Communication Networks:
Technology & Protocols

Stavros Tripakis (stavros@eecs)

Lecture 7

September 8

Logistics

- Web site:
  - www.cs.berkeley.edu/~amc/eecs122
- **Homework 2** available on web-site.
  - Homework 1 due this Friday (9/10): **will count for enrollment**.
  - Homework 2 due Monday 9/13.
- Textbook: check publisher's web-site for errata, on-line material, etc:

  www.mhhe.com

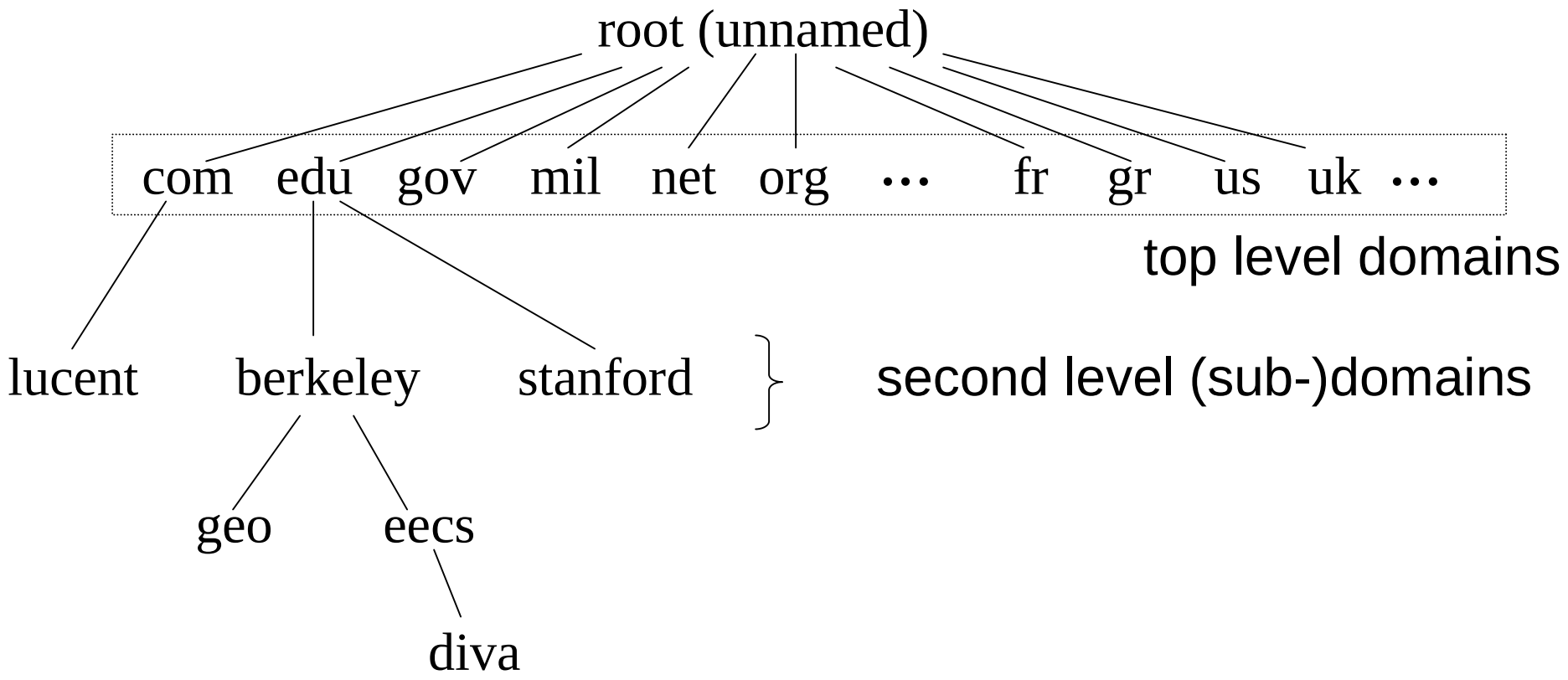
Names and IP addresses

- Domain Name System (DNS)
  - Hierarchical naming space
  - Name-granting authorities
  - DNS servers and name resolution
- IP addressing
  - Hierarchical structure
  - Class-based
  - Subnetting
  - Classless

