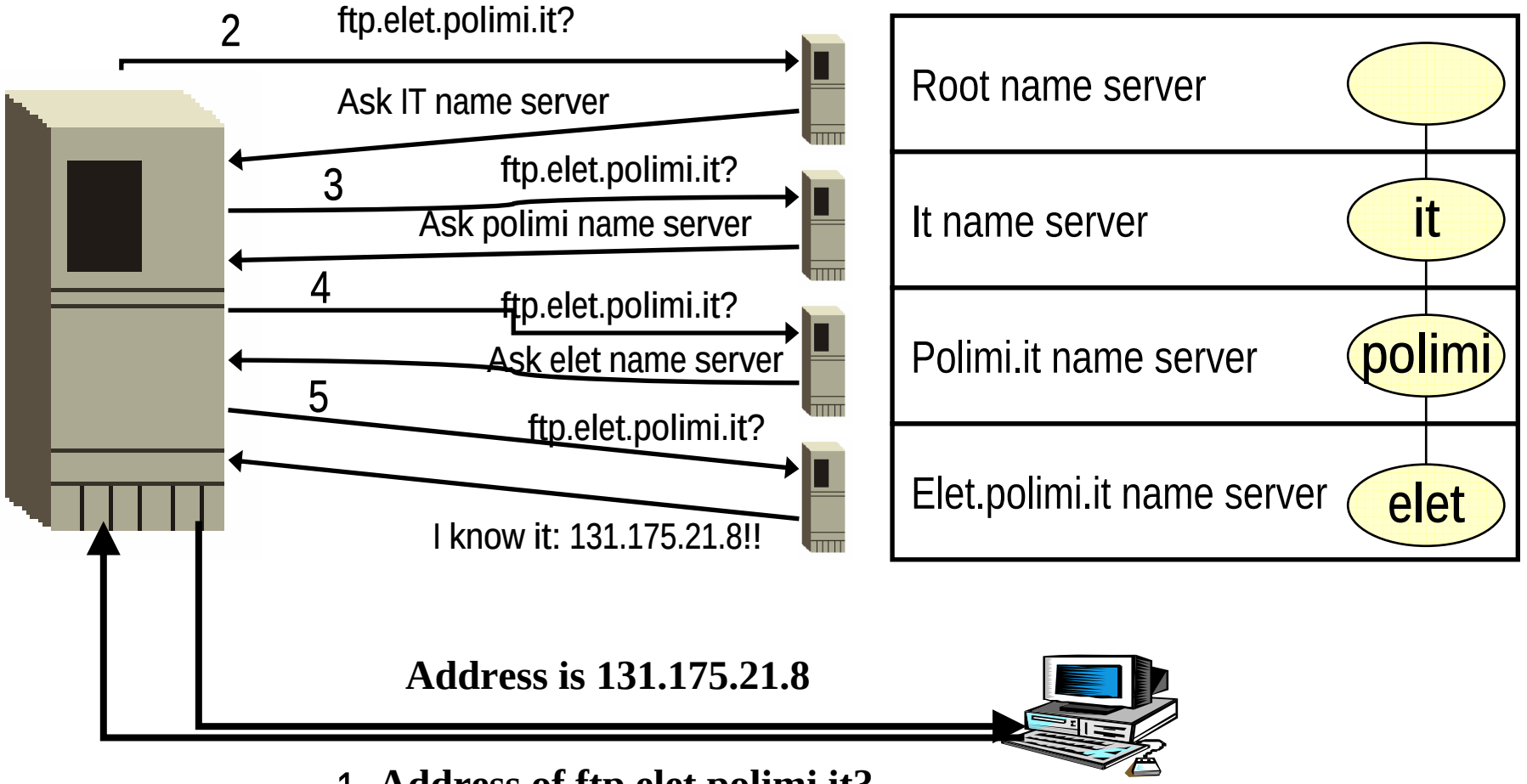
⇨iterative:



DNS resolution

(when our NS has no idea...)

