# Introduction to LAN/WAN

## Application Layer

# Application Layer Topics

- Domain Name System (DNS) (7.1) ←
- Electronic Mail (E-mail) (7.2)
- World Wide Web (7.3)

# Domain Name System(DNS)

- Humans love ASCII addresses
  - E.g. *tana@art.ucsb.edu*
- Machines prefer numbers:
  - E.g: *tana@128.111.24.41*
- Need to associate human-readable ASCII with machine readable numbers
  - *art.ucsb.edu == 128.111.24.41*
- Simple solution:
  - Text file *hosts.txt* on every machine with mappings
  - Machines update this file every night

# Domain Name System(DNS)

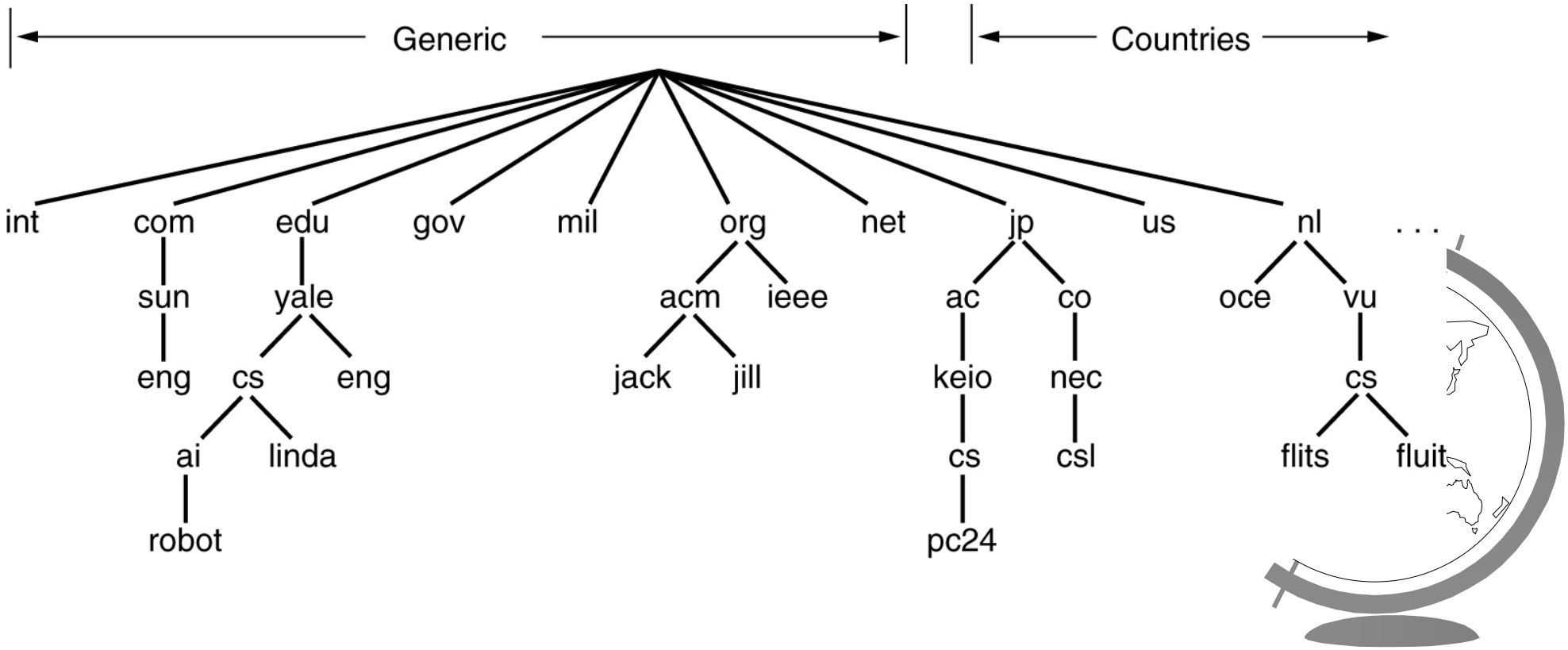
- Simple file-based solution works well for small network
- Larger network
  - Updates become a headache
  - Name conflicts would give people ulcers
  - DNS: one distributed hierarchical mapping “database”
- DNS Steps:
  - Application program calls *resolver* procedure, passes parameters (e.g. *gethostbyname( )*)
  - Resolver sends UDP packet to local DNS server
  - Local DNS server looks up name and returns IP address
  - Calling application (e.g. email, web) then uses IP address