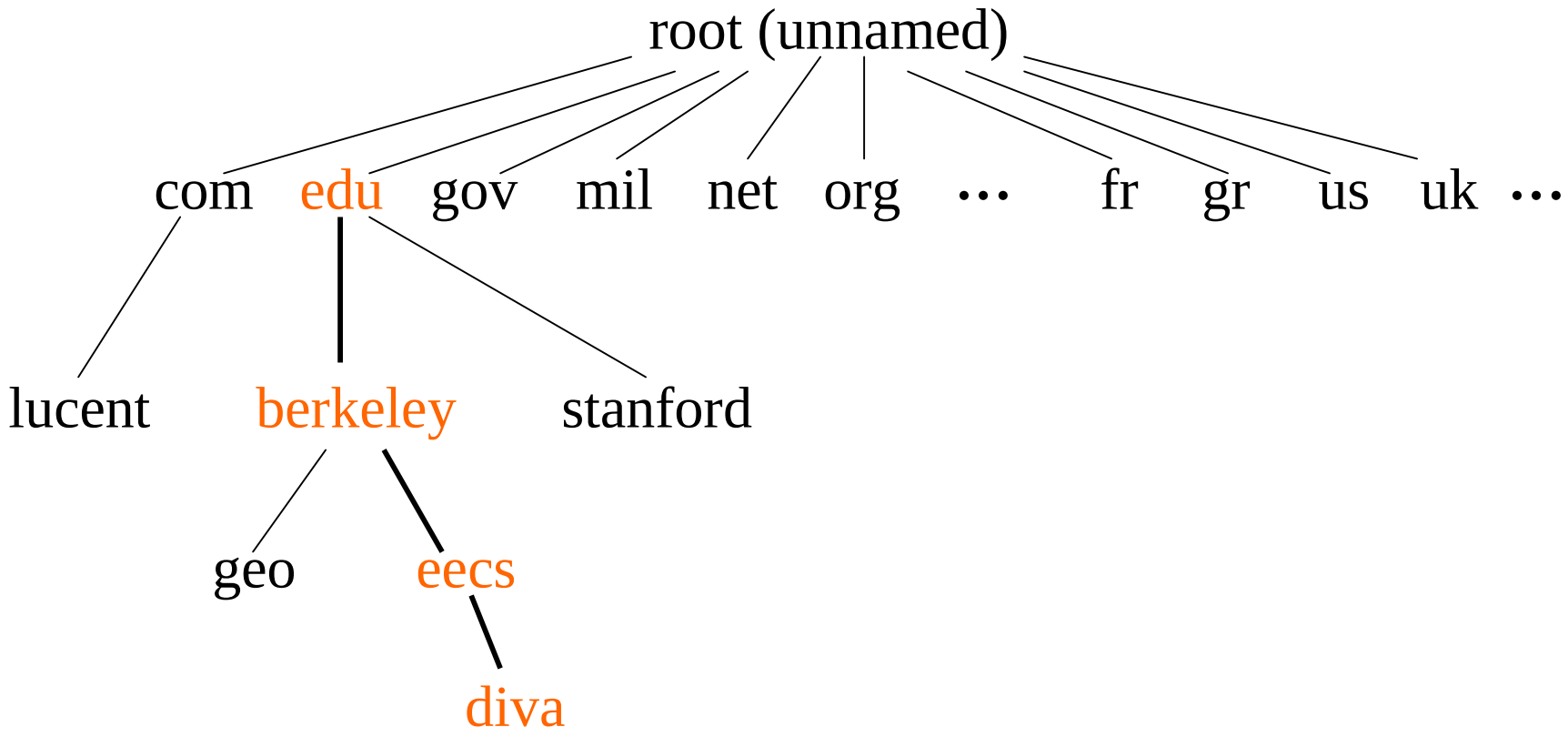
Names and addresses: why both?

- Name: stout.eecs.berkeley.edu
- IP address: 128.32.239.44
  - (Also Ethernet or other link-layer addresses.)
- IP addresses are fixed-size numbers.
  - 32 bits. 128.32.239.44 = 10000000.00100000.11101111.00101100
- Names are memorizable, flexible:
  - Variable-length
  - Many names for a single IP address.
  - Change address doesn't imply change name.