(recursive + iterative approach)



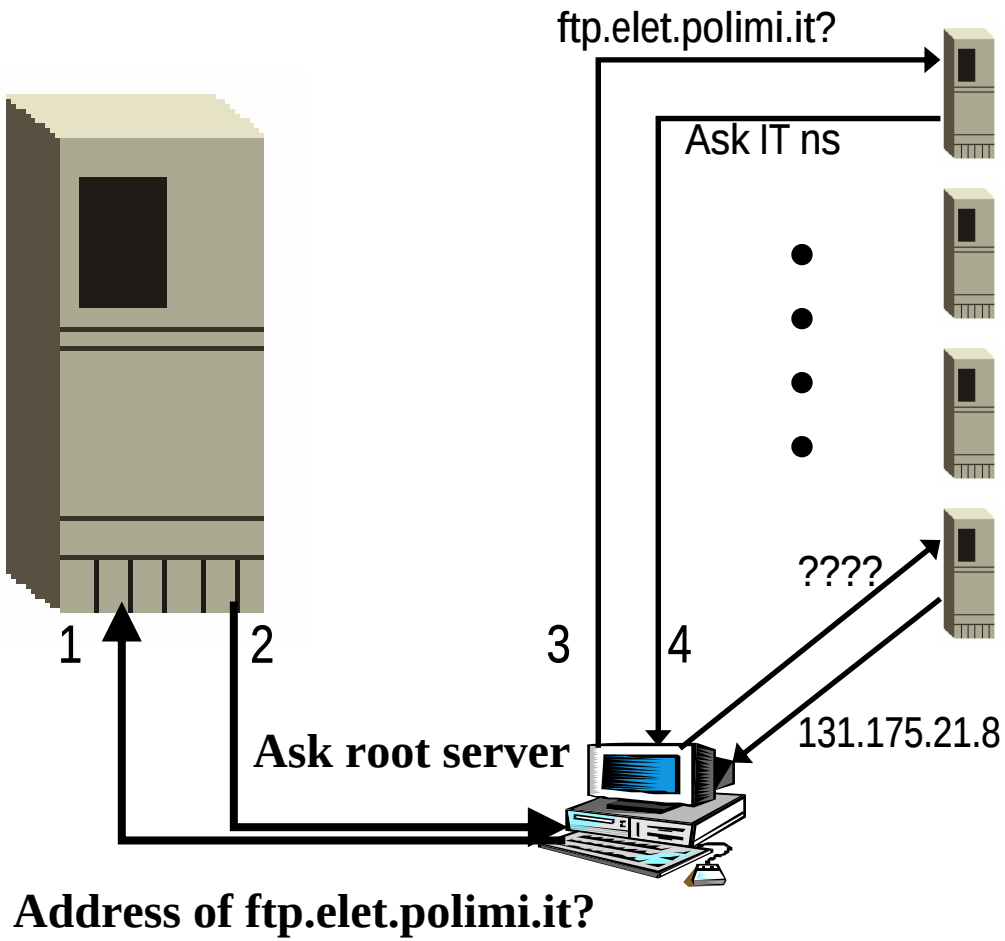
Information needed

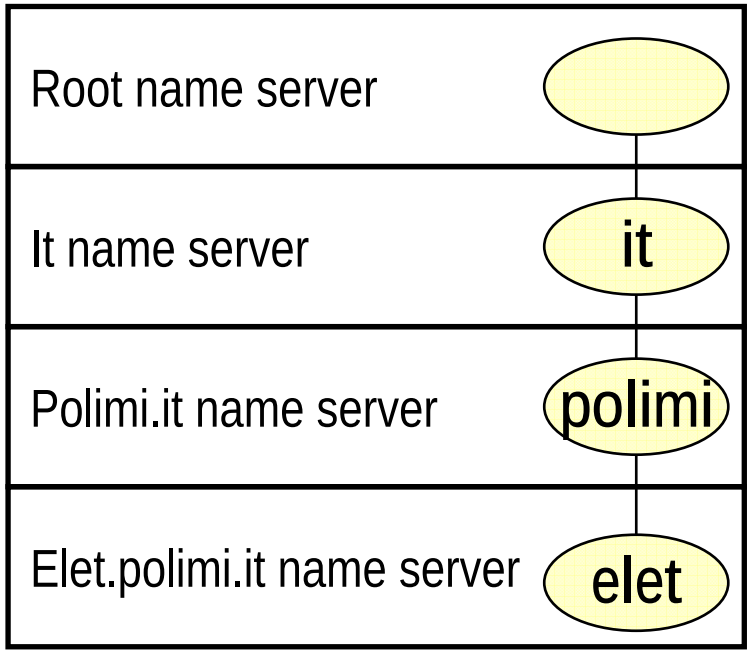
➔**To work correctly, each name server must know:**