# Domain Name Space

- Managing large and constantly changing human-readable sets of names and IP addresses is a non-trivial problem
- 200 top level domains (.*edu*, .*gov*, .*uk*,.. Etc)
- Two categories of domains:
  - generic and countries
- Initially 6 generic domains
  - *com (*commercial*)*, *edu* (educational), *gov* (government), *mil* (military), *net* (network providers), *org* (non-profit), and *int* (int. organizations)
- Countries: one for each country as defined in ISO 3166

# Domain Name Space

- In November 2000, ICANN approved four new general-purpose, top-level domains
  - *biz* (businesses), *info* (information), *name* (people's names) and *pro* (professionals like lawyers, doctors)



# Domain Name Space

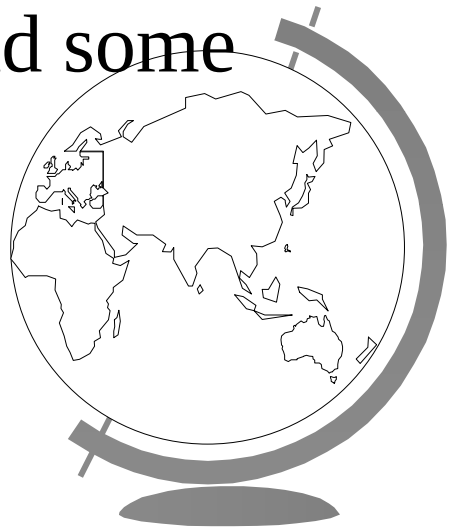
- Getting second-level domain like *EmmanuelAgu.com* is easy
  - Simply contact registrar to find out if name is taken
  - Registrar's: www.networksolutions.com
  - Registrar also checks for trademark infringements
  - If available, Emmanuel simply pays small fee and launches
- Domain is named by path upward from it to named root, separated by periods
  - E.g: Engr. Department at Sun Microsystems (*eng.sun.com*)
  - Due to hierarchy, no potential conflicts with *eng.yale.edu*
- Sub-domains applications to manager:
  - E.g. New VLSI dept. at Yale (*vlsi.cs.yale.edu*) simply talks to the manager (system admin) of (*cs.yale.edu*)

# Resource Records

- Information about *domain-IP address* mappings are stored as resource records
- So DNS lookup returns complete resource record
- Could contain more than *domain-IP address* mappings
- Resource record is five tuple of (*Domain name*, *Time_to_live*, *Class*, *Type*, *Value*)