Domain name structure

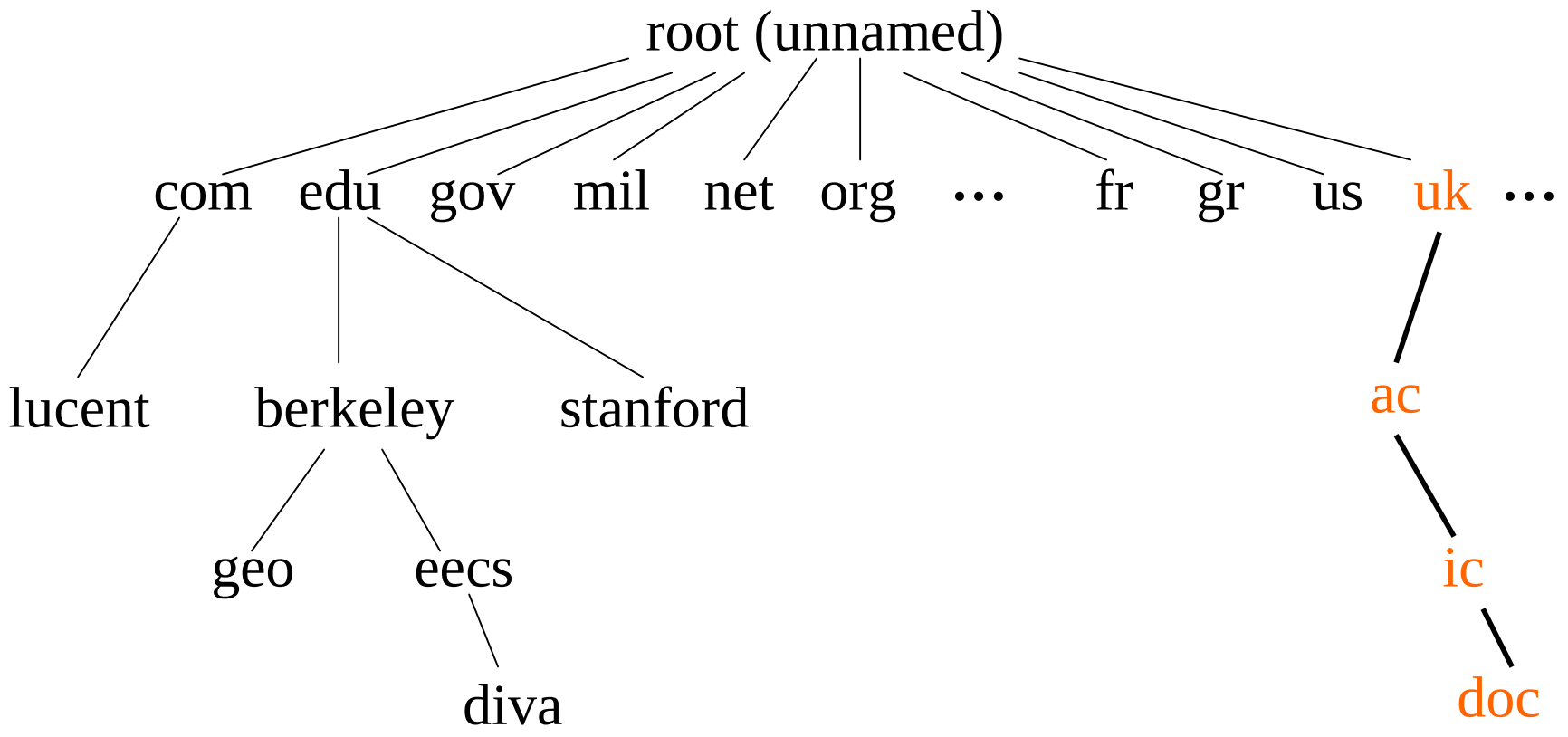


Domain name structure



Full name: diva.eecs.berkeley.edu

Domain name structure

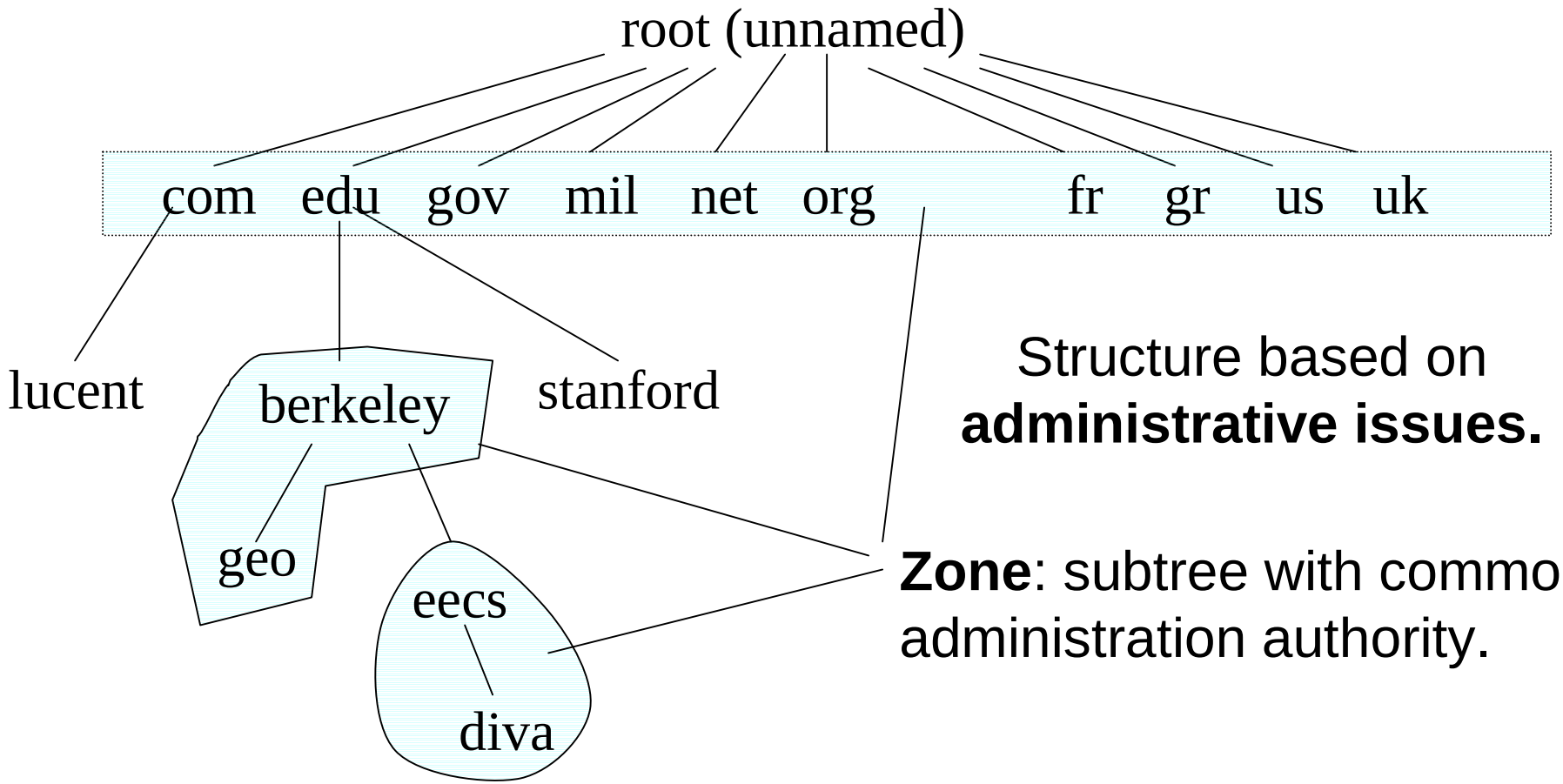


other names: scorch.doc.ic.ac.uk

Name-granting authorities

- Name management: Internet Assigned Number Authority, www.iana.org, to be replaced by Internet Corporation for Assigned Names and Numbers, www.icann.org.
- Internet Registries:
  - RIPE for Europe, www.ripe.net
  - APNIC for Asia and Pacific, www.apnic.net
  - InterNIC for US and the rest, www.apnic.net
    (in fact, today it is NetworkSolutions.com)