- ⇨the IP address of all the “children” hosts (i.e. in the domain)
- ⇨the IP address of the name servers of each subdomain (when they are name zones)
- ⇨the root name server (not mandatory: should be known by resolver)
  - » Updated list (13 root name servers as of aug 1997) at **ftp://ftp.rs.internic.net/domain/named.root**

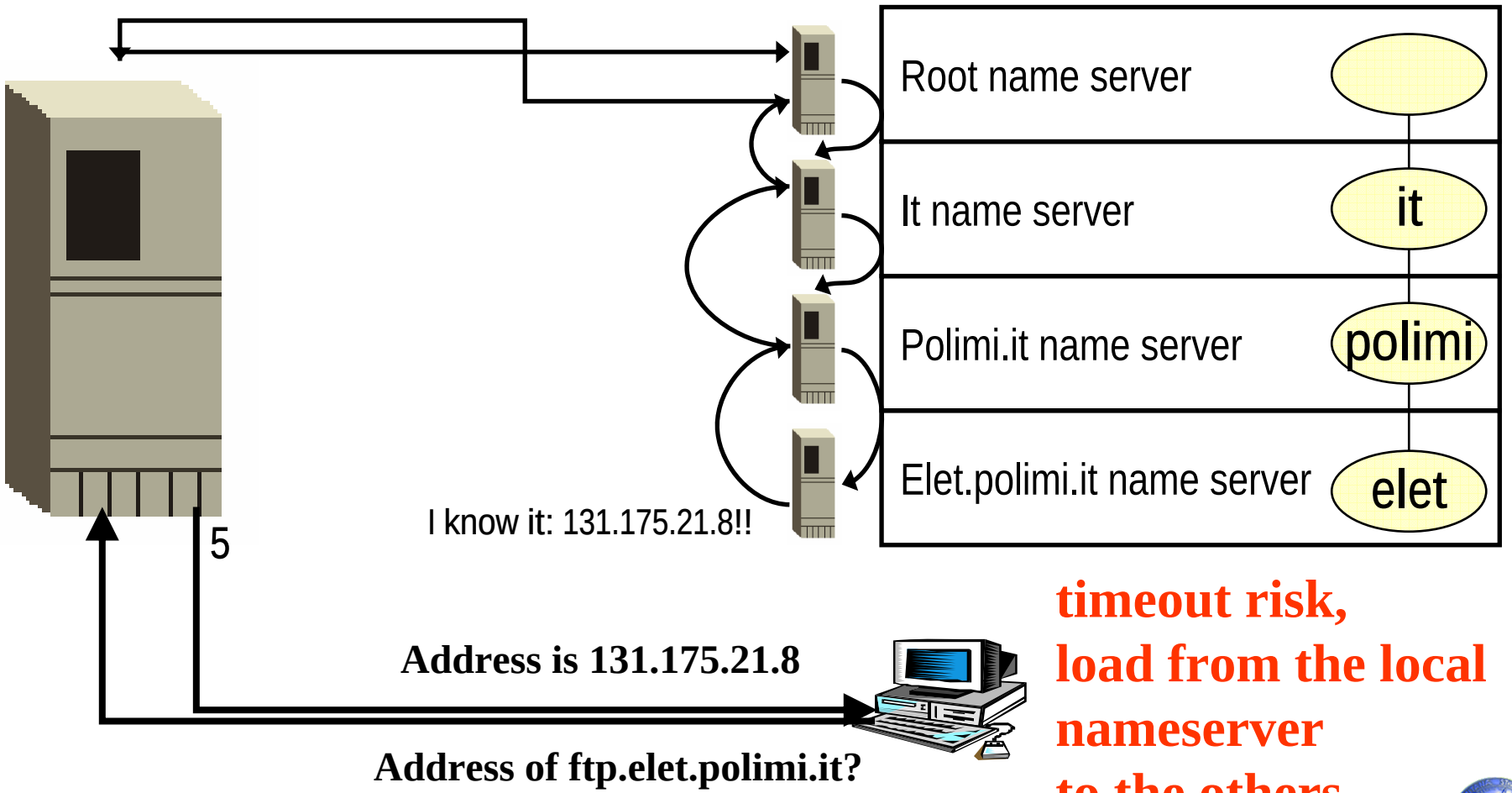
DNS resolution (pure iterative)