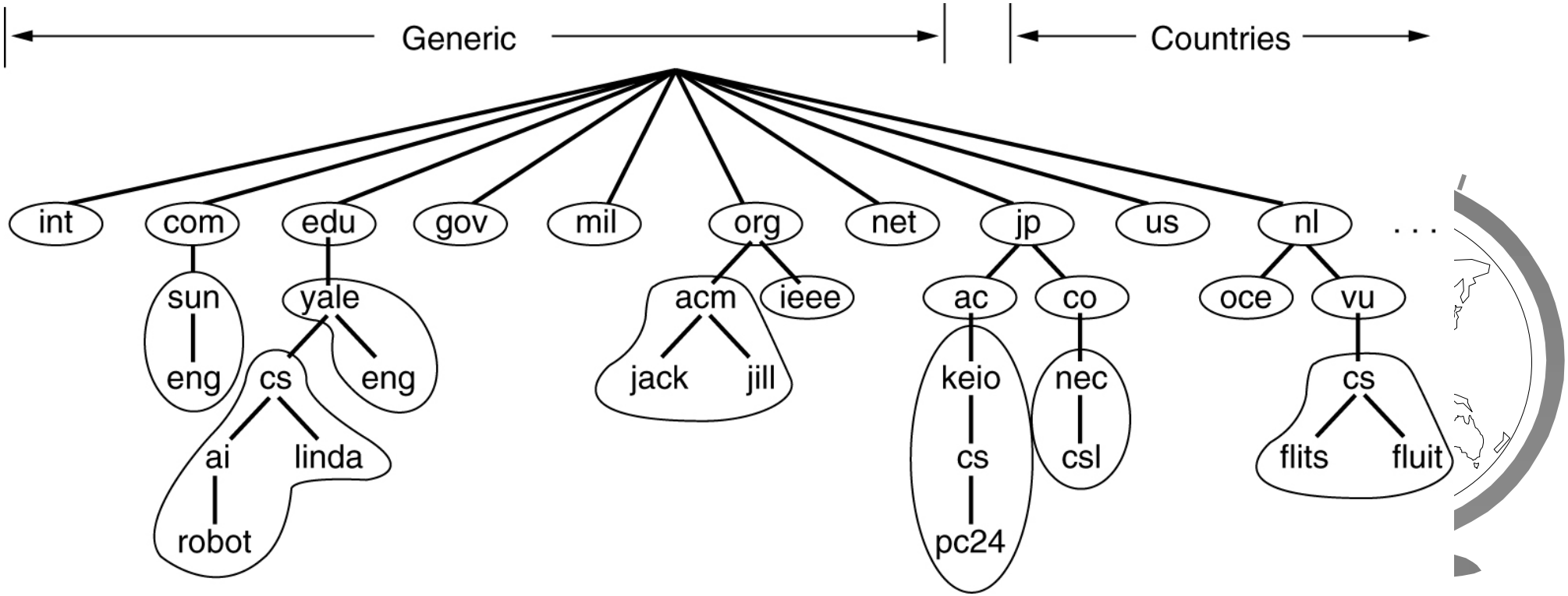
# Name Servers

- Theoretically, single server could work
- In practice,
  - this one server would be overloaded
  - Also, where to put it? Australia?
- Solution
  - DNS name space divided into nonoverlapping zones
  - Each zone contains some part of a tree and some nameservers for that zone



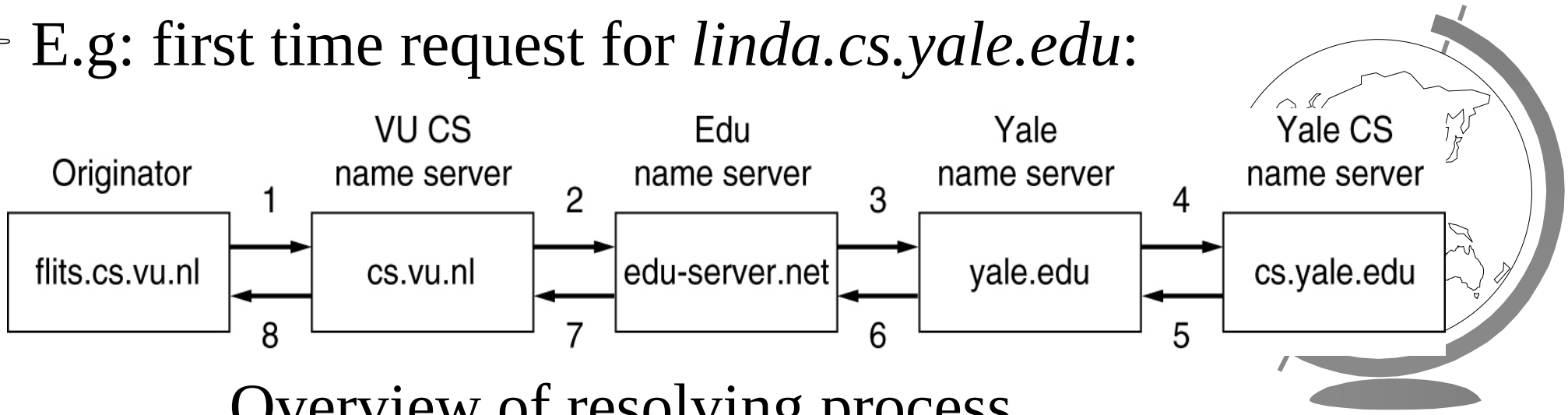
# Name Servers: Zones

- Usually one primary nameserver per zone
- Possibly many secondary nameservers
- Primary nameservers read records from file on disk
- Secondaries get information from primary nameserver



# Resolving Process

- When resolver gets query, passes it to one of local nameservers
- If domain under queried nameserver, returns *authoritative record*
- *Authoritative* comes from file, not cached (outdated)
- If requested domain is remote, send message to top-level name server for requested domain
- E.g: first time request for *linda.cs.yale.edu*:




Overview of resolving process

# Resolving Process

- DNS queries are recursive first time around
- Results are cached for specified (*time_to_live*)
- *time_to_live* may be varied for different addresses
- Addresses that haven't changed in years could get higher *time_to_live* than volatile ones (owners in court….cache for seconds!!)
- Next:
  - Email
  - WWW (maybe)