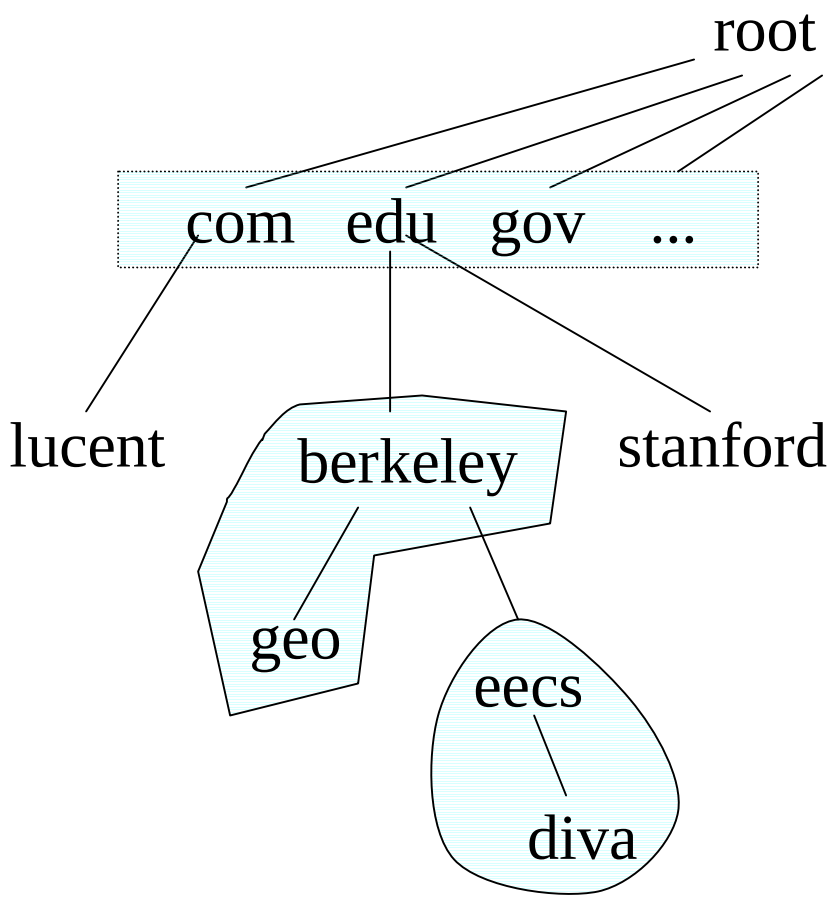
Domain name structure

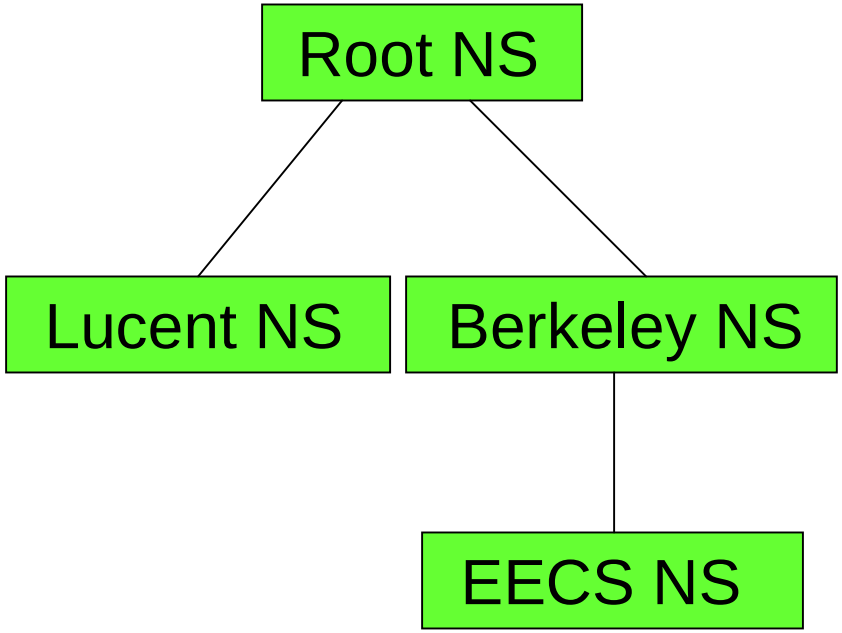


Structure based on **administrative issues.**

Zone: subtree with common administration authority.

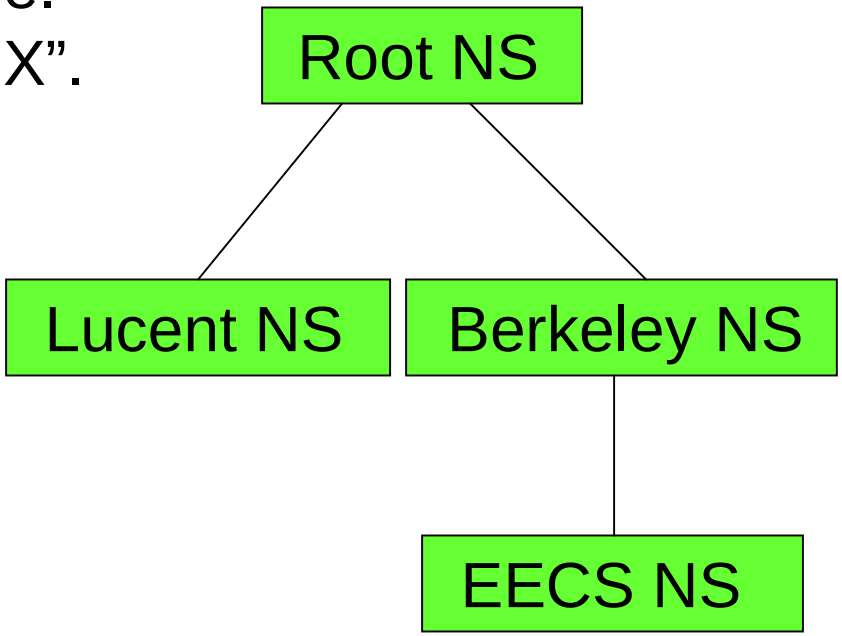
Name Servers (NS)





Name Servers (NS)

- NS: a process running on a host.
- Maps names to IP addresses (**name resolution**).
- Each NS is responsible for a zone.
- Request: “give me IP address of X”.
- Response:
  - either A (IP address of X)
  - or “contact name server N to get the address of X”.