some servers may not implement recursive resolution



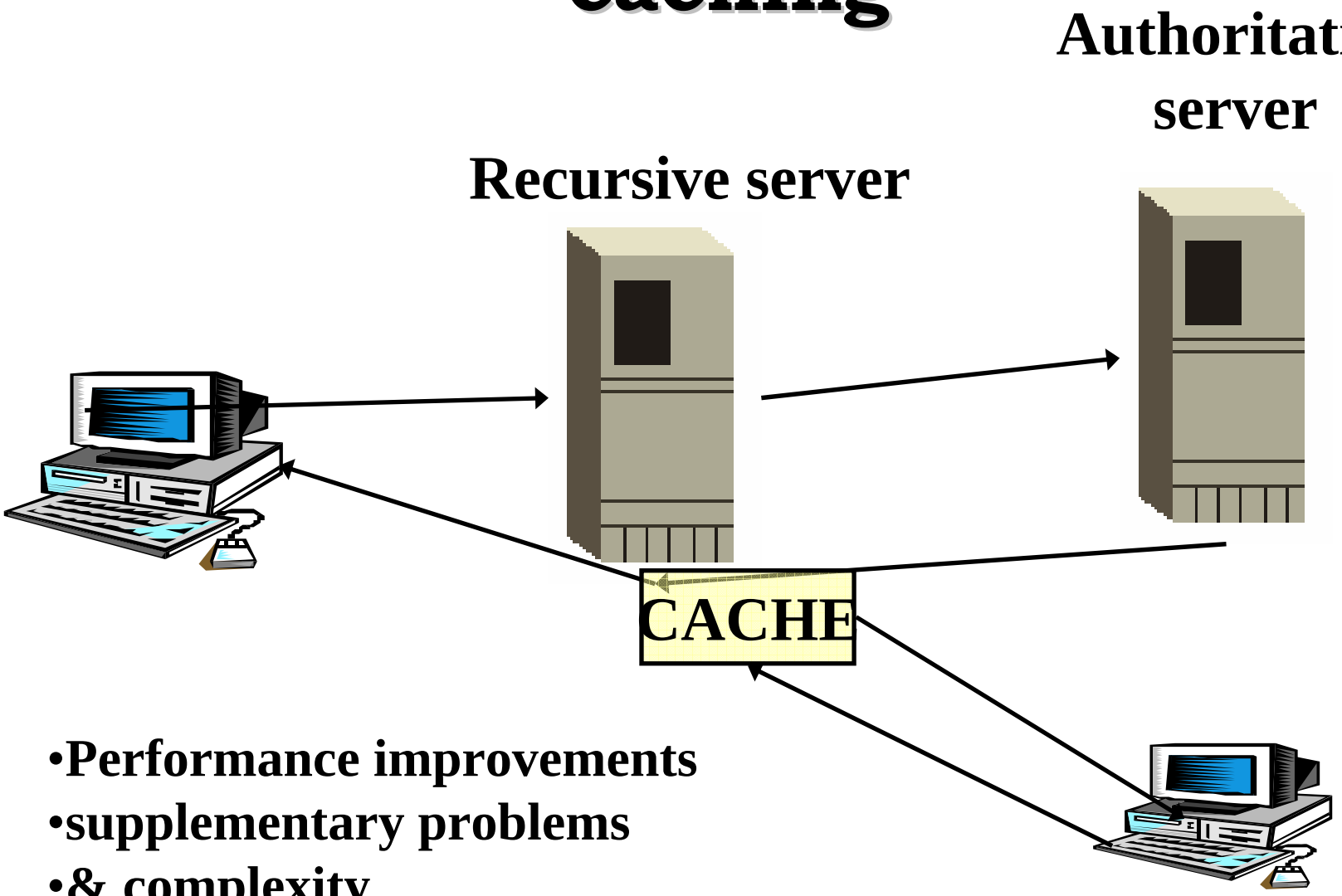


DNS resolution (pure recursive)