- A, N = IP addresses
- X = name

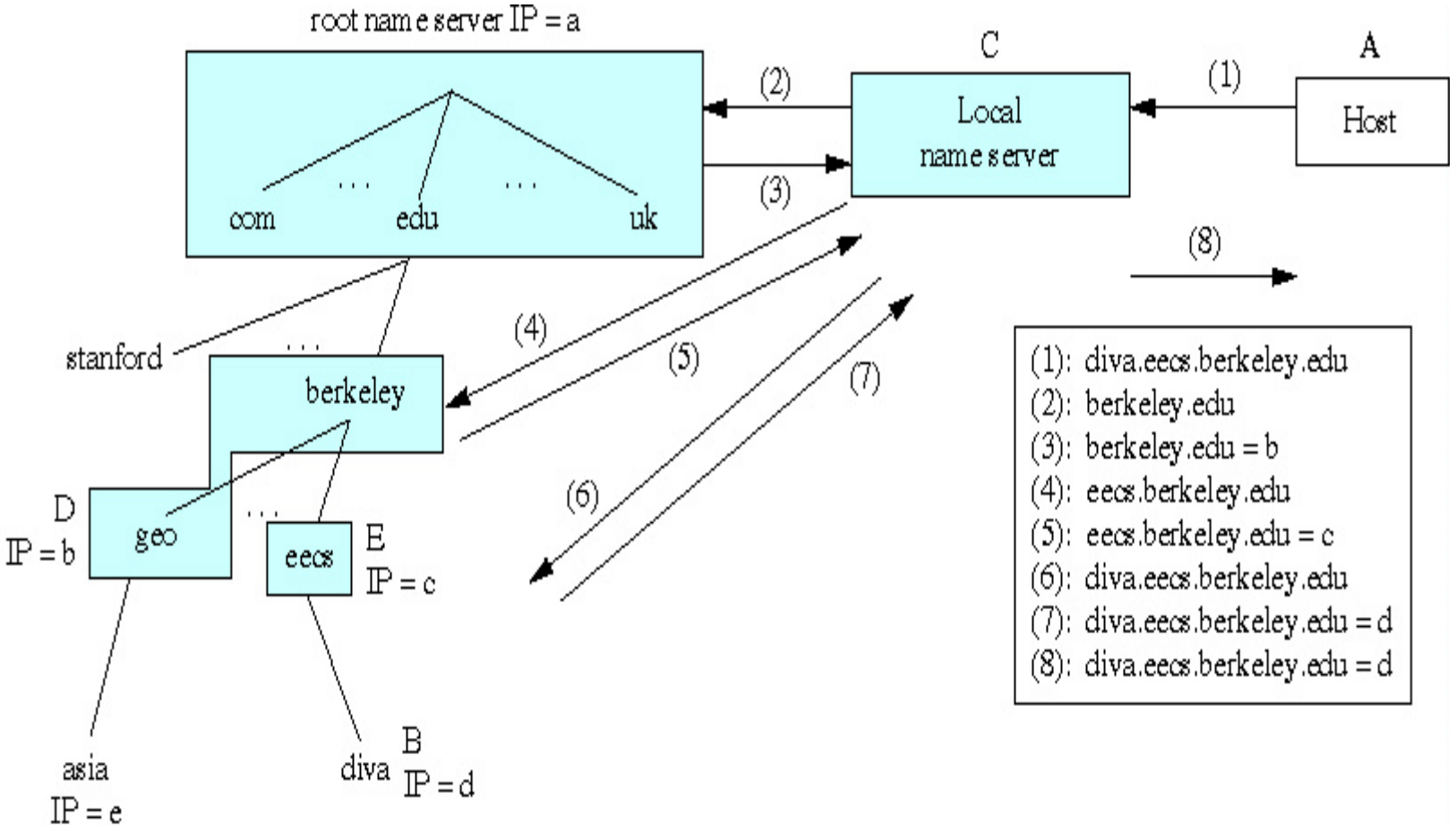
Name Servers (NS)

- NSs are **duplicated** for reliability.
- Anonymous ftp from:

  ftp.rs.internic.net, netinfo/root-server.txt

  gives the current root NSs (about 10).
- Each host knows the IP address of the **local** NS.
- Each NS knows the IP addresses of all root NSs.
- **Caching** is used.