timeout risk, load from the local nameserver to the others

caching

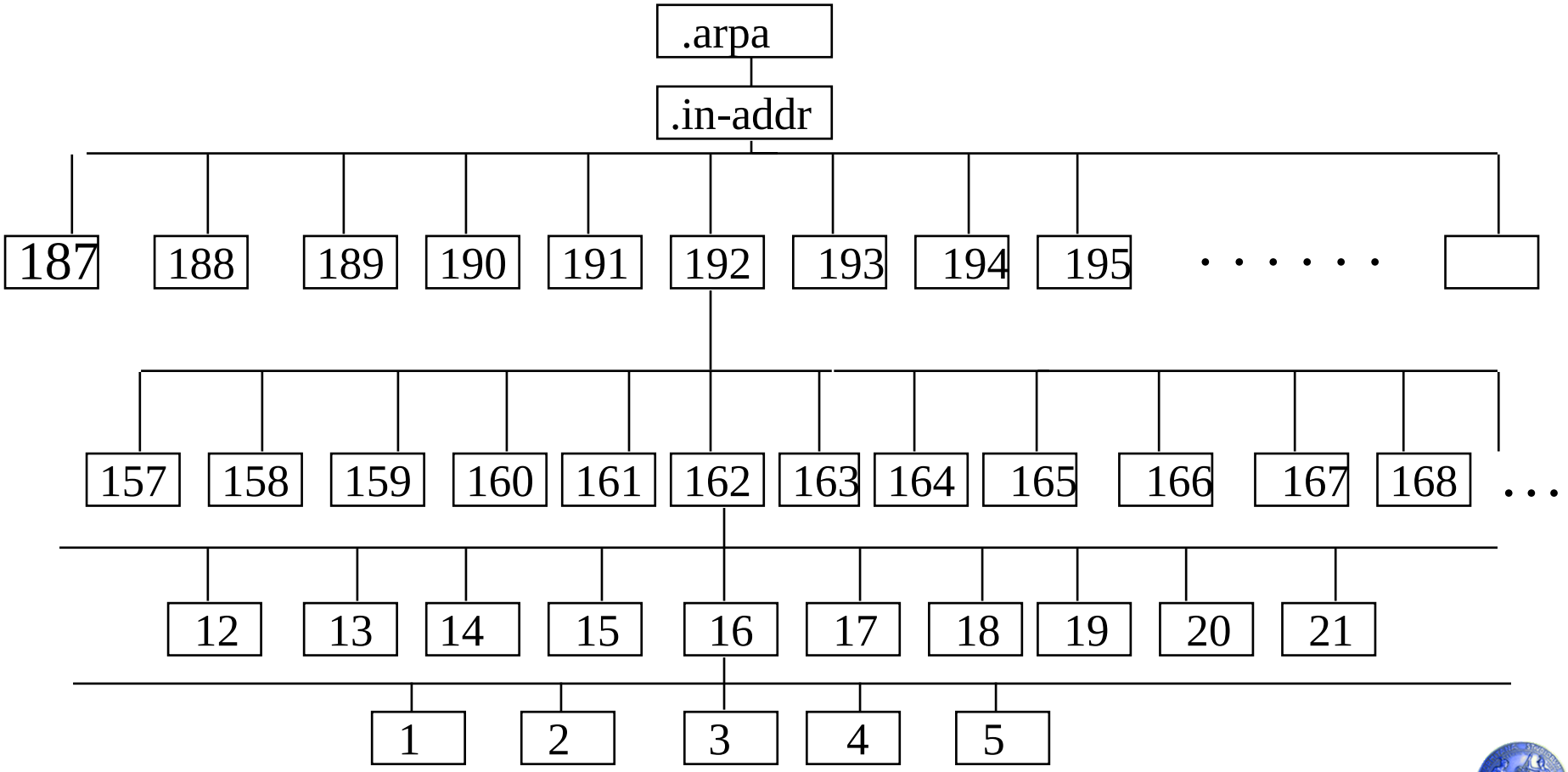


- Performance improvements
- supplementary problems
- & complexity

Reverse lookup

- **DNS allows to retrieve name from IP address.**
  - ⇨Used by destination host for accounting, authentication, access rights (IP packets do not contain names!)
- **special domain "in-addr.arpa" used:**
  - →to reverse lookup 131.175.21.1:
  - →direct lookup of 1.21.175.131.in-addr.arpa
- **Reverse domains form a hierarchical tree and are treated as any other Internet domain.**
- **Rfc2317 Classless In-ADDR.ARPA delegation**

In-addr.arpa tree



DB records and message formats

Resource Records

- **Information stored into DNS servers**
- **5-tuple format:**
  - Name
  - Time_To_Live
  - Class
  - Type
  - Rdata (or Value)

Name

- **Owner of the specific record**
  - i.e. the name to which this resource record pertains
  - the specific meaning of domain name and rdata depends on the type

Time To Live (TTL)

- How stable the record is (for caching purposes).
- in seconds (tipically 86400=1D for stable fields, 60=1M for unstable)

Class

- Always IN (Internet)
- may be different when DNS structure used for other distributed DB purposes

Type & value

What kind of record is & what value assumes.

basic types (many others practically unused):

- A = IP Address
- SOA = Start Of Authority
- MX = Mail Exchange
- NS = Name Server
- CNAME = Canonical Name
- HINFO = Host Description
- TXT = Text
- PTR = Pointer

IP address (A)

➔**The basic RR:**

- ⇨name = hostname
- ⇨rdata = IP address

cerbero.elet.polimi.it 86400 IN A 131.175.15.1

but try to look at the www.yahoo.com entry…!!

www.yahoo.com 86400 IN A 204.71.200.74

www.yahoo.com 86400 IN A 204.71.202.160

www.yahoo.com 86400 IN A 216.115.105.2

……

Mail Exchange (MX)

- **name = email domain**
- **rdata contains 2 fields:**
  - preference value
  - name of host that receives incoming email

often backup mailserver is listed (lower preference value = higher priority). Example:

```
unipa.it    MX      0  www.unipa.it
unipa.it    MX      10 sunipa.cuc.unipa.it
```

CNAME

- **Allows to register aliases**
  - ⇨ name is non-canonical domain name (alias)
  - ⇨ rdata is canonical domain name

www.elet.polimi.it	**CNAME**	**e45.elet.polimi.it**
mbox.unipa.it	**CNAME**	**www.unipa.it**

TXT, HINFO, PTR

- **HINFO**
  - allows to find out operating system and machine
- **TXT**
  - allows to store generic textual information
  - administrators generally store information about domain identification (e.g. name of organization, address, etc)
- **PTR**
  - a pointer to another part of the domain space
  - usually used to associate a name to an IP address

Information needed by DNS infrastructure

- **SOA**
  - exactly ONE record for each zone
  - stores administrative informations & flags
- **NS**
  - authoritative nameservers (primary and secondary) for the zone (possibly in random order…)
  - one record for each nameserver
  - rdata: server NAME (IP address is found checking the A entry for the server name!)