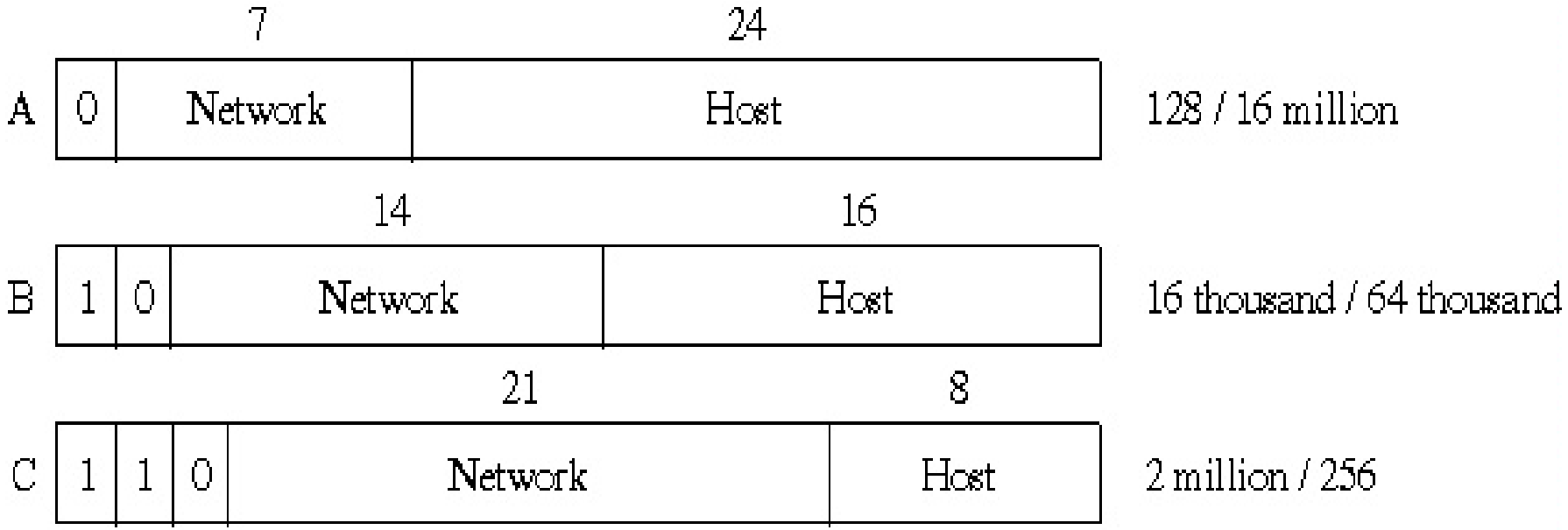
Name resolution: example



IP addressing

- **Hierarchical,** based on geographical location, like telephone numbers: +1 510 642 5649
  - Scalability in **routing**.
  - Scalability in assigning addresses.
- 3 stages:
  - class-based
  - subnetting
  - classless

Class-based addressing



E.g., UC Berkeley network address:
128.32 = 10000000.00100000 (class B network).

Class-based addressing: problems

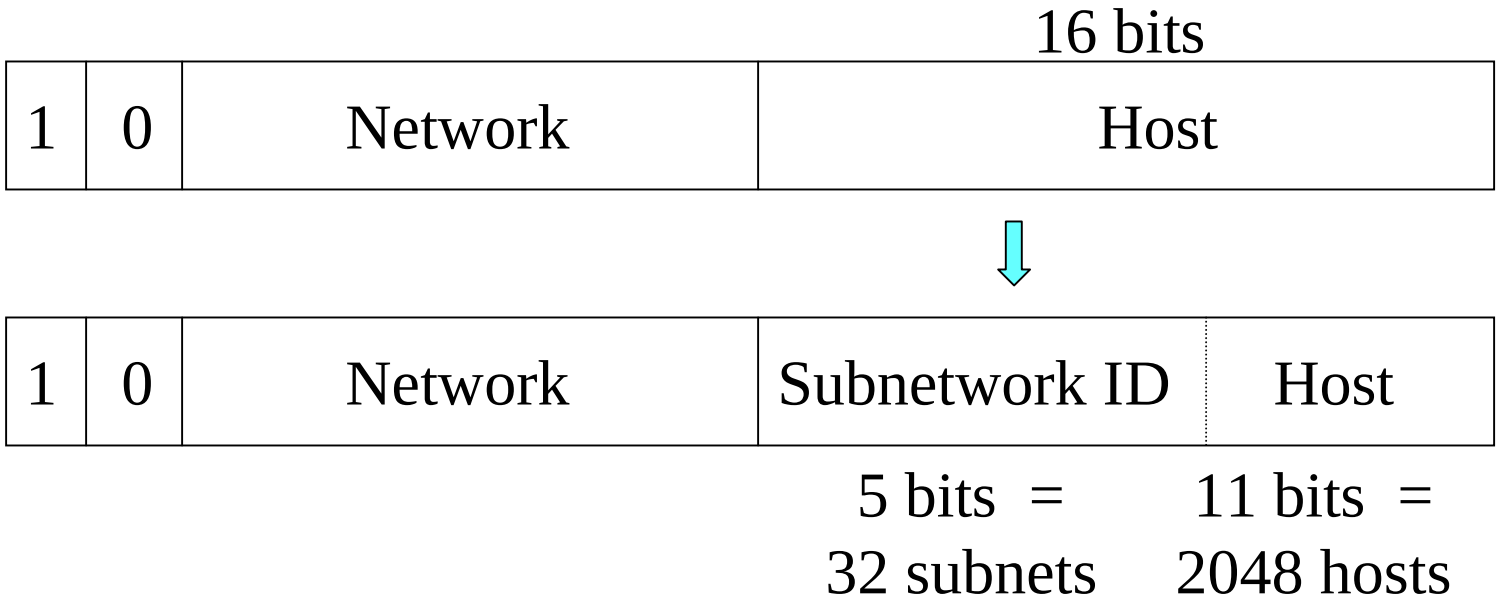
- Class A networks: too big, too few.
- Class C networks: too small, too many.
- Class B networks: not enough (run out of already).
- However:
  - 128 x 16 million + 16,000 x 64,000 + 2 million x 256
    $\approx$ 4 billion !!!

  $\Rightarrow$ should have enough addresses for everybody.
- Problem: bad utilization.
- Question: how to have an addressing scheme that meets exactly the needs of the users ?

Subnetting

- Idea: 10 depts. x 2000 hosts per dept.

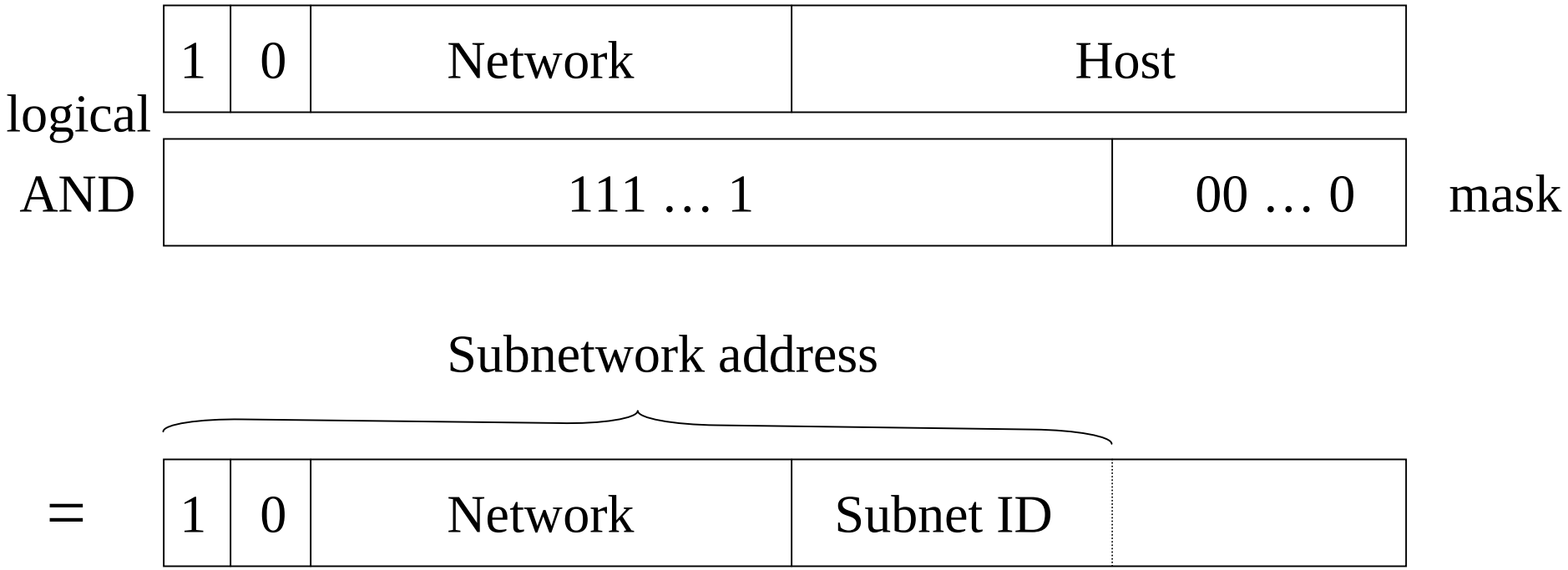
  = 1 class B network address (instead of 10)



Subnetting

- Implementation:

  Subnetwork address = IP address AND mask



Subnetting: problem

- Relatively small networks (between 256 and 1000 hosts) still need at least a class B network address. (In fact, so does any network that could prospectively have more than 255 hosts.)

Classless addressing (supernetting)

- Idea: aggregate many class C networks under a common network address, e.g., get 16 consecutive class C addresses: 192.4.16-31 = 16 x 256 = 4096 hosts.
- All addresses share common **prefix**: first 20 bits:

  11000000.00000100.0001xxxx
- This prefix is the "supernet" address (it defies class boundaries, something between class B and class C network).

Classless addressing (supernetting)

- Generalization of subnetting:
  - Subnetting: split class-based address to multiple subnet addresses.
  - Supernetting: also allow aggregation of multiple class-based addresses into a supernet address.
- Current Internet routing protocols use subnetting and supernetting (CIDR):
  - View a collection of subnets as a single IP address.
  - View a collection of IP addresses as a single supernet address.