Start Of Authority infos

- **Name of master nameserver**
  - allows to determine primary nameserver: it was not possible with an NS query
- **email address of zone administrator**
- **serial number**
  - unique worldwide
- 4 configuration parameters
  - refresh
  - retry
  - expire
  - minimum ttl

Nslookup software

- **Standard command on unix machines**
  - few web interfaces around, most with extremely limited capabilities
- **Allows interactive domain lookups**
- **set querytype=ANY to dump all domain entries**
- **ls -d domainname to dump all DB**
  - remote servers refuse the query...

SOA + NS Examples (nslookup print format; queries to nameserver fusberta.elet.polimi.it)

elet.polimi.it

origin = morgana.elet.polimi.it
mail addr = root.morgana.elet.polimi.it
serial = 2000121203
refresh = 10800 (3H)
retry = 3600 (1H)
expire = 604800 (1W)
minimum ttl = 86400 (1D)

elet.polimi.it nameserver = ns.polimi.it
elet.polimi.it nameserver = fusberta.elet.polimi.it
elet.polimi.it nameserver = venus.elet.polimi.it
elet.polimi.it nameserver = morgana.elet.polimi.it
ns.polimi.it internet address = 131.175.12.1
fusberta.elet.polimi.it internet address = 131.175.21.8
venus.elet.polimi.it internet address = 131.175.26.5
morgana.elet.polimi.it internet address = 131.175.21.1

unipa.it

origin = sunipa.cuc.unipa.it
mail addr = root.sunipa.cuc.unipa.it
serial = 2000121200
refresh = 86400 (1D)
retry = 7200 (2H)
expire = 2592000 (4w2d)
minimum ttl = 172800 (2D)

unipa.it nameserver = sunipa.cuc.unipa.it
unipa.it nameserver = cucaix.cuc.unipa.it
unipa.it nameserver = dns2.nic.it
sunipa.cuc.unipa.it internet address = 147.163.1.22
cucaix.cuc.unipa.it internet address = 147.163.1.3
dns2.nic.it internet address = 193.205.245.8

Example: yahoo it & com

yahoo.it

origin = ns0.corp.yahoo.com
mail addr = hostmaster.yahoo-inc.com
serial = 2000121201
refresh = 3600 (1H)
retry = 1800 (30M)
expire = 604800 (1W)
minimum ttl = 21600 (6H)

yahoo.it nameserver = ns.europe.yahoo.com
yahoo.it nameserver = ns.yahoo.com
yahoo.it nameserver = av1.yahoo.com
ns.yahoo.com internet address = 204.71.177.33
av1.yahoo.com internet address = 204.123.2.85

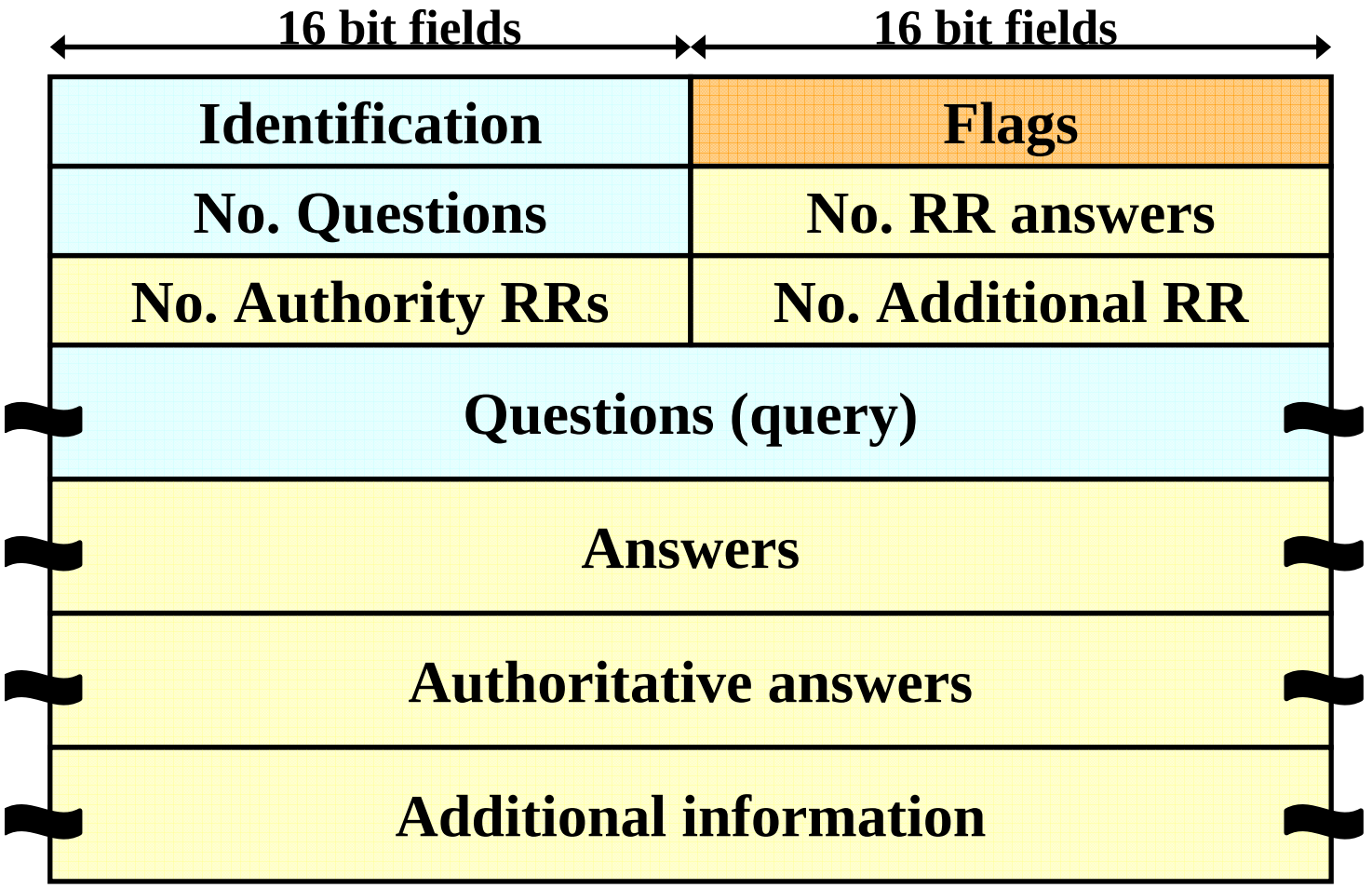
yahoo.com

origin = ns0.corp.yahoo.com
mail addr = hostmaster.yahoo-inc.com
serial = 2000121305
refresh = 1800 (30M)
retry = 900 (15M)
expire = 1209600 (2W)
minimum ttl = 9 (9S)

yahoo.com nameserver = ns1.yahoo.com
yahoo.com nameserver = ns3.europe.yahoo.com
yahoo.com nameserver = ns5.dcx.yahoo.com
ns1.yahoo.com internet address = 204.71.200.33
ns3.europe.yahoo.com internet address = 194.237.108.51
ns5.dcx.yahoo.com internet address = 216.32.74.10

DNS Message Format

12 bytes fixed header + variable payload



Same UDP packet up & down

- request
- response
- both

Università degli Studi di Palermo

Identification

Set by client, returned unmodified by server
(to match request with response)

Message Flags

- QR: Query=0, Response=1
- AA: Authoritative Answer
- TC: response truncated (> 512 bytes)
- RD: recursion desired
- RA: recursion available
- rcode: return code

Query & Response formats

- **Question format:**
  - Name: domain name (or IP address)
  - Query type (A=1, NS=2, MX=15, CNAME=12…)
  - Query class (1 for IN = Internet)
- **Response resource record**
  - Domain Name
  - Response type
  - Class (IN)
  - Time to live (in seconds)
  - Length of resource data
  - Resource data